7TH EDITION

# THE FINITE ELEMENT METHOD
## ITS BASIS & FUNDAMENTALS

O.C. Zienkiewicz, R.L. Taylor & J.Z. Zhu

B H

# The Finite Element Method: Its Basis and Fundamentals

**Professor O.C. Zienkiewicz**, CBE, FRS, FREng died on 2 January 2009. Prior to his death he was Professor Emeritus at the Civil and Computational Engineering Centre, University of Wales, Swansea and previously Director of the Institute for Numerical Methods in Engineering at the University of Wales, Swansea, UK. He also held the UNESCO Chair of Numerical Methods in Engineering at the Technical University of Catalunya, Barcelona, Spain. He was the head of the Civil Engineering Department at the University of Wales, Swansea between 1961 and 1989. During this period he established that department as one of the primary centers of finite element research. In 1968 he became the Founder Editor of the *International Journal for Numerical Methods in Engineering* which still remains today the major journal in this field. The recipient of 27 honorary degrees and many medals, Professor Zienkiewicz was a member of five academies—an honor he received for his many contributions to the fundamental developments of the finite element method. In 1978, he became a Fellow of the Royal Society and the Royal Academy of Engineering. This was followed by his election as a foreign member to the U.S. Academy of Engineering (1981), the Polish Academy of Science (1985), the Chinese Academy of Sciences (1998), and the National Academy of Science, Italy (Academia dei Lincei) (1999). He published the first edition of this book in 1967 and it remained the only book on the subject until 1971.

**Professor R.L. Taylor** has more than 50 years' experience in the modeling and simulation of structures and solid continua including eight years in industry. He is Professor of the Graduate School and the Emeritus T.Y. and Margaret Lin Professor of Engineering at the University of California at Berkeley and Corporate Fellow at Dassault Systèmes SIMULIA in Providence, Rhode Island. In 1991 he was elected to membership in the US National Academy of Engineering in recognition of his educational and research contributions to the field of computational mechanics. He is a Fellow of the US Association of Computational Mechanics—USACM (1996) and a Fellow of the International Association of Computational Mechanics—IACM (1998). He has received numerous awards including the Berkeley Citation, the highest honor awarded by the University of California at Berkeley, the USACM John von Neumann Medal, the IACM Gauss–Newton Congress Medal, and a Dr.-Ingenieur ehrenhalber awarded by the Technical University of Hannover, Germany. He has written several computer programs for finite element analysis of structural and non-structural systems, one of which, *FEAP*, is used worldwide in education and research environments. A personal version, *FEAPpv*, available at his UC website, is incorporated into this book.

**Dr. J.Z. Zhu** has more than 30 years' experience in the development of finite element methods. During the last 20 years he has worked in industry where he has been developing commercial finite element software to solve multi-physics problems. Dr Zhu read for his Bachelor of Science degree at Harbin Engineering University and his Master of Science at Tianjin University, both in China. He was awarded his doctoral degree in 1987 from the University of Wales, Swansea, working under the supervision of Professor Zienkiewicz. Dr Zhu is the author of more than 40 technical papers on finite element methods including several on error estimation and adaptive automatic mesh generation. These have resulted in his being named in 2000 as one of the most highly cited researchers for engineering in the world and in 2001 as one of the top 20 most highly cited researchers for engineering in the United Kingdom.

# The Finite Element Method: Its Basis and Fundamentals

## Seventh Edition

**O.C. Zienkiewicz, CBE, FRS**
Previously UNESCO Professor of Numerical Methods in Engineering
International Centre for Numerical Methods in Engineering, Barcelona
Previously Director of the Institute for Numerical Methods in Engineering
University of Wales Swansea

**R.L. Taylor**
Professor in the Graduate School
Department of Civil and Environmental Engineering
University of California at Berkeley
Berkeley, California

**J.Z. Zhu**
Senior Scientist
ESI US R & D
9891 Broken Land Parkway, Suite 200
Columbia, Maryland

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Butterworth-Heinemann is an imprint of Elsevier

Butterworth-Heinemann is an imprint of Elsevier
The Boulevard, Langford Lane, Kidlington, Oxford, OX5 1GB, UK
225 Wyman Street, Waltham, MA 02451, USA

First published 1967 by McGraw-Hill
Fifth edition published by Butterworth-Heinemann 2000
Reprinted 2002
Sixth edition 2005
Seventh edition 2013



**Notices**

Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods, professional practices, or medical treatment may become necessary.

Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information, methods, compounds, or experiments described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility.

To the fullest extent of the law, neither the Publisher nor the authors, contributors, or editors, assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

**British Library Cataloguing in Publication Data**

A catalogue record for this book is available from the British Library

**Library of Congress Cataloging in Publication Data**

A catalog record for this book is available from the Library of Congress

ISBN: 978-1-85617-633-0



Printed and bound in the United Kingdom

13 14 15 16 10 9 8 7 6 5 4 3 2 1

*This book is dedicated to the memory of*
*Olgierd C. (Olek) Zienkiewicz:*
*Pioneer, mentor, and close friend.*

This page is intentionally left blank

# Contents

This page is intentionally left blank

# List of Figures

This page intentionally left blank

# List of Tables

# Preface

The present revision of *The Finite Element Method* was undertaken shortly before the passing in January 2009 of our close friend and co-author Olgierd C. (Olek) Zienkiewicz. His inspiration and guidance have been greatly missed in the intervening years. However, we hope that the essence of his writings is retained in the new work so that current and future scholars can continue to benefit from his insights and many contributions to the field of computational mechanics. The story of his life and works is summarized in *International Journal for Numerical Methods in Engineering*, **80**, 2009, pp. 1–45.

It is now 46 years since the *The Finite Element Method in Structural and Continuum Mechanics* was first published by Olek Zienkiewicz. This book, which was the first dealing with the finite element method, provided the basis from which many further developments occurred. The expanding research and field of application of finite elements led to the second edition in 1971, the third in 1977, the fourth as two volumes in 1989 and 1991, and the fifth and sixth as three volumes in 2000 and 2005, respectively. The size of each of these editions expanded geometrically (from 272 pages in 1967 to the sixth edition of nearly 1800 pages). This was necessary to do justice to a rapidly expanding field of professional application and research. Even so, much filtering of the contents was necessary to keep these editions within reasonable bounds.

In the present edition we have retained the complete works as three separate volumes, each one capable of being used without the others and each one appealing perhaps to a different audience.

In particular this first volume, *The Finite Element Method: Its Basis and Fundamentals,* is designed to cover quite completely all the steps necessary to solve problems represented by linear differential equations. Applications to problems of elasticity, field problems, and plate and shell structural problems form the primary basis from which the finite element steps are enumerated. After a summary of the basic equations in matrix form, two new chapters on applications to one-dimensional problems are added. This describes the steps necessary to convert a differential equation to a form from which finite element approximation may be made in a very simple context. Two methodologies are presented: weak forms (which may be used for any linear differential equation) and variational theorems (which are restricted here to steady-state applications). The basic concepts include interpolation of solution variables, numerical integration to evaluate the final matrices appearing in the finite element approximation, and solution of the resulting matrix equations. Both steady-state and transient problems are covered at an early point to permit the methods to be available for later applications.

Subsequent chapters extend the finite element theory to applications of multi-dimensional field and elasticity problems. In addition the general concepts necessary to construct interpolation forms for various element shapes, differentiate parametric forms, and carry out numerical integration are consolidated into a single chapter. A

chapter on the patch test is used to establish engineering criteria to measure convergence of a finite element formulation.

An important application is dealing with constraints in a theory. In elasticity a constraint is imposed when the Poisson ratio approaches one-half. In plate and shell problems modeled by a shear deformable theory a constraint is imposed as the thickness becomes small compared to other dimensions. To be able to treat both types of constraints we have moved the treatment of plates and shells for linear behavior to this volume. This allows for a more complete discussion on treatment methods to be considered. We believe that understanding the effects of constraints is essential before treating such topics as large strain or inelastic material behavior.

This volume concludes with chapters on error estimation, adaptivity, and mesh generation. A beginner in the finite element field will find very rapidly that much of the work of solving problems consists of preparing a suitable mesh to deal with the whole problem and as the size of computers has seemed to increase without limits the size of problems capable of being dealt with is also increasing. Thus, meshes containing sometimes many million nodes have to be prepared with details of the material interfaces, boundaries, and loads being well specified. There are many books devoted exclusively to the subject of mesh generation but we feel that the essence of dealing with this difficult problem should be included here for those who wish to have a solid grasp of the subject.

To permit the reader to solve problems or extend concepts covered in the text a computer program called *FEAPpv* and user instructions are available at the authors' web page: http://www.ce.berkeley.edu/feap/feappv.

Each chapter concludes with a set of problems which the reader may use to test comprehension of the basic theory and applications covered.

The two further volumes form again separate books. The first of these is entitled *The Finite Element Method in Solid and Structural Mechanics* and the second is a text entitled *The Finite Element Method in Fluid Dynamics*. Each of these two volumes is a stand-alone text which provides the full knowledge of the subject for those who have acquired an introduction to the finite element method through other texts or this volume.

We emphasize here the fact that all three books stress the importance of considering the finite element method as a unique and whole basis of approach and that the method contains many of the other numerical analysis methods as special cases.

R.L. Taylor and J.Z. Zhu

CHAPTER

# 1 The Standard Discrete System and Origins of the Finite Element Method

## 1.1 Introduction

The limitations of the human mind are such that it cannot grasp the behavior of its complex surroundings and creations in one operation. Thus the process of subdividing all systems into their individual components or "elements," whose behavior is readily understood, and then rebuilding the original system from such components to study its behavior is a natural way in which the engineer, the scientist, or even the economist proceeds.

In many situations an adequate model is obtained using a finite number of well-defined components. We shall term such problems *discrete*. In others the subdivision is continued indefinitely and the problem can only be defined using the mathematical fiction of an infinitesimal. This leads to differential equations or equivalent statements which imply an infinite number of elements. We shall term such systems *continuous*.

With the advent of digital computers, *discrete* problems can generally be solved readily even if the number of elements is very large. As the capacity of all computers is finite, *continuous* problems can only be solved exactly by mathematical manipulation. The available mathematical techniques for exact solutions usually limit the possibilities to oversimplified situations.

To overcome the intractability of realistic types of continuous problems (a continuum), various methods of *discretization* have been proposed by engineers, scientists, and mathematicians. All involve an *approximation* which, hopefully, approaches in the limit the true continuum solution as the number of discrete variables increases.

The discretization of continuous problems has been approached differently by mathematicians and engineers. Mathematicians have developed general techniques applicable directly to differential equations governing the problem, such as finite difference approximations [1–3], various weighted residual procedures [4,5], or approximate techniques for determining the stationarity of properly defined "functionals" [6]. The engineer, on the other hand, often approaches the problem more intuitively by creating an analogy between real discrete elements and finite portions of a continuum domain. For instance, in the field of solid mechanics in the 1940s McHenry [7], Hrenikoff [8], Newmark [9], and Southwell [2] showed that reasonably good solutions to an elastic continuum problem can be obtained by replacing small portions of the continuum by an arrangement of simple elastic bars. Later, in the same context, Turner et al. [10] showed that a more direct, but no less intuitive, substitution

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00001-0

of properties can be made much more effectively by considering that small portions or "elements" in a continuum behave in a simplified manner.

It is from the engineering "direct analogy" view that the term "finite element" was born. Clough [11] appears to be the first to use this term, which implies in it a direct use of a *standard methodology applicable to discrete systems* (see also Ref. [12] for a history on early developments). Both conceptually and from the computational viewpoint this is of the utmost importance. The first allows an improved understanding to be obtained; the second offers a unified approach to the variety of problems and the development of standard computational procedures.

Since the early 1960s much progress has been made, and today the purely mathematical and "direct analogy" approaches are fully reconciled. It is the object of this volume to present a view of the finite element method as *a general discretization procedure of continuum mechanics problems posed by mathematically defined statements*.

In the analysis of problems of a discrete nature, a standard methodology has been developed over the years. The civil engineer, dealing with structures, first calculates force-displacement relationships for each element of the structure and then proceeds to assemble the whole by following a well-defined procedure of establishing local equilibrium at each "node" or connecting point of the structure. The resulting equations can be solved for the unknown displacements. Similarly, the electrical or hydraulic engineer, dealing with a network of electrical components (resistors, capacitances, etc.) or hydraulic conduits, first establishes a relationship between currents (fluxes) and potentials for individual elements and then proceeds to assemble the system by ensuring continuity of flows.

All such analyses follow a standard pattern which is universally adaptable to discrete systems. It is thus possible to define a *standard discrete system*, and this chapter will be primarily concerned with establishing the processes applicable to such systems. Much of what is presented here will be known to engineers, but some reiteration at this stage is advisable. As the treatment of elastic solid structures has been the most developed area of activity this will be introduced and generalized along with examples from other fields.

The existence of a unified treatment of "standard discrete problems" leads us to the first definition of the finite element process as a method of approximation to continuum problems such that

**(a)** the continuum is divided into a finite number of parts (elements), the behavior of which is specified by a finite number of parameters, and

**(b)** the solution of the complete system as an assembly of its elements follows precisely the same rules as those applicable to *standard discrete problems*.

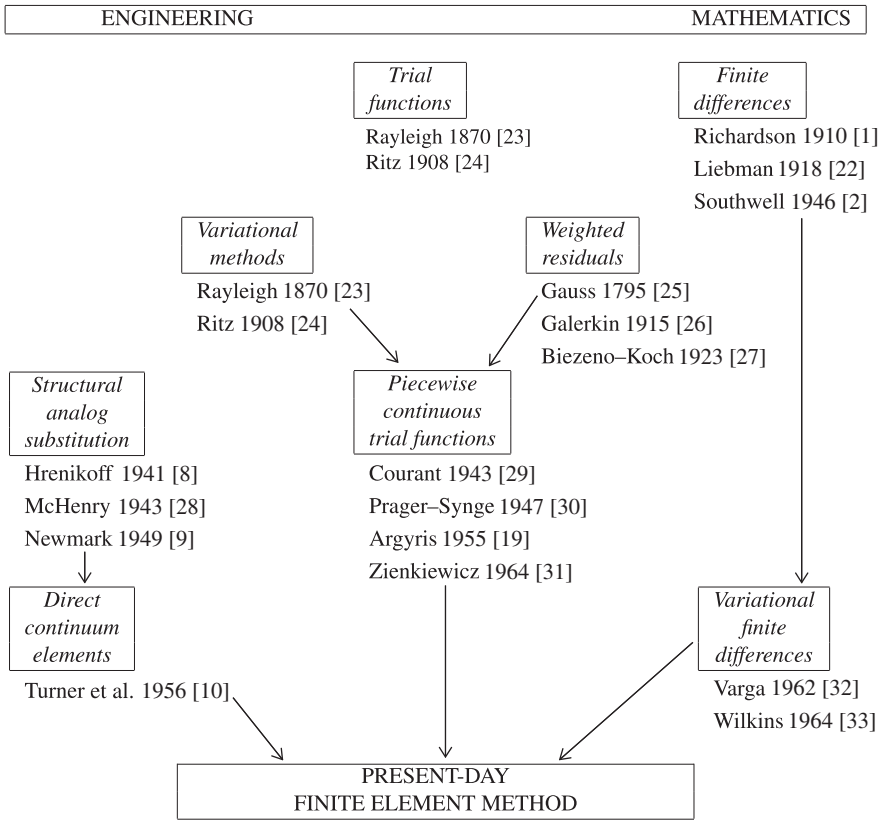
The development of the standard discrete system can be followed most closely through the work done in structural engineering during the 19th and 20th centuries. It appears that the "direct stiffness process" was first introduced by Navier in the early part of the 19th century and brought to its modern form by Clebsch [13] and others. In the 20th century much use of this has been made and Southwell [14], Cross [15],

and others have revolutionized many aspects of structural engineering by introducing a relaxation iterative process. Just before the Second World War matrices began to play a larger part in casting the equations and it was convenient to restate the procedures in matrix form. The work of Duncan and Collar [16–18], Argyris and Kelsey [19], Kron [20], and Turner et al. [10] should be noted. A thorough study of direct stiffness and related methods was recently conducted by Samuelsson and Zienkiewicz [21].

It will be found that most classical mathematical approximation procedures as well as the various direct approximations used in engineering fall into this category. It is thus difficult to determine the origins of the finite element method and the precise moment of its invention.

Table 1.1 shows the process of evolution which led to the present-day concepts of finite element analysis. A historical development of the subject of finite element methods has been presented by the first author in Refs. [34–36]. Chapters 3 and 4

**Table 1.1** History of Approximate Methods [1, 2, 8–10, 19, 22–33]



will give, in more detail, the mathematical basis which emerged from these classical ideas [1,22–27,29,30,32].

## 1.2 The structural element and the structural system

To introduce the reader to the general concept of discrete systems we shall first consider a structural engineering example with linear elastic behavior.

Figure 1.1 represents a two-dimensional structure assembled from individual components and interconnected at the nodes numbered 1 to 6. The joints at the nodes in this case are pinned so that moments cannot be transmitted.

As a starting point it will be assumed that by separate calculation, or for that matter from the results of an experiment, the characteristics of each element are precisely known. Thus, if a typical element labeled (1) and associated with nodes 1, 2, and 3 is examined, the forces acting at the nodes are uniquely defined by the displacements of these nodes, the distributed loading acting on element ($p$), and its initial strain. The last may be due to temperature, shrinkage, or simply an initial "lack of fit." The forces and the corresponding displacements are defined by appropriate components ($U$, $V$) and ($u$, $v$) in a common coordinate system ($x$, $y$).

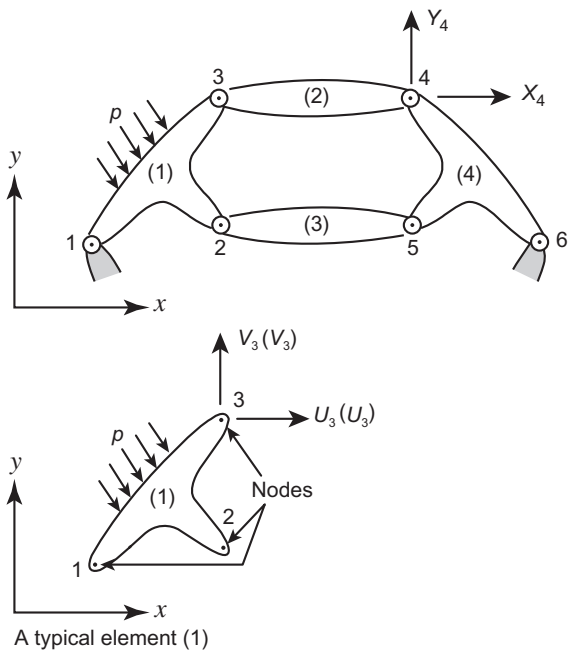



**FIGURE 1.1**
A typical structure built up from interconnected elements.

Listing the forces acting on all the nodes (three in the case illustrated) of element (1) as a matrix[1] we have

$$\mathbf{r}^1 = \begin{Bmatrix} \mathbf{r}_1^1 \\ \mathbf{r}_2^1 \\ \mathbf{r}_3^1 \end{Bmatrix}; \qquad \mathbf{r}_1^1 = \begin{Bmatrix} U_1 \\ V_1 \end{Bmatrix} \quad \text{etc.} \tag{1.1}$$

and for the corresponding nodal displacements

$$\mathbf{u}^1 = \begin{Bmatrix} \mathbf{u}_1^1 \\ \mathbf{u}_2^1 \\ \mathbf{u}_3^1 \end{Bmatrix}; \qquad \mathbf{u}_1^1 = \begin{Bmatrix} u_1 \\ v_1 \end{Bmatrix} \quad \text{etc.} \tag{1.2}$$

Assuming linear elastic behavior of the element, the characteristic relationship will always be of the form

$$\mathbf{r}^1 = \mathbf{K}^1\mathbf{u}^1 - \mathbf{f}^1 \tag{1.3}$$

in which $\mathbf{f}^1$ represents the nodal forces required to balance any concentrated or distributed loads acting on the element. The first of the terms represents the forces induced by displacement of the nodes. The matrix $\mathbf{K}^e$ is known as the *stiffness matrix* for element $(e)$.

Equation (1.3) is illustrated by an example of an element with three nodes with the interconnection points capable of transmitting only two components of force. Clearly, the same arguments and definitions will apply generally. Element (2) of the hypothetical structure will possess only two points of interconnection; others may have quite a large number of such points. Quite generally, therefore,

$$\mathbf{r}^e = \begin{Bmatrix} \mathbf{r}_1^e \\ \mathbf{r}_2^e \\ \vdots \\ \mathbf{r}_m^e \end{Bmatrix} \quad \text{and} \quad \mathbf{u}^e = \begin{Bmatrix} \mathbf{u}_1^e \\ \mathbf{u}_2^e \\ \vdots \\ \mathbf{u}_m^e \end{Bmatrix} \tag{1.4}$$

with each $\mathbf{r}_a^e$ and $\mathbf{u}_a^e$ possessing the same number of components or *degrees of freedom*.

The stiffness matrices of the element will clearly always be square and of the form

$$\mathbf{K}^e = \begin{bmatrix} \mathbf{K}_{11}^e & \mathbf{K}_{12}^e & \cdots & \mathbf{K}_{1m}^e \\ \mathbf{K}_{21}^e & \ddots & & \vdots \\ \vdots & \vdots & & \vdots \\ \mathbf{K}_{m1}^e & \cdots & \cdots & \mathbf{K}_{mm}^e \end{bmatrix} \tag{1.5}$$

[1] A limited knowledge of matrix algebra will be assumed throughout this book. This is necessary for reasonable conciseness and forms a convenient bookkeeping form. For readers not familiar with the subject a brief appendix (Appendix A) is included in which sufficient principles of matrix algebra are given to follow the development intelligently. Matrices and vectors will be distinguished by bold print throughout.

in which $\mathbf{K}^e_{11}$, $\mathbf{K}^e_{12}$, etc., are submatrices which are again square and of the size $l \times l$, where $l$ is the number of force and displacement components to be considered at each node. The element properties were assumed to follow a simple linear relationship. In principle, similar relationships could be established for nonlinear materials, but discussion of such problems is not covered in this volume. Interested readers are referred to Ref. [37] for basic information on solid and structural mechanics applications and to Ref. [38] for applications to fluid dynamics problems. In most cases considered in this volume the element matrices $\mathbf{K}^e$ will be symmetric, that is

$$\mathbf{K}^e = (\mathbf{K}^e)^{\mathrm{T}} \tag{1.6}$$

where $(\cdot)^{\mathrm{T}}$ denotes transpose of a matrix (see Appendix A).

## 1.3 Assembly and analysis of a structure

Consider again the hypothetical structure of Fig. 1.1. To obtain a complete solution the two conditions of

**(a)** displacement compatibility and
**(b)** equilibrium

have to be satisfied throughout.

Any system of nodal displacements $\mathbf{u}$,

$$\mathbf{u} = \begin{Bmatrix} \mathbf{u}^1 \\ \vdots \\ \mathbf{u}^n \end{Bmatrix} \tag{1.7}$$

listed now for the whole structure in which all the elements participate, automatically satisfies the first condition.

As the conditions of overall equilibrium have already been satisfied *within* an element, all that is necessary is to establish equilibrium conditions at the nodes (or assembly points) of the structure. The resulting equations will contain the displacements as unknowns, and once these have been solved the structural problem is determined. The internal forces in elements, or the stresses, can easily be found by using the characteristics established *a priori* for each element.

If now the equilibrium conditions of a typical node, $a$, are to be established, the sum of the component forces contributed by the elements meeting at the node is simply accumulated. Thus, considering *all* the force components we have

$$\sum_{e=1}^{n_e} \mathbf{r}^e_a = \mathbf{r}^1_a + \mathbf{r}^2_a + \cdots = \mathbf{0} \tag{1.8}$$

in which $\mathbf{r}^1_a$ is the force contributed to node $a$ by element (1), $\mathbf{r}^2_a$ by element (2), etc. Clearly, only the elements which include point $a$ will contribute nonzero forces, but for conciseness in notation all the elements are included in the summation.

Substituting the forces contributing to node $a$ from definition (1.3) and noting that nodal variables $\mathbf{u}_a$ are common (thus omitting the superscript $e$), we have

$$\left(\sum_{e=1}^{n_e} \mathbf{K}_{a1}^e\right)\mathbf{u}_1 + \left(\sum_{e=1}^{n_e} \mathbf{K}_{a2}^e\right)\mathbf{u}_2 + \cdots - \sum_{e=1}^{n_e} \mathbf{f}_i^e = \mathbf{0} \tag{1.9}$$

The summation again only concerns the elements which contribute to node $a$. If all such equations are assembled we have simply

$$\mathbf{K}\mathbf{u} - \mathbf{f} = \mathbf{0} \tag{1.10}$$

in which the submatrices are

$$\mathbf{K}_{ab} = \sum_{e=1}^{n_e} \mathbf{K}_{ab}^e \quad \text{and} \quad \mathbf{f}_a = \sum_{e=1}^{n_e} \mathbf{f}_a^e \tag{1.11}$$

with summations including all elements. This simple rule for assembly is very convenient because as soon as a coefficient for a particular element is found it can be put immediately into the appropriate "location" specified in the computer. *This general assembly process can be found to be the common and fundamental feature of all finite element calculations and should be well understood by the reader.*

If different types of structural elements are used and are to be coupled it must be remembered that at any given node the rules of matrix summation permit this to be done only if these are of identical size. The individual submatrices to be added have therefore to be built up of the same number of individual components of force or displacement.

## 1.4 The boundary conditions

The system of equations resulting from Eq. (1.10) can be solved once the prescribed support displacements have been substituted. In the example of Fig. 1.1, where both components of displacement of nodes 1 and 6 are zero, this will mean the substitution of

$$\mathbf{u}_1 = \mathbf{u}_6 = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}$$

which is equivalent to reducing the number of equilibrium equations (in this instance 12) by deleting the first and last pairs and thus reducing the total number of unknown displacement components to eight. It is, nevertheless, often conceptually convenient to assemble the equation according to relation (1.10) so as to include all the nodes.

Clearly, in a "static" or steady-state problem without substitution of a minimum number of prescribed displacements to prevent rigid body movements of the structure, it is impossible to solve this system, because the displacements cannot be uniquely determined by the forces in such a situation. This physically obvious fact will be interpreted mathematically as the matrix $\mathbf{K}$ being singular, i.e., not possessing an inverse. The prescription of appropriate displacements after the assembly stage will

permit a unique solution to be obtained by deleting appropriate rows and columns of the various matrices.

If all the equations of a system are assembled, their form is

$$\begin{aligned}\mathbf{K}_{11}\mathbf{u}_1 + \mathbf{K}_{12}\mathbf{u}_2 + \mathbf{K}_{13}\mathbf{u}_3 + \cdots - \mathbf{f}_1 &= \mathbf{0}\\ \mathbf{K}_{21}\mathbf{u}_1 + \mathbf{K}_{22}\mathbf{u}_2 + \mathbf{K}_{23}\mathbf{u}_3 + \cdots - \mathbf{f}_2 &= \mathbf{0}\\ \mathbf{K}_{31}\mathbf{u}_1 + \mathbf{K}_{32}\mathbf{u}_2 + \mathbf{K}_{33}\mathbf{u}_3 + \cdots - \mathbf{f}_3 &= \mathbf{0}\\ \text{etc.}\end{aligned} \tag{1.12}$$

and it will be noted that if any displacement, such as $\mathbf{u}_1 = \bar{\mathbf{u}}_1$, is prescribed then the total "force" $\mathbf{f}_1$ cannot be simultaneously specified and remains unknown until all values for the remaining $\mathbf{u}_i$ are determined. The first equation can be *deleted* and substitution of known values $\bar{\mathbf{u}}_1$ made in the remaining equations to give

$$\begin{aligned}\mathbf{K}_{22}\mathbf{u}_2 + \mathbf{K}_{23}\mathbf{u}_3 + \cdots (\mathbf{K}_{21}\bar{\mathbf{u}}_1 - \mathbf{f}_2) &= \mathbf{0}\\ \mathbf{K}_{32}\mathbf{u}_2 + \mathbf{K}_{33}\mathbf{u}_3 + \cdots (\mathbf{K}_{31}\bar{\mathbf{u}}_1 - \mathbf{f}_3) &= \mathbf{0}\\ \text{etc.}\end{aligned} \tag{1.13}$$

When all the boundary conditions are inserted the equations of the system can be solved for the unknown nodal displacements and the internal forces in each element obtained.

## 1.5 Electrical and fluid networks

Identical principles of deriving element characteristics and of assembly will be found in many nonstructural fields. Consider, for instance, the assembly of electrical resistances ($r^e$) shown in Fig. 1.2.

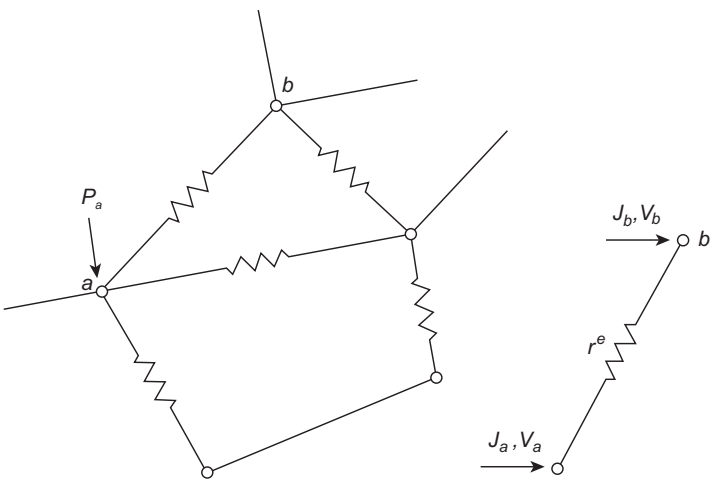



**FIGURE 1.2**
Network of electrical resistances.

If a typical resistance element, *ab*, is isolated from the system we can write, by Ohm's law, the relation between the currents ($J$) *entering* the element at the ends and the end voltages ($V$) as

$$J_a^e = \frac{1}{r^e}(V_a - V_b) \quad \text{and} \quad J_b^e = \frac{1}{r^e}(V_b - V_a) \tag{1.14}$$

or in matrix form

$$\begin{Bmatrix} J_a^e \\ J_b^e \end{Bmatrix} = \frac{1}{r^e}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}\begin{Bmatrix} V_a \\ V_b \end{Bmatrix}$$

which in our standard form is simply

$$\mathbf{J}^e = \mathbf{K}^e\mathbf{V}^e \tag{1.15}$$

This form clearly corresponds to the stiffness relationship (1.3); indeed if an external current were supplied along the length of the element "force" terms could also be found.

To assemble the whole network the continuity of the voltage ($V$) at the nodes is assumed and a current balance imposed there. With no external input of current at node $a$ we must have, with complete analogy to Eq. (1.9),

$$\sum_{b=1}^{n}\sum_{e=1}^{n_e} K_{ab}^e V_b = 0 \tag{1.16}$$

where the second summation is over all "elements," and once again for all the nodes

$$\mathbf{KV} = \mathbf{0} \tag{1.17}$$

in which

$$K_{ab} = \sum_{e=1}^{n_e} K_{ab}^e$$

Matrix notation in the latter has been dropped since the quantities such as voltage and current, and hence also the coefficients of the "stiffness" matrix, are scalars.

If the resistances were replaced by fluid-carrying pipes in which a laminar regime pertained, an identical formulation would once again result, with $V$ standing for the hydraulic head and $J$ for the flow.

For pipe networks that are usually encountered, however, the linear laws are in general not valid and nonlinear equations must be solved.

## 1.6 The general pattern

An example will be considered to consolidate the concepts discussed in this chapter. This is shown in Fig. 1.3a where five discrete elements are interconnected. These may be of structural, electrical, or any other linear type. In the solution:

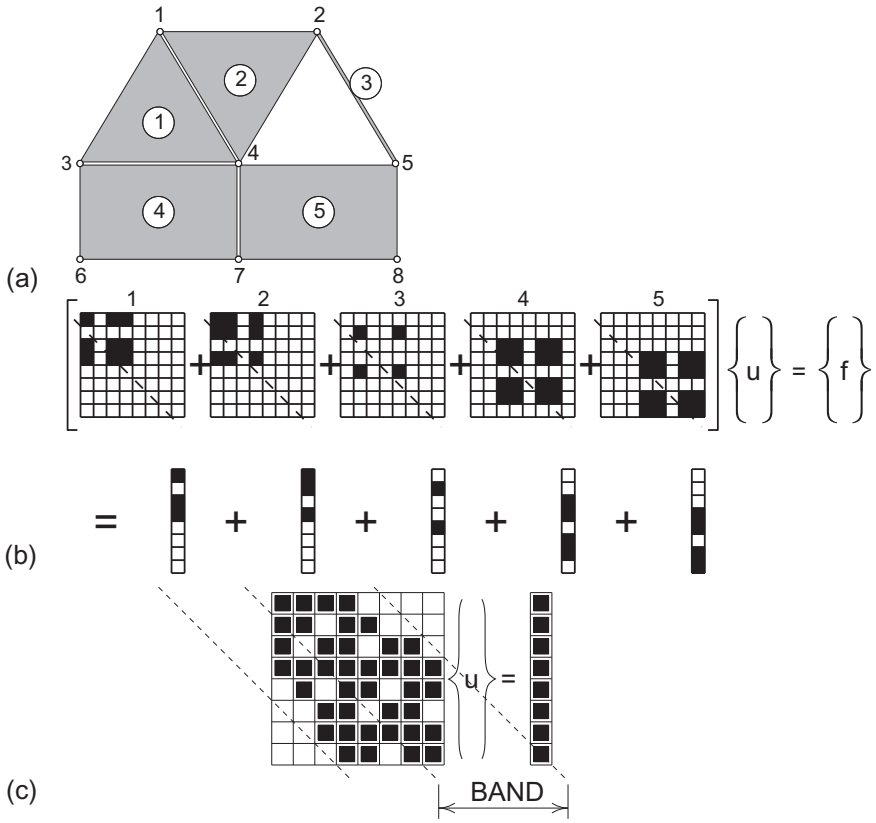



**FIGURE 1.3**
The general pattern: (a) elements, (b) element matrices, and (c) global matrices.

*The first step* is the determination of element properties from the geometric material and loading data. For each element the "stiffness matrix" as well as the corresponding "nodal loads" are found in the form of Eq. (1.3). Each element shown in Fig. 1.3a has its own identifying number and specified nodal connection. For example:

| | | | | | | |
|---|---|---|---|---|---|---|
| element | 1 | connection | 1 | 3 | 4 | |
| | 2 | | 1 | 4 | 2 | |
| | 3 | | 2 | 5 | | |
| | 4 | | 3 | 6 | 7 | 4 |
| | 5 | | 4 | 7 | 8 | 5 |

Assuming that properties are found in global coordinates we can enter each "stiffness" or "force" component in its position of the global matrix as shown in Fig. 1.3b. Each shaded square represents a single coefficient or a submatrix of type $\mathbf{K}_{ab}$ if more than one quantity is being considered at the nodes. Here the separate contribution of each element is shown and the reader can verify the

position of the coefficients. Note that the various types of "elements" considered here present no difficulty in specification. (All "forces," including nodal ones, are here associated with elements for simplicity.)

*The second step* is the assembly of the final equations of the type given by Eq. (1.10). This is accomplished according to the rule of Eq. (1.11) by *simple addition* of all numbers in the appropriate space of the global matrix. The result is shown in Fig. 1.3c where the nonzero coefficients are indicated by shading.

If the matrices are symmetric only the half above (or below) the diagonal shown needs, in fact, to be found.

All the nonzero coefficients are confined within a *band* or *profile* which can be calculated *a priori* for the nodal connections. Thus in computer programs only the storage of the elements within the profile (or sparse structure) is necessary, as shown in Fig. 1.3c. Indeed, if $\mathbf{K}$ is symmetric only the upper (or lower) half need be stored.

*The third step* is the insertion of prescribed boundary conditions into the final assembled matrix, as discussed in Section 1.4. This is followed by the final step.

*The final step* solves the resulting equation system. Here many different methods can be employed, some of which are summarized in Appendix D. The general subject of equation solving, though extremely important, is in general beyond the scope of this book.

The final step discussed above can be followed by substitution to obtain stresses, currents, or other desired *output* quantities. All operations involved in structural or other network analysis are thus of an extremely simple and repetitive kind. We can now define *the standard discrete system* as one in which such conditions prevail.

## 1.7 The standard discrete system

In the *standard discrete system*, whether it is structural or of any other kind, we find that:

1. A set of discrete parameters, say $\mathbf{u}_a$, can be identified which describes simultaneously the behavior of each element, $e$, and of the whole system. We shall call these the *system parameters*.
2. For each element a set of quantities $\mathbf{r}_a^e$ can be computed in terms of the system parameters $\mathbf{u}_a$. The general function relationship can be nonlinear, for example

$$\mathbf{r}_a^e = \mathbf{r}_a^e(\mathbf{u}) \tag{1.18}$$

but in most of the cases considered in this volume a linear form exists giving

$$\mathbf{r}_a^e = \mathbf{K}_{a1}^e \mathbf{u}_1 + \mathbf{K}_{a2}^e \mathbf{u}_2 + \cdots - \mathbf{f}_a^e \tag{1.19}$$

**3.** The final *system equations* are obtained by a simple addition

$$\mathbf{r}_a = \sum_{e=1}^{n_e} \mathbf{r}_a^e = \mathbf{0} \tag{1.20}$$

where $\mathbf{r}_a$ are system quantities. In the linear case this results in a system of equations

$$\mathbf{K}\mathbf{u} - \mathbf{f} = \mathbf{0} \tag{1.21}$$

such that

$$\mathbf{K}_{ab} = \sum_{e=1}^{n_e} \mathbf{K}_{ab}^e \quad \text{and} \quad \mathbf{f}_a = \sum_{e=1}^{n_e} \mathbf{f}_a^e \tag{1.22}$$

from which the solution for the system variables $\mathbf{u}$ can be found after imposing necessary boundary conditions.

The reader will observe that this definition includes the structural, hydraulic, and electrical examples already discussed. However, it is broader. In general neither linearity nor symmetry of matrices need exist—although in many problems this will arise naturally. Further, the narrowness of interconnections existing in usual elements is not essential.

While much further detail could be discussed we feel that the general exposure given here should suffice for further study of the methods included in this book.

Only one further matter relating to the change of discrete parameters needs to be mentioned here. The process of so-called transformation of coordinates is vital in many contexts and must be fully understood.

## 1.8 Transformation of coordinates

It is often convenient to establish the characteristics of an individual element in a coordinate system which is different from that in which the external forces and displacements of the assembled system will be measured. A different coordinate system may, in fact, be used for every element, to ease the computation. It is a simple matter to transform the coordinates of the displacement and force components of Eq. (1.3) to any other coordinate system. Clearly, it is necessary to do so before an assembly of the global system equations can be attempted.

Let the local coordinate system in which the element properties have been evaluated be denoted by a prime suffix and the common coordinate system necessary for assembly have no embellishment. The displacement components can be transformed by a suitable matrix of direction cosines $\mathbf{L}$ as

$$\mathbf{u}' = \mathbf{L}\mathbf{u} \tag{1.23}$$

As the corresponding force components must perform the same amount of work in either system

$$\mathbf{r}^{\mathrm{T}}\mathbf{u} = \mathbf{r}'^{\mathrm{T}}\mathbf{u}' \tag{1.24}$$

On inserting (1.23) we have

$$\mathbf{r}^{\mathrm{T}}\mathbf{u} = \mathbf{r}'^{\mathrm{T}}\mathbf{L}\mathbf{u}$$

or

$$\mathbf{r} = \mathbf{L}^{\mathrm{T}}\mathbf{r}' \tag{1.25}$$

The set of transformations given by (1.23) and (1.25) is called *contravariant* or *dual*.

To transform "stiffnesses" and "loads" which may be available in local coordinates to global ones note that if we write

$$\mathbf{r}' = \mathbf{K}'\mathbf{u}' - \mathbf{f}' \tag{1.26}$$

then by (1.25), (1.26), and (1.23)

$$\mathbf{r} = \mathbf{L}^{\mathrm{T}}\mathbf{K}'\mathbf{L}\mathbf{u} - \mathbf{L}^{\mathrm{T}}\mathbf{f}' = \mathbf{K}\,\mathbf{u} - \mathbf{f}$$

or in global coordinates

$$\mathbf{K} = \mathbf{L}^{\mathrm{T}}\mathbf{K}'\mathbf{L} \quad \text{and} \quad \mathbf{f} = \mathbf{L}^{\mathrm{T}}\mathbf{f}' \tag{1.27}$$

In many complex problems an external constraint of some kind may be imagined, enforcing the requirement (1.23) with the number of degrees of freedom of $\mathbf{u}$ and $\mathbf{u}'$ being quite different. Even in such instances the relations (1.24) and (1.25) continue to be valid.

An alternative and more general argument can be applied to many other situations of discrete analysis. We wish to replace a set of parameters $\mathbf{u}$ in which the system equations have been written by another set related to it by a transformation matrix $\mathbf{T}$ as

$$\mathbf{u} = \mathbf{T}\mathbf{v} \tag{1.28}$$

In the linear case the system equations are of the form

$$\mathbf{K}\mathbf{u} = \mathbf{f} \tag{1.29}$$

and on the substitution we have

$$\mathbf{K}\mathbf{T}\mathbf{v} = \mathbf{f} \tag{1.30}$$

The new system can be premultiplied simply by $\mathbf{T}^{\mathrm{T}}$, yielding

$$(\mathbf{T}^{\mathrm{T}}\mathbf{K}\mathbf{T})\mathbf{v} = \mathbf{T}^{\mathrm{T}}\mathbf{f} \tag{1.31}$$

which will preserve the symmetry of equations if the matrix $\mathbf{K}$ is symmetric. However, occasionally the matrix $\mathbf{T}$ is not square and expression (1.28) represents in fact an *approximation* or a *projection* in which a larger number of parameters $\mathbf{u}$ is *constrained* to a smaller number. Clearly the system of equations (1.30) gives more equations than are necessary for a solution of the reduced set of parameters $\mathbf{v}$, and the final expression (1.31) presents a reduced system which in some sense approximates the original one.

We have thus introduced the basic idea of approximation, which will be the subject of subsequent chapters where infinite sets of quantities are reduced to finite sets.

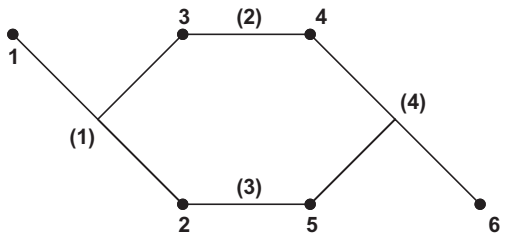



**FIGURE 1.4**
Fluid network for Problem 1.1.

## 1.9 Problems

**1.1** A simple fluid network to transport water is shown in Fig. 1.4. Each "element" of the network is modeled in terms of the flow, $\mathbf{J}$, and head, $\mathbf{V}$, which are approximated by the linear relation

$$\mathbf{J}^e = -\mathbf{K}^e\mathbf{V}^e$$

where $\mathbf{K}^e$ is the coefficient array for element $(e)$. The individual terms in the flow vector denote the total amount of flow entering $(+)$ or leaving $(-)$ each end point. The properties of the elements are given by

$$\mathbf{K}^e = c^e \begin{bmatrix} 3 & -2 & -1 \\ -2 & 4 & -2 \\ -1 & -2 & 3 \end{bmatrix}$$

for elements (1) and (4), and for elements (2) and (3) by

$$\mathbf{K}^e = c^e \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

where $c^e$ is an element-related parameter. The system is operating with a known head of 100 m at node 1 and 30 m at node 6. At node 2, 30 $m^3h^{-1}$ of water per hour are being used and at node 4, 10 $m^3$ per hour.

**(a)** For all $c^e = 1$, assemble the total matrix from the individual elements to give

$$\mathbf{J} = \mathbf{K}\,\mathbf{V}$$

N.B. $\mathbf{J}$ contains entries for the specified usage and connection points.

**(b)** Impose boundary conditions by modifying $\mathbf{J}$ and $\mathbf{K}$ such that the known heads at nodes 1 and 6 are recovered.

**(c)** Solve the equations for the heads at nodes 2 to 5. (The result at node 4 should be $V_4 = 30.8133$ m.)

**(d)** Determine the flow entering and leaving each element.

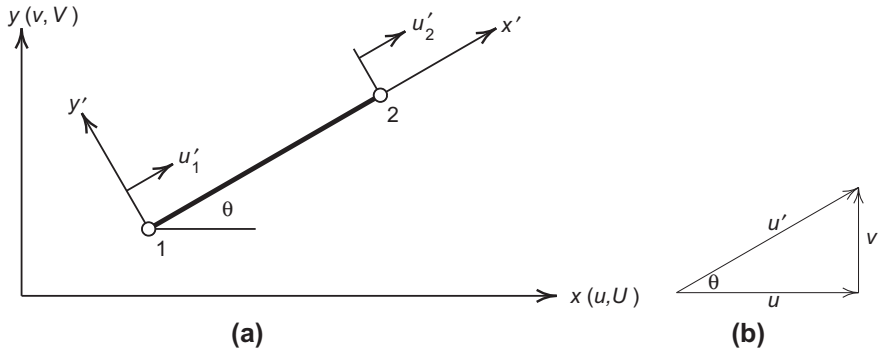



**FIGURE 1.5**
Truss member for Problem 1.2: (a) truss member description and (b) displacements.

**1.2** A plane truss may be described as a standard discrete problem by expressing the characteristics for each member in terms of end displacements and forces. The behavior of the elastic member is shown in Fig. 1.5 with modulus $E$, cross-section $A$, and length $L$ given by

$$\mathbf{r}' = \mathbf{K}'_e \, \mathbf{u}'$$

where

$$\mathbf{r}' = \begin{Bmatrix} U'_1 \\ U'_2 \end{Bmatrix}, \quad \mathbf{u}' = \begin{Bmatrix} u'_1 \\ u'_2 \end{Bmatrix} \quad \text{and} \quad \mathbf{K}'_e = \frac{EA}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

To obtain the final assembled matrices for a standard discrete problem it is necessary to transform the behavior to a global frame using Eqs. (1.25) and (1.27) where

$$\mathbf{L} = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ 0 & 0 & \cos\theta & \sin\theta \end{bmatrix}, \quad \mathbf{r} = \begin{Bmatrix} U_1 \\ V_1 \\ U_2 \\ V_2 \end{Bmatrix} \quad \text{and} \quad \mathbf{u} = \begin{Bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \end{Bmatrix}$$

**(a)** Compute relations for $\mathbf{r}$ and $\mathbf{K}$ in terms of $\mathbf{L}$, $\mathbf{r}'$, and $\mathbf{K}'_e$.
**(b)** If the numbering for the end nodes is reversed what is the final form for $\mathbf{K}$ compared to that given in (a)? Verify your answer when $\theta = 30°$.

**1.3** A plane truss has nodes numbered as shown in Fig. 1.6a.

**(a)** Use the procedure shown in Fig. 1.3 to define the nonzero structure of the coefficient matrix $\mathbf{K}$. Compute the maximum bandwidth.
**(b)** Determine the nonzero structure of $\mathbf{K}$ for the numbering of nodes shown in Fig. 1.6b. Compute the maximum bandwidth.

Which order produces the smallest band?

**1.4** Write a small computer program (e.g., using MATLAB [39] or GNU Octave [40]) to solve the truss problem shown in Fig. 1.6b. Let the total span of the truss

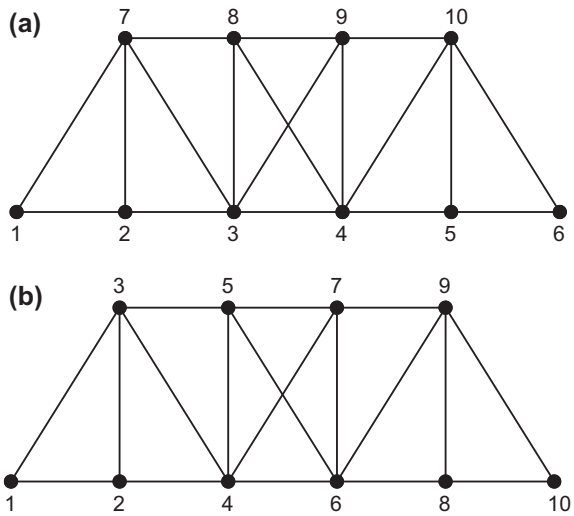



**FIGURE 1.6**
Truss for Problems 1.3 and 1.4.

be 2.5 m and the height 0.8 m and use steel as the property for each member with $E = 200$ GPa and $A = 0.001$ m$^2$. Restrain node 1 in both the $u$ and $v$ directions and the right bottom node in the $v$ direction only. Apply a vertical load of 100 N at the position of node 6 shown in Fig. 1.6b. Determine the maximum vertical displacement at any node. Plot the undeformed and deformed position of the truss (increase the magnitude of displacements to make the shape visible on the plot). You can verify your result using the program *FEAPpv* available at the second author's website (see Chapter 18).

**1.5** An axially loaded elastic bar has a variable cross-section and lengths as shown in Fig. 1.7a. The problem is converted into a standard discrete system by considering each prismatic section as a separate member. The array for each member segment is given as

$$\mathbf{r}^e = \mathbf{K}^e \mathbf{u}^e$$

where

$$\mathbf{K}^e = \frac{EA_e}{h}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}; \quad \mathbf{r}^e = \begin{Bmatrix} r_e^e \\ r_{e+1}^e \end{Bmatrix} \quad \text{and} \quad \mathbf{u}^e = \begin{Bmatrix} u_e \\ u_{e+1} \end{Bmatrix}$$

Equilibrium for the standard discrete problem at joint $e$ is obtained by combining results from segment $e-1$ and $e$ as

$$r_e^{e-1} + r_e^e + U_e = 0$$

where $U_e$ is any external force applied to a joint. Boundary conditions are applied for any joint at which the value of $u_e$ is known *a priori*.

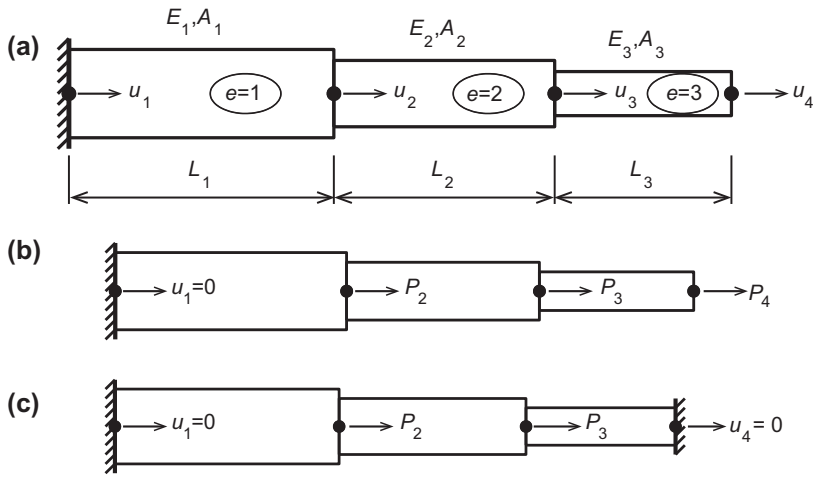



**FIGURE 1.7**

Elastic bars for Problems 1.5 and 1.6: (a) bar geometry, (b) Problem 1.5, and (c) Problem 1.6.

Solve the problem shown in Fig. 1.7b for the joint displacements using the data $E_1 = E_2 = E_3 = 200$ GPa, $A_1 = 25$ cm$^2$, $A_2 = 20$ cm$^2$, $A_3 = 12$ cm$^2$, $L_1 = 37.5$ cm, $L_2 = 25.0$ cm, $L_3 = 12.5$ cm, $P_2 = 10$ kN, $P_3 = -3.5$ kN, and $P_4 = 6$ kN.

**1.6** Solve Problem 1.5 for the boundary conditions and loading shown in Fig. 1.7c. Let $E_1 = E_2 = E_3 = 200$ GPa, $A_1 = 30$ cm$^2$, $A_2 = 20$ cm$^2$, $A_3 = 10$ cm$^2$, $L_1 = 37.5$ cm, $L_2 = 30.0$ cm, $L_3 = 25.0$ cm, $P_2 = -10$ kN, and $P_3 = 3.5$ kN.

**1.7** A tapered bar is loaded by an end load $P$ and a uniform loading $b$ as shown in Fig. 1.8a. The area varies as $A(x) = A\,x/L$ when the origin of coordinates is located as shown in the figure.

The problem is converted into a standard discrete system by dividing it into equal length segments of constant area as shown in Fig. 1.8b. The array for each

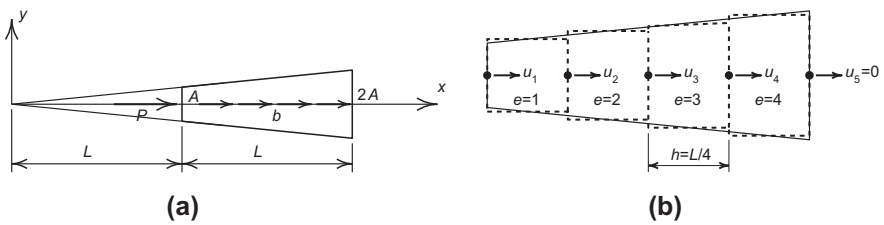



**FIGURE 1.8**

Tapered bar for Problem 1.7: (a) tapered bar geometry and (b) approximation by 4 segments.

segment is determined from

$$\mathbf{r}^e = \mathbf{K}^e\mathbf{u}^e + \mathbf{f}^e$$

where $\mathbf{K}^e$ and $\mathbf{u}^e$ are defined in Problem 1.5 and

$$\mathbf{f}^e = \frac{1}{2}bh\begin{Bmatrix}1\\1\end{Bmatrix}$$

For the properties $L = 100$ cm, $A = 2$ cm$^2$, $E = 10^4$ kN/cm$^2$, $P = 2$ kN, $b = -0.25$ kN/cm, and $u(2L) = 0$, the displacement from the solution of the differential equation is $u(L) = -0.03142513$ cm.
Write a small computer program (e.g., using MATLAB [39] or GNU Octave [40]) that solves the problem for the case where $e = 1, 2, 4, 8, \ldots$ segments. Continue the solution until the absolute error in the tip displacement is $< 10^{-5}$ cm (let error be $E = |u(L) - u_1|$ where $u_1$ is the numerical solution at the end).

## References

[1] L.F. Richardson, The approximate arithmetical solution by finite differences of physical problems, Trans. R. Soc. (Lond.) A 210 (1910) 307–357.
[2] R.V. Southwell, Relaxation Methods in Theoretical Physics, Clarendon Press, Oxford, 1946.
[3] D.N. de G. Allen, Relaxation Methods, McGraw-Hill, London, 1955.
[4] S. Crandall, Engineering Analysis, McGraw-Hill, New York, 1956.
[5] B.A. Finlayson, The Method of Weighted Residuals and Variational Principles, Academic Press, New York, 1972.
[6] K. Washizu, Variational Methods in Elasticity and Plasticity, third ed., Pergamon Press, New York, 1982.
[7] D. McHenry, A lattice analogy for the solution of plane stress problems, J. Inst. Civ. Eng. 21 (1943) 59–82.
[8] A. Hrenikoff, Solution of problems in elasticity by the framework method, J. Appl. Mech. ASME A 8 (1941) 169–175.
[9] N.M. Newmark, Numerical methods of analysis in bars, plates and elastic bodies, in: L.E. Grinter (Ed.), Numerical Methods in Analysis in Engineering, Macmillan, New York, 1949.
[10] M.J. Turner, R.W. Clough, H.C. Martin, L.J. Topp, Stiffness and deflection analysis of complex structures, J. Aeronaut. Sci. 23 (1956) 805–823.
[11] R.W. Clough, The finite element method in plane stress analysis, in: Proceedings of the Second ASCE Conference on Electronic Computation, Pittsburgh, PA, September, 1960.
[12] R.W. Clough, Early history of the finite element method from the view point of a pioneer, Int. J. Numer. Methods Eng. 60 (2004) 283–287.
[13] R.F. Clebsch, Théorie de l'elasticité des corps solides, Dunod, Paris, 1883.

[14] R.V. Southwell, Stress calculation in frame works by the method of systematic relaxation of constraints, Proc. R. Soc. Lond. A 151 (1935) 56–95 (Part I and II).
[15] Hardy Cross, Continuous Frames of Reinforced Concrete, John Wiley & Sons, New York, 1932.
[16] W.J. Duncan, A.R. Collar, A method for the solution of oscillation problems by matrices, Phil. Mag. 17 (1934) 865 (Series 7).
[17] W.J. Duncan, A.R. Collar, Matrices applied to the motions of damped systems, Phil. Mag. 19 (1935) 197 (Series 7).
[18] R.R. Frazer, W.J. Duncan, A.R. Collar, Elementary Matrices, Cambridge University Press, London, 1960.
[19] J.H. Argyris, S. Kelsey, Energy Theorems and Structural Analysis, Butterworths, London, 1960 (Reprinted from a series of articles in Aircraft Eng., 1954–1955).
[20] G. Kron, Equivalent Circuits of Electrical Machinery, John Wiley & Sons, New York, 1951.
[21] A. Samuelsson, O.C. Zienkiewicz, History of the stiffness method, Int. J. Numer. Methods Eng. 67 (2006) 149–157.
[22] H. Liebman, Die angenäherte Ermittlung: harmonishen, functionen und konformer Abbildung, Sitzber. Math. Physik Kl. Bayer Akad. Wiss. München 3 (1918) 65–75.
[23] Lord Rayleigh (J.W. Strutt), On the theory of resonance, Trans. R. Soc. (Lond.) A 161 (1870) 77–118.
[24] W. Ritz, Über eine neue Methode zur Lösung gewisser variationsproblem der mathematischen physik, J. Reine Angew. Math. 135 (1908) 1–61.
[25] C.F. Gauss, Werke, Dietrich, Göttingen, 1863–1929 (See: Theoretishe Astronomie, Bd. VII).
[26] B.G. Galerkin, Series solution of some problems in elastic equilibrium of rods and plates, Vestn. Inzh. Tech. 19 (1915) 897–908.
[27] C.B. Biezeno, J.J. Koch, Over een Nieuwe Methode ter Berekening van Vlokke Platen, Ing. Grav. 38 (1923) 25–36.
[28] D. McHenry, A new aspect of creep in concrete and its application to design, Proc. ASTM 43 (1943) 1064.
[29] R. Courant, Variational methods for the solution of problems of equilibrium and vibration, Bull. Am. Math. Soc. 49 (1943) 1–61.
[30] W. Prager, J.L. Synge, Approximation in elasticity based on the concept of function space, Quart. J. Appl. Math. 5 (1947) 241–269.
[31] O.C. Zienkiewicz, Y.K. Cheung, The finite element method for analysis of elastic isotropic and orthotropic slabs, Proc. Inst. Civ. Eng. 28 (1964) 471–488.
[32] R.S. Varga, Matrix Iterative Analysis, Prentice-Hall, Englewood Cliffs, NJ, 1962.
[33] M.L. Wilkins, Calculation of elastic-plastic flow, in: B. Alder (Ed.), Methods in Computational Physics, vol. 3, Academic Press, New York, 1964, pp. 211–263.
[34] O.C. Zienkiewicz, Origins, milestones and directions of the finite element method, Arch. Comput. Methods Eng. 2 (1995) 1–48.

[35] O.C. Zienkiewicz, Origins, milestones and directions of the finite element method. A personal view, in: P.G. Ciarlet, J.L. Lyons (Eds.), Handbook of Numerical Analysis, vol. IV, North Holland, 1996, pp. 3–65.
[36] O.C. Zienkiewicz, The birth of the finite element method and of computational mechanics, Int. J. Numer. Methods Eng. 60 (2004) 3–10.
[37] O.C. Zienkiewicz, R.L. Taylor, D. Fox, The Finite Element Method for Solid and Structural Mechanics, seventh ed., Elsevier, Oxford, 2013.
[38] O.C. Zienkiewicz, R.L. Taylor, P. Nithiarasu, The Finite Element Method for Fluid Dynamics, seventh ed., Elsevier, Oxford, 2013.
[39] MATLAB, www.mathworks.com, 2012.
[40] GNU Octave, www.gnu.org/software/octave, 2012.

CHAPTER

# Problems in Linear Elasticity and Fields

# 2

## 2.1 Introduction

In this chapter we present the set of equations for the theory of elasticity and theory of fields which are common in many engineering applications.

We shall first deal with the elasticity problem for general three-dimensional applications, as well as its simplification to some special two-dimensional situations. In the elasticity equations presented here we assume that strains are small compared to unity. Also we ignore any possibility of buckling. We assume the reader is familiar with the theory of linear elasticity. However, for completeness we will summarize the basic equations for the different problem classes to be considered. For a more general discussion the reader is referred to standard references on the subject (e.g., see Refs. [1–6]) and for problems that consider large strain effects to the companion volume [7].

In some situations an indicial form of the equations can be advantageous, however in this chapter the basic equations will be formulated in a matrix form that will be used later to construct finite element developments. The indicial form is included in Appendix C for completeness.

## 2.2 Elasticity equations

The basic equations for the theory of elasticity are described in terms of displacements, strains, stresses, boundary conditions, initial conditions, and constitutive relations that relate the behavior between strain and stress. We start by specifying each equation set for a general three-dimensional problem in Cartesian coordinates. However, we will also consider some two-dimensional forms. The two-dimensional problems we consider are of three types:

**(a)** The *plane stress* case. In this problem the only nonzero stresses are those in the plane of the problem and normal to the lamina we have no stresses as shown in Fig. 2.1a.

**(b)** The *plane strain* case. Here all straining normal to the plane considered is prevented. Such a situation may arise in the long prism shown in Fig. 2.1b in which loading does not vary in the direction normal to the plane.

**(c)** The third and final case of two-dimensional analysis is that in which the situation is *axisymmetric*. Here the plane considered is one at constant $\theta$ in a cylindrical

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00002-2

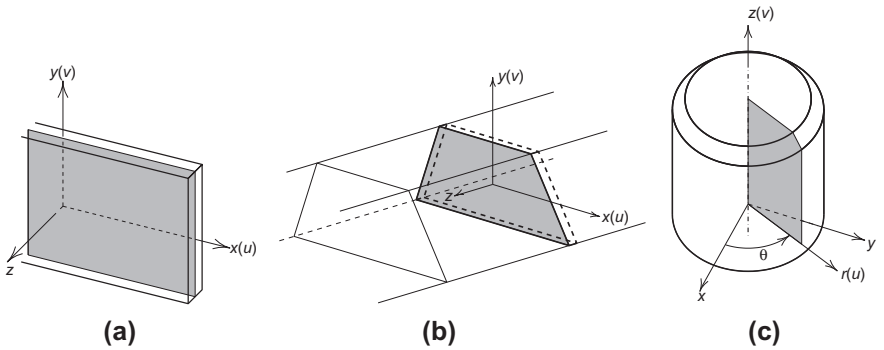



**FIGURE 2.1**
Two-dimensional analysis types for (a) plane stress, (b) plane strain, and (c) axisymmetry.

coordinate system $r$–$z$–$\theta$ (Fig. 2.1c) and all components of displacement, stress, and strain are assumed dependent on $r$ and $z$ only.

### 2.2.1 Displacement function

For the three-dimensional problem the displacement field is given by

$$\mathbf{u}(\mathbf{x},\ t) = \begin{Bmatrix} u(x,\ y,\ z,\ t) \\ v(x,\ y,\ z,\ t) \\ w(x,\ y,\ z,\ t) \end{Bmatrix} \tag{2.1}$$

where positions are denoted by the Cartesian coordinates $x,\ y,\ z$ which we also write in matrix form as

$$\mathbf{x} = \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} \tag{2.2}$$

For two-dimensional plane stress and plane strain cases the displacement field is given by

$$\mathbf{u}(\mathbf{x},\ t) = \begin{Bmatrix} u(x,\ y,\ t) \\ v(x,\ y,\ t) \end{Bmatrix} \tag{2.3}$$

and only the first two components of (2.2) are needed. Finally, in the axisymmetric case we use cylindrical coordinates to define the $r-z$ plane for which

$$\mathbf{x} = \begin{Bmatrix} r \\ z \end{Bmatrix} \tag{2.4}$$

with the displacements given by

$$\mathbf{u}(\mathbf{x},\ t) = \begin{Bmatrix} u(r,\ z,\ t) \\ v(r,\ z,\ t) \end{Bmatrix} \tag{2.5}$$

The only difference in the latter two is the coordinates used: $x$, $y$ for Cartesian coordinates and $r$, $z$ for cylindrical coordinates. The reader may notice that in plane stress problems, changes of thickness occur; however, no explicit displacement assumption is given and the result will be included directly within the strain approximation.

### 2.2.2 Strain matrix

In a three-dimensional problem there are six independent components of strain which we order and denote in matrix form by[1]

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_x & \varepsilon_y & \varepsilon_z & \gamma_{xy} & \gamma_{yz} & \gamma_{zx} \end{bmatrix}^{\mathrm{T}} \tag{2.6}$$

where $[\cdot]^{\mathrm{T}}$ denotes the matrix transpose (see Appendix A). This form is known in the mechanics literature as Voigt notation [8]. It is a way of writing a symmetric second order tensor in terms of a reduced set of components. The strain is a symmetric form where $\gamma_{xy} = \gamma_{yx}$, $\gamma_{yz} = \gamma_{zy}$, and $\gamma_{zx} = \gamma_{xz}$; thus, Voigt notation reduces nine components to six. For the two-dimensional problems considered in this volume the last two components are always zero. Thus, only four components of $\boldsymbol{\varepsilon}$ need be considered as shown in the next section.

#### 2.2.2.1 Strain-displacement matrix

The strains for a problem undergoing small deformations are computed from the displacements and may be expressed in matrix form as

$$\boldsymbol{\varepsilon} = \boldsymbol{\mathcal{S}}\,\mathbf{u} \tag{2.7}$$

where $\boldsymbol{\mathcal{S}}$ is a matrix of differential operators and $\mathbf{u}$ is the displacement field. For the three-dimensional problem the *strain-displacement* relations are given by[2]

$$\boldsymbol{\varepsilon} = \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{Bmatrix} = \begin{bmatrix} \dfrac{\partial}{\partial x} & 0 & 0 \\ 0 & \dfrac{\partial}{\partial y} & 0 \\ 0 & 0 & \dfrac{\partial}{\partial z} \\ \dfrac{\partial}{\partial y} & \dfrac{\partial}{\partial x} & 0 \\ 0 & \dfrac{\partial}{\partial z} & \dfrac{\partial}{\partial y} \\ \dfrac{\partial}{\partial z} & 0 & \dfrac{\partial}{\partial x} \end{bmatrix} \begin{Bmatrix} u \\ v \\ w \end{Bmatrix} \tag{2.8}$$

[1] It is known that strain is a second rank tensor by its transformation properties, however, in this book we will normally represent quantities using matrix (Voigt) notation. The interested reader is encouraged to consult Appendix C for the relations between tensor forms and the matrix quantities.

[2] Note that in matrix form shear strain components are twice that given in tensor form in Appendix C (e.g., $\gamma_{xy} = 2\,\varepsilon_{xy}$).

For convenience in considering all three classes of two-dimensional problems in a unified manner, we include *four* components of strain in $\boldsymbol{\varepsilon}$ and write them as

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial}{\partial x} & 0 \\ 0 & \dfrac{\partial}{\partial y} \\ 0 & 0 \\ \dfrac{\partial}{\partial y} & \dfrac{\partial}{\partial x} \end{bmatrix} \begin{Bmatrix} u \\ v \end{Bmatrix} + \begin{Bmatrix} 0 \\ 0 \\ \varepsilon_z \\ 0 \end{Bmatrix} = \boldsymbol{\mathcal{S}}_p \, \mathbf{u} + \boldsymbol{\varepsilon}_z \tag{2.9}$$

for plane problems (where $\varepsilon_z$ is zero for plane strain but not for plane stress) and

$$\boldsymbol{\varepsilon} = \begin{Bmatrix} \varepsilon_r \\ \varepsilon_z \\ \varepsilon_\theta \\ \gamma_{rz} \end{Bmatrix} = \begin{bmatrix} \dfrac{\partial}{\partial r} & 0 \\ 0 & \dfrac{\partial}{\partial z} \\ \dfrac{1}{r} & 0 \\ \dfrac{\partial}{\partial z} & \dfrac{\partial}{\partial r} \end{bmatrix} \begin{Bmatrix} u \\ v \end{Bmatrix} = \boldsymbol{\mathcal{S}}_a \, \mathbf{u} \tag{2.10}$$

for the axisymmetric case (see Fig. 2.2).

The two-dimensional problem types differ only by the presence of $\boldsymbol{\varepsilon}_z$ in the plane stress problem and the $\varepsilon_\theta$ component in the axisymmetric case.

#### 2.2.2.2 *Volume change and deviatoric strain*

When a small cube deforms it both changes volume and distorts. The change in volume may be expressed in terms of the extensional strains $\varepsilon_x$, $\varepsilon_y$, and $\varepsilon_z$ as

$$\varepsilon_v = \varepsilon_x + \varepsilon_y + \varepsilon_z = \mathbf{m}^{\mathrm{T}} \boldsymbol{\varepsilon} \tag{2.11}$$

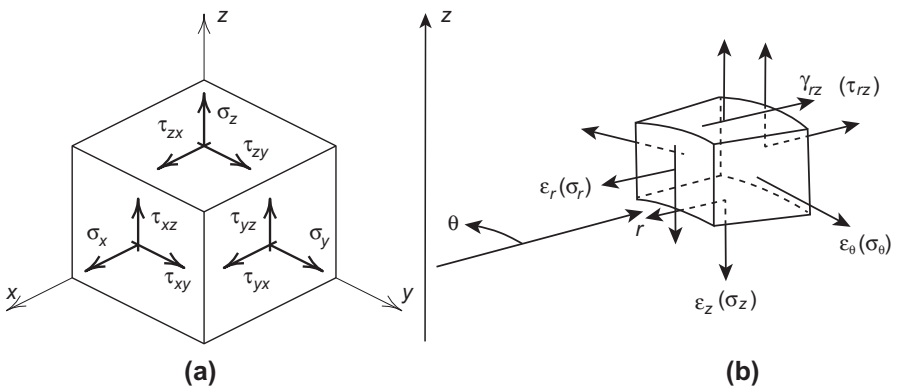



**FIGURE 2.2**
Stresses involved in the analysis of solids: (a) three-dimensional and (b) axisymmetric.

where in three dimensions $\mathbf{m} = [\,1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0\,]^{\mathrm{T}}$ and in two dimensions $\mathbf{m} = [\,1 \quad 1 \quad 1 \quad 0\,]^{\mathrm{T}}$. The distortion is then measured by the *deviatoric strains* which are the difference between the total and mean volumetric strains. The deviatoric components are expressed as

$$\mathbf{e} = \boldsymbol{\varepsilon} - \frac{1}{3}\mathbf{m}\,\varepsilon_v = \left(\mathbf{I} - \frac{1}{3}\mathbf{m}\mathbf{m}^{\mathrm{T}}\right)\boldsymbol{\varepsilon} = \mathbf{I}_{dev}\,\boldsymbol{\varepsilon} \tag{2.12}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{I}_{dev}$ is a *deviatoric projection matrix*. Both are of size 6 for three-dimensional problems and size 4 for the two-dimensional cases.

### 2.2.3 Stress matrix

In a three-dimensional body there are nine components of stress which act on an element of volume as shown in Fig. 2.2a. The components $\sigma_x,\ \sigma_x,\ \sigma_x$ are called normal stresses and $\tau_{xy},\ \tau_{yx},\ \tau_{yz},\ \tau_{zy},\ \tau_{xz},\ \tau_{zx}$ are called shearing stresses. The shearing components are symmetric (by angular momentum or moment equilibrium on the cube) and, thus,

$$\tau_{xy} = \tau_{yx},\ \tau_{yz} = \tau_{zy} \text{ and } \tau_{zx} = \tau_{xz}$$

Thus, similar to strain, the stresses may be written in terms of six components that are ordered and denoted in matrix form by

$$\boldsymbol{\sigma} = [\,\sigma_x \quad \sigma_y \quad \sigma_z \quad \tau_{xy} \quad \tau_{yz} \quad \tau_{zx}\,]^{\mathrm{T}} \tag{2.13}$$

For two-dimensional plane problems we consider only four components of stress and use

$$\boldsymbol{\sigma} = [\,\sigma_x \quad \sigma_y \quad \sigma_z \quad \tau_{xy}\,]^{\mathrm{T}} \tag{2.14}$$

In axisymmetry we define the components of stress as

$$\boldsymbol{\sigma} = [\,\sigma_r \quad \sigma_z \quad \sigma_\theta \quad \tau_{rz}\,]^{\mathrm{T}} \tag{2.15}$$

In plane stress problems we know *a priori* that $\sigma_z = 0$, however, it is convenient to retain the component to allow all two-dimensional problems to be considered in a more uniform manner.

#### 2.2.3.1 Mean stress and deviatoric stress

Similar to strain, we can split the stress into its mean and deviatoric parts. The *mean stress* is defined by

$$p = \frac{1}{3}(\sigma_x + \sigma_y + \sigma_z) = \frac{1}{3}\mathbf{m}^{\mathrm{T}}\boldsymbol{\sigma} \tag{2.16}$$

We shall also refer to $p$ as a pressure. The *deviatoric stress* is then defined as

$$\mathbf{s} = \boldsymbol{\sigma} - \mathbf{m}\,p = \left(\mathbf{I} - \frac{1}{3}\mathbf{m}\mathbf{m}^{\mathrm{T}}\right)\boldsymbol{\sigma} = \mathbf{I}_{dev}\,\boldsymbol{\sigma} \tag{2.17}$$

The split form for stress and strain will be useful when we discuss stress-strain relations and later in Chapter 10 for incompressible problems.

### 2.2.4 Equilibrium equations

The linear momentum or *equilibrium equations* for the three-dimensional behavior of a solid may be written in Cartesian coordinates as

$$\begin{aligned}
\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + b_x &= \rho \frac{\partial^2 u}{\partial t^2} \\
\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + b_y &= \rho \frac{\partial^2 v}{\partial t^2} \\
\frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \sigma_z}{\partial z} + b_z &= \rho \frac{\partial^2 w}{\partial t^2}
\end{aligned} \tag{2.18}$$

in which we recall that by angular momentum (moment equilibrium) the stress is symmetric. Using only the independent components of the symmetric stress given in (2.13), the equilibrium equations in Cartesian coordinates may be written in a matrix Voigt form as

$$\boldsymbol{\mathcal{S}}^{\mathrm{T}} \boldsymbol{\sigma} + \mathbf{b} = \rho \, \ddot{\mathbf{u}} \tag{2.19}$$

where in Cartesian coordinates $\boldsymbol{\mathcal{S}}$ is the same differential operator as that given for strains in (2.8), $\mathbf{b}$ is the vector of body forces given as

$$\mathbf{b} = [\, b_x \quad b_y \quad b_z \,]^{\mathrm{T}} \tag{2.20}$$

$\rho$ is the mass density per unit volume and $\ddot{\mathbf{u}} = \partial^2 \mathbf{u} / \partial t^2$ is the acceleration vector.

#### 2.2.4.1 Plane stress and plane strain problems

In two-dimensional plane problems we omit $\tau_{yz}$, $\tau_{zx}$, and $b_z$. We also note that $\sigma_z$ will not depend on the $z$-coordinate. Thus the stress matrix is given by (2.14) and the remaining two equilibrium equations become

$$\begin{aligned}
\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + b_x &= \rho \frac{\partial^2 u}{\partial t^2} \\
\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} + b_y &= \rho \frac{\partial^2 v}{\partial t^2}
\end{aligned} \tag{2.21}$$

Noting that $\tau_{xy} = \tau_{yx}$, the equilibrium equations may be written in matrix form as

$$\boldsymbol{\mathcal{S}}_p^{\mathrm{T}} \boldsymbol{\sigma} + \mathbf{b} = \rho \, \ddot{\mathbf{u}} \tag{2.22}$$

where $\boldsymbol{\mathcal{S}}_p$ is identical to that given in (2.9); also $\mathbf{b}$ and $\mathbf{u}$ now only have two components.

#### 2.2.4.2 Axisymmetric problems

For axisymmetric problems the stress is replaced by (2.15) and body force by $\mathbf{b} = [\, b_r \quad b_z \,]^{\mathrm{T}}$. Using the stresses on the element of volume shown in Fig. 2.2 and summing

forces in the $r$ and $z$ directions the two equilibrium equations are given by

$$\begin{aligned} \frac{\partial \sigma_r}{\partial r} + \frac{\sigma_r - \sigma_\theta}{r} + \frac{\partial \tau_{zr}}{\partial z} + b_r &= \rho \frac{\partial^2 u}{\partial t^2} \\ \frac{\partial \tau_{rz}}{\partial r} + \frac{\partial \sigma_z}{\partial z} + b_z &= \rho \frac{\partial^2 v}{\partial t^2} \end{aligned} \tag{2.23}$$

By moment equilibrium (angular momentum) we again obtain $\tau_{rz} = \tau_{zr}$. Note that in the axisymmetric problem the differential operator on equilibrium is given by

$$\bar{\boldsymbol{\mathcal{S}}}_a^{\mathrm{T}} = \begin{bmatrix} \left(\dfrac{\partial}{\partial r} + \dfrac{1}{r}\right) & 0 & -\dfrac{1}{r} & \dfrac{\partial}{\partial z} \\ 0 & \dfrac{\partial}{\partial z} & 0 & \dfrac{\partial}{\partial r} \end{bmatrix} \tag{2.24}$$

and is not equal to $\boldsymbol{\mathcal{S}}_a^{\mathrm{T}}$. This difference occurs due to the curvilinear coordinate system. Working with the differential equation form this is a disadvantage, however, for methods used in finite element analysis we will show that the difference disappears and formulations in curvilinear coordinates greatly simplify.

### 2.2.5 Boundary conditions

The strain-displacement and equilibrium equations are valid at each point of the body *domain*, which we denote by $\Omega$. Boundary conditions must be specified for each point on the surface of the solid, which we denote as $\Gamma$. Here we consider only two types of boundary conditions: (a) displacement boundary conditions and (b) traction boundary conditions. We divide the boundary into two parts, which we denote as $\Gamma_u$ for displacement conditions and $\Gamma_t$ for traction conditions. Thus, the total boundary is always the sum of the displacement and traction parts.

Displacement boundary conditions are specified at each point of the boundary $\Gamma_u$ as

$$\mathbf{u} = \bar{\mathbf{u}}(\mathbf{x},\ t) \tag{2.25}$$

where $\bar{\mathbf{u}}$ are known values and $\mathbf{x}$ are points on the boundary.

Traction boundary conditions are specified for each point of the boundary $\Gamma_t$ and are given in terms of stresses by

$$\mathbf{t} = \mathbf{G}^{\mathrm{T}}\boldsymbol{\sigma} = \bar{\mathbf{t}}(\mathbf{x},\ t) \tag{2.26}$$

in which for three-dimensional problems $\mathbf{G}^{\mathrm{T}}$ is the matrix

$$\mathbf{G}^{\mathrm{T}} = \begin{bmatrix} n_x & 0 & 0 & n_y & 0 & n_z \\ 0 & n_y & 0 & n_x & n_z & 0 \\ 0 & 0 & n_z & 0 & n_y & n_x \end{bmatrix} \tag{2.27}$$

where $n_x$, $n_y$, $n_z$ are direction cosines for the *outward pointing normal*, $\mathbf{n}$, to the boundary $\Gamma_t$. The reader should note that the matrices $\mathbf{G}$ and $\boldsymbol{\mathcal{S}}$ have identical nonzero

structure. In two-dimensional plane problems $\mathbf{G}$ reduces to

$$\mathbf{G}_p^{\mathrm{T}} = \begin{bmatrix} n_x & 0 & 0 & n_y \\ 0 & n_y & 0 & n_x \end{bmatrix} \tag{2.28}$$

with $n_x,\ n_y$ the components of an outward pointing unit normal vector of the boundary. In axisymmetry $n_x = n_r,\ n_y = n_z$, and $\mathbf{G}_a$ has the same nonzero form as $\mathbf{G}_p$.

Both boundary condition types are vectors and, at any point on the boundary, it is possible to specify some components by displacement conditions and others by traction ones.

#### 2.2.5.1 *Boundary conditions on inclined coordinates*

It is also possible that the conditions are given with respect to coordinates $x',\ y',\ z'$ which are oriented with respect to the global axes $x,\ y,\ z$ by

$$\mathbf{x}' = \mathbf{T}\,\mathbf{x} \tag{2.29}$$

where $\mathbf{T}$ is an orthonormal matrix of direction cosines given by

$$\mathbf{T} = \begin{bmatrix} \cos(x',\ x) & \cos(x',\ y) & \cos(x',\ z) \\ \cos(y',\ x) & \cos(y',\ y) & \cos(y',\ z) \\ \cos(z',\ x) & \cos(z',\ y) & \cos(z',\ z) \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{bmatrix} \tag{2.30}$$

in which $\cos(x',\ x)$ is the cosine of the angle between the $x'$ direction and the $x$ direction. For two-dimensions, rotation of the axes by an angle $\beta$ gives

$$\begin{aligned} \cos(x',\ x) &= \ \ \cos(y',\ y) = \cos(\beta) \\ \cos(x',\ y) &= -\cos(y',\ x) = \sin(\beta) \\ \cos(x',\ z) &= \ \ \cos(z',\ x) = 0 \\ \cos(y',\ z) &= \ \ \cos(z',\ y) = 0 \\ \cos(z',\ z) &= \ \ 1 \end{aligned}$$

and $\mathbf{T}$ may be treated as a $2 \times 2$ matrix given by

$$\mathbf{T} = \begin{bmatrix} \cos(\beta) & \sin(\beta) \\ -\sin(\beta) & \cos(\beta) \end{bmatrix}$$

Since $\mathbf{T}$ is orthonormal it has the property

$$\mathbf{T}^{-1} = \mathbf{T}^{\mathrm{T}} \tag{2.31}$$

The displacement and traction vectors transform to the prime system in exactly the same way as the coordinates. Hence, we have

$$\begin{aligned} \mathbf{u}' &= \mathbf{T}\mathbf{u} = \bar{\mathbf{u}}' \\ \mathbf{t}' &= \mathbf{T}\mathbf{t} = \mathbf{T}\mathbf{G}^{\mathrm{T}}\boldsymbol{\sigma} = \bar{\mathbf{t}}' \end{aligned} \tag{2.32}$$

### 2.2.5.2 Normal pressure loading

When a pressure loading is applied normal to a surface the traction may be specified as

$$\bar{\mathbf{t}} = \bar{p}\,\mathbf{n} \quad ; \quad \mathbf{n} = [\,n_x \quad n_y \quad n_z\,]^T \tag{2.33}$$

where $\bar{p}$ is the magnitude of the specified traction and $\mathbf{n}$ again is the unit outward normal to the boundary. This is a condition which is often encountered in practical situations in which $\bar{p}$ arises from a fluid or gas loading and tangential surface components are zero.

### 2.2.5.3 Symmetry and repeatability

In many problems, the advantage of symmetry in loading and geometry can be considered when imposing the boundary conditions, thus reducing the whole problem to more manageable proportions. The use of symmetry conditions is so well known to the engineer and physicist that no statement needs to be made about it explicitly. Less known, however, appears to be the use of *repeatability* [9] when an identical structure (and) loading is continuously repeated, as shown in Fig. 2.3 for an infinite blade cascade. Here it is evident that a typical segment (shaded in the figure) behaves identically to the next one, and thus functions such as velocities and displacements at corresponding points of AA and BB are simply identified, i.e.,

$$\mathbf{u}_{\mathrm{I}} = \mathbf{u}_{\mathrm{II}}$$

Similar repeatability, in radial coordinates, occurs very frequently in problems involving turbine or pump impellers. Figure 2.4 shows a typical three-dimensional analysis of such a repeatable segment.

Proper use of symmetry and repeatability can reduce the required compute effort significantly. Similar conditions also can obviously be imposed to enforce conditions of "asymmetry" where

$$\mathbf{u}_{\mathrm{I}} = -\mathbf{u}_{\mathrm{II}}$$

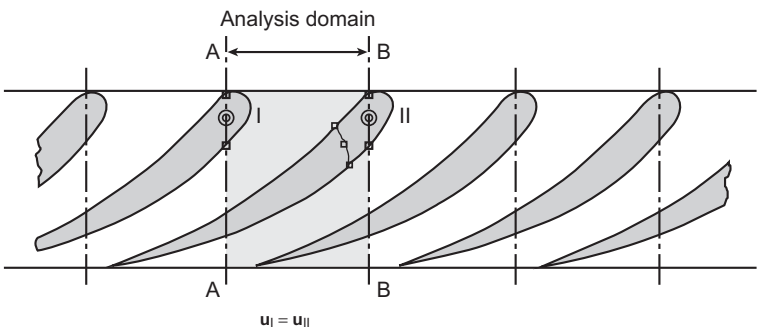



**FIGURE 2.3**
Repeatability segments and analysis domain (shaded).

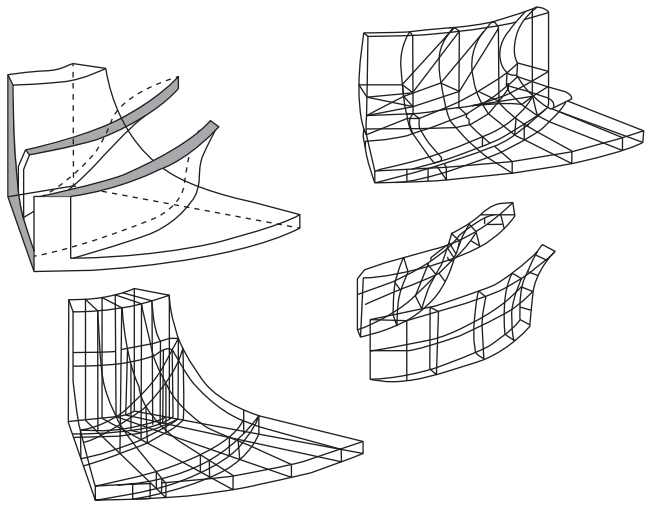


**FIGURE 2.4**
Repeatable sector in analysis of an impeller.

### 2.2.6 Initial conditions

For transient problems it is necessary to describe the state of the body when the analysis starts. These are called *initial conditions* and since a second derivative in time appears to describe the acceleration it is necessary to specify the initial displacement and velocity (i.e., the first time derivative of the displacement). Accordingly, the initial displacement is given as

$$\mathbf{u}(\mathbf{x},\ 0) = \bar{\mathbf{d}}(\mathbf{x}) \tag{2.34}$$

and the initial velocity by

$$\mathbf{v}(\mathbf{x},\ 0) = \dot{\mathbf{u}}(\mathbf{x},\ 0) = \bar{\mathbf{v}}(\mathbf{x}) \tag{2.35}$$

### 2.2.7 Transformation of stress and strain

The transformation of coordinates to a prime system given in Eq. (2.29) may also be used to define transformations for stresses and strains. Expressing the stress in the Cartesian tensor form (see Appendix C)

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix} \tag{2.36}$$

the transformations are given by

$$\boldsymbol{\sigma}' = \mathbf{T}\boldsymbol{\sigma}\mathbf{T}^{\mathrm{T}} \tag{2.37}$$

Using Voigt notation (2.37) may be expressed in matrix form as

$$\boldsymbol{\sigma}' = \mathbf{T}_\sigma \boldsymbol{\sigma} \tag{2.38}$$

where the stress matrices are expanded using (2.13) and the transformation matrix is given by

$$\mathbf{T}_\sigma = \begin{bmatrix} t_{11}t_{11} & t_{12}t_{12} & t_{13}t_{13} & 2\,t_{11}t_{12} & 2\,t_{12}t_{13} & 2\,t_{13}t_{11} \\ t_{21}t_{21} & t_{22}t_{22} & t_{23}t_{23} & 2\,t_{21}t_{22} & 2\,t_{22}t_{23} & 2\,t_{23}t_{21} \\ t_{31}t_{31} & t_{32}t_{32} & t_{33}t_{33} & 2\,t_{31}t_{32} & 2\,t_{32}t_{33} & 2\,t_{33}t_{31} \\ t_{11}t_{21} & t_{12}t_{22} & t_{13}t_{23} & (t_{11}t_{22}+t_{12}t_{21}) & (t_{12}t_{23}+t_{13}t_{22}) & (t_{13}t_{21}+t_{11}t_{23}) \\ t_{21}t_{31} & t_{22}t_{32} & t_{23}t_{33} & (t_{21}t_{32}+t_{22}t_{31}) & (t_{22}t_{33}+t_{23}t_{32}) & (t_{23}t_{31}+t_{21}t_{33}) \\ t_{31}t_{11} & t_{32}t_{12} & t_{33}t_{13} & (t_{31}t_{12}+t_{32}t_{11}) & (t_{32}t_{13}+t_{33}t_{12}) & (t_{33}t_{11}+t_{31}t_{13}) \end{bmatrix} \tag{2.39}$$

Similarly the strain transformation in Voigt notation is expressed by

$$\boldsymbol{\varepsilon}' = \mathbf{T}_\varepsilon \boldsymbol{\varepsilon} \tag{2.40}$$

where strains are expanded using (2.6) and the transformation matrix is given by

$$\mathbf{T}_\varepsilon = \begin{bmatrix} t_{11}t_{11} & t_{12}t_{12} & t_{13}t_{13} & t_{11}t_{12} & t_{12}t_{13} & t_{13}t_{11} \\ t_{21}t_{21} & t_{22}t_{22} & t_{23}t_{23} & t_{21}t_{22} & t_{22}t_{23} & t_{23}t_{21} \\ t_{31}t_{31} & t_{32}t_{32} & t_{33}t_{33} & t_{31}t_{32} & t_{32}t_{33} & t_{33}t_{31} \\ 2\,t_{11}t_{21} & 2\,t_{12}t_{22} & 2\,t_{13}t_{23} & (t_{11}t_{22}+t_{12}t_{21}) & (t_{12}t_{23}+t_{13}t_{22}) & (t_{13}t_{21}+t_{11}t_{23}) \\ 2\,t_{21}t_{31} & 2\,t_{22}t_{32} & 2\,t_{23}t_{33} & (t_{21}t_{32}+t_{22}t_{31}) & (t_{22}t_{33}+t_{23}t_{32}) & (t_{23}t_{31}+t_{21}t_{33}) \\ 2\,t_{31}t_{11} & 2\,t_{32}t_{12} & 2\,t_{33}t_{13} & (t_{31}t_{12}+t_{32}t_{11}) & (t_{32}t_{13}+t_{33}t_{12}) & (t_{33}t_{11}+t_{31}t_{13}) \end{bmatrix} \tag{2.41}$$

The differences between $\mathbf{T}_\sigma$ and $\mathbf{T}_\varepsilon$ occur from the use of the engineering definition of shearing strain where we have introduced

$$\gamma_{xy} = 2\,\varepsilon_{xy} \text{ etc.}$$

for tensor components (see Appendix C).

#### 2.2.7.1 Energy

Stored energy is an important quantity in mechanics that may be used as the basis of conservation principles. It may also be used to define the relationship between stress and strain transformations. Using Voigt notation we define the stored energy as

$$\mathcal{E} = \boldsymbol{\sigma}^{\mathrm{T}} \boldsymbol{\varepsilon} \tag{2.42}$$

Energy is invariant with respect to the coordinate frame we use for its computation. Accordingly,

$$\mathcal{E} = \boldsymbol{\sigma}'^{T} \boldsymbol{\varepsilon}' = \boldsymbol{\sigma}^{\mathrm{T}} \mathbf{T}_\sigma^{\mathrm{T}} \mathbf{T}_\varepsilon \boldsymbol{\varepsilon} \tag{2.43}$$

which yields

$$\mathbf{T}_\sigma^{\mathrm{T}} \mathbf{T}_\varepsilon = \mathbf{I} \tag{2.44}$$

From this we can observe that

$$\mathbf{T}_{\varepsilon}^{-1} = \mathbf{T}_{\sigma}^{\mathrm{T}} \text{ and } \mathbf{T}_{\sigma}^{-1} = \mathbf{T}_{\varepsilon}^{\mathrm{T}} \tag{2.45}$$

and, thus, the inverses to the stress and strain transformation matrices may be computed by a simple transpose.

### 2.2.8 Stress-strain relations: Elasticity matrix

The equations to this point are all independent of the particular material being considered. To complete the theory we need to provide a statement of how the stresses are related to the strains. Such relations are known as *stress-strain equations* or *constitutive equations*. They can be of very complex form; however, in this volume we shall consider the simplest "linear" elastic case in which a strain response is instantaneous and complete upon application of a constant stress. Upon removal of the stress the recovery is also complete and instantaneous.

Using this hypothesis the stress-strain equations for a *linearly elastic material* may be expressed by

$$\boldsymbol{\sigma} = \mathbf{D}\left(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_0\right) + \boldsymbol{\sigma}_0 \tag{2.46a}$$

or by

$$\boldsymbol{\varepsilon} = \mathbf{D}^{-1}\left(\boldsymbol{\sigma} - \boldsymbol{\sigma}_0\right) + \boldsymbol{\varepsilon}_0 \tag{2.46b}$$

The $\mathbf{D}$ matrix is known as the *elasticity matrix of moduli* and the $\mathbf{D}^{-1}$ matrix as the *elasticity matrix of compliances* [5]. The quantities $\boldsymbol{\sigma}_0$ and $\boldsymbol{\varepsilon}_0$ represent stress and strain values that are created by sources other than mechanical strain, $\boldsymbol{\varepsilon}$. For example, in thermal problems a change in temperature can create a strain $\boldsymbol{\varepsilon}_0$.

For elastic materials it is also possible to deduce the stress from a stored energy function $W$ expressed in terms of strains using

$$\boldsymbol{\sigma} = \frac{\partial W}{\partial \boldsymbol{\varepsilon}} \tag{2.47}$$

This holds for both linear and nonlinear materials and, thus, may be used to generalize the above expressions. For the linear elastic equations given in (2.46a) the stored energy function is given by

$$W(\boldsymbol{\varepsilon}) = \frac{1}{2}\boldsymbol{\varepsilon}^{\mathrm{T}}\mathbf{D}\boldsymbol{\varepsilon} + \boldsymbol{\varepsilon}^{\mathrm{T}}\left(\boldsymbol{\sigma}_0 - \mathbf{D}\boldsymbol{\varepsilon}_0\right) \tag{2.48}$$

One can also define a complementary energy $U(\boldsymbol{\sigma})$ from which the strains are deduced using

$$\boldsymbol{\varepsilon} = \frac{\partial U}{\partial \boldsymbol{\sigma}} \tag{2.49}$$

where for the linear elastic case

$$U(\boldsymbol{\sigma}) = \frac{1}{2}\boldsymbol{\sigma}^{\mathrm{T}}\mathbf{D}^{-1}\boldsymbol{\sigma} + \boldsymbol{\sigma}^{\mathrm{T}}\left(\boldsymbol{\varepsilon}_0 - \mathbf{D}^{-1}\boldsymbol{\sigma}_0\right) \tag{2.50}$$

Use of a stored or complementary energy form also ensures that $\mathbf{D}$ and $\mathbf{D}^{-1}$ are symmetric.

For the present we shall ignore the $\boldsymbol{\varepsilon}_0$ and $\boldsymbol{\sigma}_0$ terms in order to describe more fully the form of the elasticity matrix. After obtaining final results they may be again added by replacing $\boldsymbol{\varepsilon}$ and $\boldsymbol{\sigma}$ by $\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_0$ and $\boldsymbol{\sigma} - \boldsymbol{\sigma}_0$, respectively.

#### 2.2.8.1 *Isotropic materials*

An isotropic material is one in which the elasticity matrix does not change when the coordinate transformation given by (2.29) is applied. We write a general expression for isotropic materials in terms of the six stress and strain terms. We may use any two independent elastic constants for an isotropic material [5,10]. Here we use Young's modulus of elasticity, $E$, and Poisson's ratio, $\nu$. In Table 2.1 we indicate the relationships between $E$, $\nu$, and other parameters frequently encountered in the literature. Using Cartesian coordinates, for example, the expression is given by

$$\begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{Bmatrix} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & -\nu & 0 & 0 & 0 \\ -\nu & 1 & -\nu & 0 & 0 & 0 \\ -\nu & -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2(1+\nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & 2(1+\nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & 2(1+\nu) \end{bmatrix} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{Bmatrix} \tag{2.51}$$

Inverting to obtain the appropriate elasticity matrix of moduli yields the result

$$\mathbf{D} = \frac{E}{d} \begin{bmatrix} (1-\nu) & \nu & \nu & 0 & 0 & 0 \\ \nu & (1-\nu) & \nu & 0 & 0 & 0 \\ \nu & \nu & (1-\nu) & 0 & 0 & 0 \\ 0 & 0 & 0 & (1-2\nu)/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & (1-2\nu)/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & (1-2\nu)/2 \end{bmatrix} \tag{2.52}$$

where $d = (1+\nu)(1-2\nu)$. The form of $d$ places restrictions on the admissible values of $\nu$ to keep all the material parameters positive. Thus, we have

$$-1 < \nu < \frac{1}{2}$$

The limiting values of the parameters (namely $-1$ and $1/2$) are permitted by rewriting the material model in which Lagrange multipliers replace the terms that are indefinite.

**Table 2.1** Relations between Isotropic Elastic Parameters

| Parameters | $E, \nu$ | $K, G$ | $\lambda, \mu$ |
|---|---|---|---|
| $E$ | – | $9KG/(3K+G)$ | $\mu(3\lambda+2\mu)/(\lambda+\mu)$ |
| $\nu$ | – | $(3K-2G)/(6K+2G)$ | $\lambda/(\lambda+\mu)/2$ |
| $K$ | $E/(1-2\nu)/3$ | – | $\lambda+2\mu/3$ |
| $G=\mu$ | $E/(1+\nu)/2$ | $G$ | $\mu$ |
| $\lambda$ | $\nu E/(1+\nu)/(1-2\nu)$ | $K-2G/3$ | – |

The case where $\nu = 1/2$ is associated with materials which are *incompressible* and we shall devote special attention to this problem in Chapter 10 since it has relevance for many applications in solid and fluid mechanics. Generally, of course, no material can be incompressible and we are only interested in the case where $\nu$ approaches $1/2$. However, even for this case we will find that care must be used when developing a finite element form.

For isotropic materials the expression for two-dimensional problems is written in terms of the four stress and strain terms by omitting the last two rows and columns in (2.51). Using Cartesian coordinates the expression then becomes

$$\begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \end{Bmatrix} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & -\nu & 0 \\ -\nu & 1 & -\nu & 0 \\ -\nu & -\nu & 1 & 0 \\ 0 & 0 & 0 & 2(1+\nu) \end{bmatrix} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \end{Bmatrix} \tag{2.53}$$

For the plane stress case we must set $\sigma_z$ to zero to compute the appropriate $\mathbf{D}$ matrix. This yields the result

$$\varepsilon_z = -\frac{\nu}{E}(\sigma_x + \sigma_y) \tag{2.54}$$

and including this in the inverse of (2.53) we obtain

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \end{Bmatrix} = \frac{E}{(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 & 0 \\ \nu & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & (1-\nu)/2 \end{bmatrix} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \end{Bmatrix} \tag{2.55}$$

We have filled in the third column and row of the $\mathbf{D}$ array so that the correct stresses are obtained.

For the plane strain and axisymmetric problems the inverse may be performed directly from (2.53), as all components of stress can exist. Accordingly, for these two cases we obtain (again writing for the Cartesian coordinate form)

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \end{Bmatrix} = \frac{E}{d} \begin{bmatrix} (1-\nu) & \nu & \nu & 0 \\ \nu & (1-\nu) & \nu & 0 \\ \nu & \nu & (1-\nu) & 0 \\ 0 & 0 & 0 & (1-2\nu)/2 \end{bmatrix} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \end{Bmatrix} \tag{2.56}$$

which is identical to the three-dimensional problem if the last two rows and columns of each array in (2.52) are omitted. Here we observe that the case where $\varepsilon_z$ is zero is treated merely by inserting that value when computing stresses; however, $\sigma_z$ will always exist unless the Poisson ratio, $\nu$, is zero.

#### 2.2.8.2 *Deviatoric and pressure-volume relations*

For an isotropic material we can substitute the expressions for deviatoric and volumetric strain into the stress-strain relation to obtain

$$\boldsymbol{\sigma} = \mathbf{D}\left(\mathbf{e} + \frac{1}{3}\mathbf{m}\varepsilon_v\right) \tag{2.57}$$

Using the relationship between $E$ and $\nu$ and $G$ and $K$ from Table 2.1 we obtain

$$\mathbf{D}\,\mathbf{e} = G \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{e} = 2\,G\,\mathbf{I}_0\,\mathbf{e} \tag{2.58a}$$

and

$$\frac{1}{3}\,\mathbf{D}\,\mathbf{m} = K\,\mathbf{m} \tag{2.58b}$$

and thus, using (2.58a), we can write the deviatoric stress-strain relation in Voigt notation as

$$\mathbf{s} = 2\,G\,\mathbf{I}_0\,\mathbf{e} \tag{2.59a}$$

and the pressure-volume relation as

$$p = K\,\varepsilon_v \tag{2.59b}$$

This form will be useful when we consider incompressible and nearly incompressible materials in Chapter 10. For an incompressible material the volume change $\varepsilon_v$ is zero and the constitutive equation for an isotropic material reduces to

$$\boldsymbol{\sigma} = p\,\mathbf{m} + 2\,G\,\mathbf{I}_0\,\mathbf{e} \tag{2.60}$$

#### 2.2.8.3 Anisotropic materials

Many materials have structure such that they are not isotropic; these are said to be *anisotropic materials*. There are many forms of stress-strain relations for anisotropic materials. We write the relationship for a *general* anisotropic linearly elastic material as

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{Bmatrix} = \begin{bmatrix} D_{11} & D_{12} & D_{13} & D_{14} & D_{15} & D_{16} \\ D_{21} & D_{22} & D_{23} & D_{24} & D_{25} & D_{26} \\ D_{31} & D_{32} & D_{33} & D_{34} & D_{35} & D_{36} \\ D_{41} & D_{42} & D_{43} & D_{44} & D_{45} & D_{46} \\ D_{51} & D_{52} & D_{53} & D_{54} & D_{55} & D_{56} \\ D_{61} & D_{62} & D_{63} & D_{64} & D_{65} & D_{66} \end{bmatrix} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{Bmatrix} \tag{2.61}$$

For an elastic material deduced from (2.47) the $\mathbf{D}$ matrix must be symmetric and, hence, $D_{ij} = D_{ji}$. This results in a possibility of 21 elastic constants for the general problem [4,11,12].

An important class of anisotropic materials is one for which three planes of symmetry exist; this is called an *orthotropic material*. Here the principal axes are also in rectangular Cartesian coordinates. For orthotropic materials it is common to define the elastic material parameters in terms of their Young's moduli, Poisson ratios, and shear moduli.

If we let $x'$, $y'$, and $z'$ be the three axes of material symmetry, the elastic strain-stress relations may be expressed as

$$\begin{Bmatrix} \varepsilon_{x'} \\ \varepsilon_{y'} \\ \varepsilon_{z'} \\ \gamma_{x'y'} \\ \gamma_{y'z'} \\ \gamma_{z'x'} \end{Bmatrix} = \begin{bmatrix} \frac{1}{E_{x'}} & -\frac{\nu_{x'y'}}{E_{y'}} & -\frac{\nu_{x'z'}}{E_{z'}} & 0 & 0 & 0 \\ -\frac{\nu_{y'x'}}{E_{x'}} & \frac{1}{E_{y'}} & -\frac{\nu_{y'z'}}{E_{x'}} & 0 & 0 & 0 \\ -\frac{\nu_{z'x'}}{E_{x'}} & -\frac{\nu_{z'y'}}{E_{y'}} & \frac{1}{E_{z'}} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G_{x'y'}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{y'z'}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{z'x'}} \end{bmatrix} \begin{pmatrix} \sigma_{x'} \\ \sigma_{y'} \\ \sigma_{z'} \\ \tau_{x'y'} \\ \tau_{y'z'} \\ \tau_{z'x'} \end{pmatrix} \tag{2.62}$$

where $E_{x'}$, $E_{y'}$, $E_{z'}$ are elastic moduli; $\nu_{x'y'}$, $\nu_{x'z'}$, etc., are Poisson ratios; and $G_{x'y'}$, $G_{y'z'}$, $G_{z'x'}$ are elastic shear moduli. Again symmetry of the $\mathbf{D}^{-1}$ matrix results in

$$\frac{\nu_{i'j'}}{E_{j'}} = \frac{\nu_{j'i'}}{E_{i'}} \tag{2.63}$$

thus reducing the number of independent components for the three-dimensional case to nine parameters (three direct moduli, 3 Poisson ratios, and three shear moduli).

The elastic moduli for the $\mathbf{D}'$ matrix are computed by inverting the square matrix appearing in Eq. (2.62).

The inverse may be written as

$$\boldsymbol{\sigma}' = \mathbf{D}'\,\boldsymbol{\varepsilon}' \tag{2.64}$$

If the principal material axes are expressed by the directions given in (2.29), it is necessary to transform Eq. (2.64) to the form of (2.46a) before proceeding with an analysis. This is most easily performed by noting an equality of work given by

$$\begin{aligned} \boldsymbol{\sigma}'^{T}\boldsymbol{\varepsilon}' &= \boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\varepsilon} \\ \boldsymbol{\varepsilon}'^{T}\mathbf{D}'\,\boldsymbol{\varepsilon}' &= \boldsymbol{\varepsilon}^{\mathrm{T}}\mathbf{D}\,\boldsymbol{\varepsilon} \end{aligned} \tag{2.65}$$

and using the expressions for transformation given by Eq. (2.41) in Eq. (2.65) gives

$$\mathbf{D} = \mathbf{T}_{\varepsilon}^{\mathrm{T}}\mathbf{D}'\mathbf{T}_{\varepsilon}. \tag{2.66}$$

Such a transformation also has been used in a slightly different context in Chapter 1 [viz. Eq. (1.27)] to transform a stiffness matrix.

For treatment of anisotropic materials in the two-dimensional problems, it is necessary for the direction normal to the plane of deformation (i.e., the $z$ direction for plane problems or $\theta$ direction for axisymmetric problems) to be a direction of *material symmetry*. For this case we may write a general relationship for linearly elastic deformation as (using the Cartesian form)

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \end{Bmatrix} = \begin{bmatrix} D_{11} & D_{12} & D_{13} & D_{14} \\ D_{21} & D_{22} & D_{23} & D_{24} \\ D_{31} & D_{32} & D_{33} & D_{34} \\ D_{41} & D_{42} & D_{43} & D_{44} \end{bmatrix} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \end{Bmatrix} \tag{2.67}$$

For an elastic material the **D** matrix must be symmetric and, hence, $D_{ij} = D_{ji}$. This results in a possibility of 10 elastic constants for the plane or axisymmetric problem.

In order to consider the plane stress case it is again necessary to impose the constraint $\sigma_z = 0$. If we solve for $\varepsilon_z$ using the third row of (2.67) we have

$$\varepsilon_z = -\left[D_{31}\,\varepsilon_x + D_{32}\,\varepsilon_y + D_{34}\,\gamma_{xy}\right]/D_{33} \tag{2.68}$$

which may be substituted into the remaining three equations in (2.67) to give

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \end{Bmatrix} = \begin{bmatrix} \widehat{D}_{11} & \widehat{D}_{12} & 0 & \widehat{D}_{14} \\ \widehat{D}_{21} & \widehat{D}_{22} & 0 & \widehat{D}_{24} \\ 0 & 0 & 0 & 0 \\ \widehat{D}_{41} & \widehat{D}_{42} & 0 & \widehat{D}_{44} \end{bmatrix} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \end{Bmatrix} \tag{2.69}$$

in which

$$\widehat{D}_{ij} = D_{ij} - D_{i3} D_{33}^{-1} D_{3j} \tag{2.70}$$

are reduced elastic moduli.

**Example 2.1.** Anisotropic, stratified, material

With the $z$-axis representing the normal to the planes of stratification, as shown in Fig. 2.5, we can rewrite (2.62) (again ignoring the initial strains and stresses for

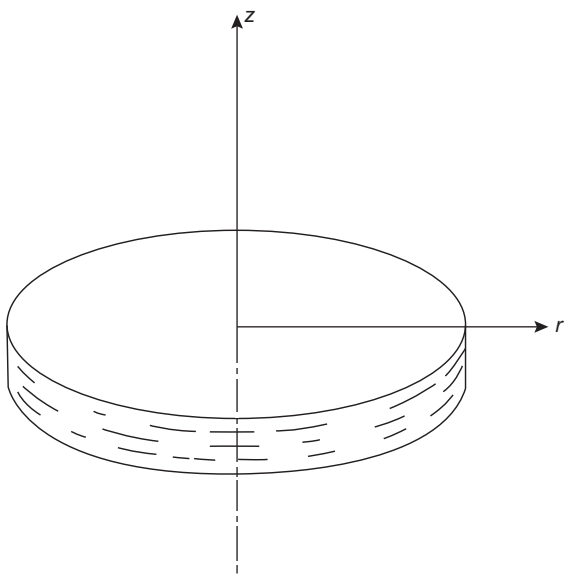



**FIGURE 2.5**
Axisymmetrically stratified material.

convenience) as

$$\begin{aligned}
\varepsilon_r &= \frac{\sigma_r}{E_1} - \frac{\nu_2\sigma_z}{E_2} - \frac{\nu_1\sigma_\theta}{E_1} \\
\varepsilon_z &= -\frac{\nu_2\sigma_r}{E_1} + \frac{\sigma_z}{E_2} - \frac{\nu_2\sigma_\theta}{E_1} \\
\varepsilon_\theta &= -\frac{\nu_1\sigma_r}{E_1} - \frac{\nu_2\sigma_z}{E_2} + \frac{\sigma_\theta}{E_1} \\
\gamma_{rz} &= \frac{\tau_{rz}}{G}
\end{aligned} \tag{2.71}$$

Writing the parameters as

$$\frac{E_1}{E_2} = n, \qquad \frac{G}{E_2} = m \quad \text{and} \quad d = (1+\nu_1)(1-\nu_1-2n\nu_2^2)$$

we have on solving for the stresses that

$$\mathbf{D} = \frac{E_2}{d}\begin{bmatrix} n(1-n\nu_2^2) & n\nu_2(1+\nu_1) & n(\nu_1+n\nu_2^2) & 0 \\ n\nu_2(1+\nu_1) & 1-\nu_1^2 & n\nu_2(1+\nu_1) & 0 \\ n(\nu_1+n\nu_2^2) & n\nu_2(1+\nu_1) & n(1-n\nu_2^2) & 0 \\ 0 & 0 & 0 & md \end{bmatrix} \tag{2.72}$$

#### 2.2.8.4 *Initial strain—thermal effects*

Initial strains may be due to many causes. Shrinkage, crystal growth, or temperature change will, in general, result in an initial strain vector

$$\boldsymbol{\varepsilon}_0 = [\,\varepsilon_{x0} \quad \varepsilon_{y0} \quad \varepsilon_{z0} \quad \gamma_{xy0} \quad \gamma_{yz0} \quad \gamma_{zx0}\,]^{\mathrm{T}} \tag{2.73}$$

The initial strain will usually depend on position.

As an example, consider the effects of change in temperature in an isotropic material. The initial strain for a temperature change $\Delta T = T - T_0$ (with $T_0$ a temperature where no straining is caused) with a *linear* coefficient of thermal expansion $\alpha$ is given by

$$\boldsymbol{\varepsilon}_0 = \alpha\,\Delta T\,\mathbf{m} \tag{2.74}$$

For an isotropic material normal strains $\varepsilon_x$, $\varepsilon_y$, $\varepsilon_z$ are all equal and no shear strains are caused by a temperature change.

Anisotropic materials present no special problems with the coefficients of thermal expansion varying with direction in the material. For example, in an orthotropic material no shearing strains are caused for the principal material directions and we may replace (2.74) by

$$\boldsymbol{\varepsilon}_0' = \Delta T\,[\,\alpha_{x'} \quad \alpha_{y'} \quad \alpha_{z'} \quad 0 \quad 0 \quad 0\,]^{\mathrm{T}} \tag{2.75}$$

It is now again necessary to use the transformation between principal material directions and those used for coordinates of the analysis using

$$\boldsymbol{\varepsilon}_0 = \mathbf{T}_\varepsilon^{-1}\,\boldsymbol{\varepsilon}_0' = \mathbf{T}_\sigma^{T}\,\boldsymbol{\varepsilon}_0' \tag{2.76}$$

where $\mathbf{T}_\sigma$ is given by (2.39). Now, in general, the $\gamma_{xy0}$, $\gamma_{yz0}$, $\gamma_{zx0}$ components are no longer equal to zero.

## 2.3 General quasi-harmonic equation

### 2.3.1 Governing equations: Flux and continuity

As a second example of a continuous formulation we consider problems that may be defined by a form we call a "quasi-harmonic equation." In many physical situations the generation of initial stresses or strains is created by the *diffusion* or *flow* of some quantity such as heat, mass, or concentration within the solid. In such problems, a scalar variable $\phi$ is defined to describe the temperature, fluid head, or some other physical variable. The balance equation is then written as the rate of transfer per unit area of a *flux*, $\mathbf{q}$, which in Cartesian coordinates has the components

$$\mathbf{q} = [\, q_x \quad q_y \quad q_z \,]^{\mathrm{T}} \tag{2.77}$$

The balance or continuity requirement per unit volume may be written as

$$-\left(\frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} + \frac{\partial q_z}{\partial z}\right) + Q = c\frac{\partial \phi}{\partial t} \tag{2.78}$$

where $Q$ is a source term describing the rate at which the relevant quantity is generated (or removed) and $c$ is a material parameter describing the transient behavior. Equation (2.78) is satisfied in the domain of the problem which we again denote by $\Omega$. Introducing the gradient operator

$$\boldsymbol{\nabla} = \left[\begin{array}{ccc} \dfrac{\partial}{\partial x} & \dfrac{\partial}{\partial y} & \dfrac{\partial}{\partial z} \end{array}\right]^{\mathrm{T}} \tag{2.79}$$

we can write (2.78) in matrix form as

$$-\boldsymbol{\nabla}^{\mathrm{T}}\mathbf{q} + Q = c\,\frac{\partial \phi}{\partial t} \tag{2.80}$$

where $\boldsymbol{\nabla}^{\mathrm{T}}\mathbf{q}$ is called the *divergence* of the flux and often written as div $\mathbf{q}$.

### 2.3.2 Boundary conditions

On the boundaries of the domain we shall usually encounter one of the following conditions:

**1.** On $\Gamma_\phi$, the potential (e.g., the temperature) is specified as

$$\phi = \bar{\phi} \tag{2.81}$$

**2.** On $\Gamma_q$, the normal component of flow (or flux), $q_n$, is given as

$$q_n = \bar{q} + H\,(\phi - \phi_0)$$

where $H$ is a transfer or radiation coefficient, $\phi_0$ is a known equilibrium value, and $\bar{q}$ is a specified value. Here $q_n$ is defined as

$$q_n = \mathbf{n}^{\mathrm{T}}\mathbf{q} \quad \text{with} \quad \mathbf{n} = [\, n_x \quad n_y \quad n_z \,]^{\mathrm{T}}$$

where $\mathbf{n}$ is a vector of direction cosines of the normal to the boundary surface. Accordingly, we may write the second boundary condition

$$q_n = \bar{q} + H\,(\phi - \phi_0)$$

which holds on $\Gamma_q$.

### 2.3.3 Initial condition

For transient problems only a single initial condition is required and it is specified as

$$\phi(\mathbf{x},\ 0) = \bar{d}(\mathbf{x}) \tag{2.82}$$

in the domain $\Omega$.

### 2.3.4 Constitutive behavior

Generally the rates of flow are related to the *gradient* of $\phi$. A general linear relationship will be of the form

$$\mathbf{q} = \begin{Bmatrix} q_x \\ q_y \\ q_z \end{Bmatrix} = - \begin{bmatrix} k_{xx} & k_{xy} & k_{xz} \\ k_{yx} & k_{yy} & k_{yz} \\ k_{zx} & k_{zy} & k_{zz} \end{bmatrix} \begin{Bmatrix} \dfrac{\partial \phi}{\partial x} \\ \dfrac{\partial \phi}{\partial y} \\ \dfrac{\partial \phi}{\partial z} \end{Bmatrix} = -\,\mathbf{k}\,\boldsymbol{\nabla}\phi \tag{2.83}$$

where $\mathbf{k}$ is a symmetric form due to energy arguments (i.e., $k_{xy} = k_{yx}$, etc.) and is variously referred to as Fourier's, Fick's, or Darcy's law depending on the physical problem.

### 2.3.5 Irreducible form in $\phi$

The final governing differential equation for the "potential" $\phi$ is obtained by substitution of Eq. (2.83) into (2.80), leading to

$$\boldsymbol{\nabla}^{\mathrm{T}}\left(\mathbf{k}\,\boldsymbol{\nabla}\phi\right) + Q = c\,\frac{\partial \phi}{\partial t} \tag{2.84}$$

which has to be solved in the domain $\Omega$. The governing equation is now expressed in a minimum number of dependent variables (namely one). We call all such forms "irreducible" in subsequent discussion.

### 2.3.6 Anisotropic and isotropic forms for k: Transformations

If we consider the general statement of Eq. (2.83) as being determined for an arbitrary set of coordinate axes $x$, $y$, $z$ we shall find that it is always possible to determine

locally another set of axes $x'$, $y'$, $z'$ with respect to which the matrix $\mathbf{k}'$ becomes diagonal, as shown in Fig. 2.6. With respect to such axes we have

$$\mathbf{k}' = \begin{bmatrix} k_{x'x'} & 0 & 0 \\ 0 & k_{y'y'} & 0 \\ 0 & 0 & k_{z'z'} \end{bmatrix} \tag{2.85}$$

Thus, the general form of $\mathbf{k}$ has only three components which are associated with three orthogonal axes. Such materials are called *anisotropic* or *orthotropic*.

The governing differential equation (2.84) for these axes can be written

$$\frac{\partial}{\partial x'}\left(k_{x'x'}\frac{\partial \phi}{\partial x'}\right) + \frac{\partial}{\partial y'}\left(k_{y'y'}\frac{\partial \phi}{\partial y'}\right) + \frac{\partial}{\partial z'}\left(k_{z'z'}\frac{\partial \phi}{\partial z'}\right) + Q = c\,\frac{\partial \phi}{\partial t} \tag{2.86a}$$

or in matrix form as

$$\left(\nabla'\right)^{\mathrm{T}}\left(\mathbf{k}'\,\nabla'\phi\right) + Q = c\,\frac{\partial \phi}{\partial t} \tag{2.86b}$$

where

$$\nabla' = \begin{bmatrix} \dfrac{\partial}{\partial x'} & \dfrac{\partial}{\partial y'} & \dfrac{\partial}{\partial z'} \end{bmatrix}^{\mathrm{T}}$$

defines the gradient operator for the "prime" coordinate system.

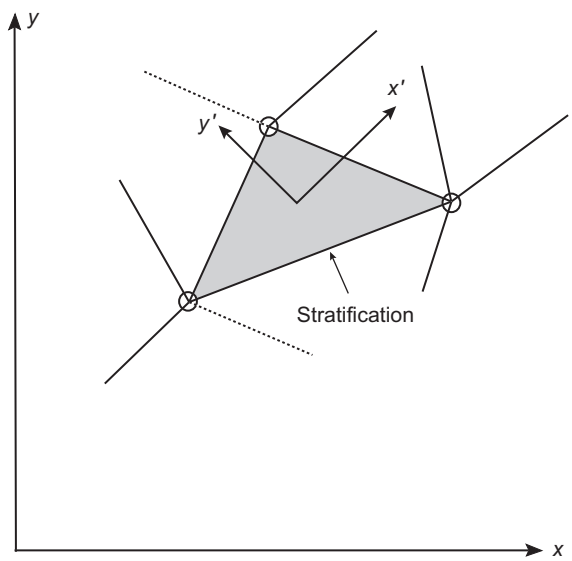



**FIGURE 2.6**

Anisotropic material. Local coordinates coincide with the principal directions of stratification.

Alternatively, knowing $\mathbf{k}'$ and the orientation of the axes $x'$, $y'$, $z'$, a transformation of coordinates is given by [viz. (2.29)]

$$\mathbf{x}' = \mathbf{T}\,\mathbf{x}$$

in which $\mathbf{T}$ are the *direction cosines* defined in (2.30). The inverse of $\mathbf{T}$ is equal to its transpose; hence

$$\mathbf{x} = \mathbf{T}^{\mathrm{T}}\,\mathbf{x}'$$

In addition we may write the derivatives with respect to the prime axes as

$$\begin{aligned}
\frac{\partial \cdot}{\partial x} &= \frac{\partial x'}{\partial x}\frac{\partial \cdot}{\partial x'} + \frac{\partial y'}{\partial x}\frac{\partial \cdot}{\partial y'} + \frac{\partial z'}{\partial x}\frac{\partial \cdot}{\partial z'} \\
\frac{\partial \cdot}{\partial y} &= \frac{\partial x'}{\partial y}\frac{\partial \cdot}{\partial x'} + \frac{\partial y'}{\partial y}\frac{\partial \cdot}{\partial y'} + \frac{\partial z'}{\partial y}\frac{\partial \cdot}{\partial z'} \\
\frac{\partial \cdot}{\partial z} &= \frac{\partial x'}{\partial z}\frac{\partial \cdot}{\partial x'} + \frac{\partial y'}{\partial z}\frac{\partial \cdot}{\partial y'} + \frac{\partial z'}{\partial z}\frac{\partial \cdot}{\partial z'}
\end{aligned}$$

and thus

$$\boldsymbol{\nabla}(\cdot) = \mathbf{T}^{\mathrm{T}}\boldsymbol{\nabla}'(\cdot)$$

or alternatively

$$\boldsymbol{\nabla}'(\cdot) = \mathbf{T}\,\boldsymbol{\nabla}(\cdot)$$

Using the above we obtain the expression

$$\left(\boldsymbol{\nabla}'\right)^{\mathrm{T}}\left(\mathbf{k}'\boldsymbol{\nabla}'\phi\right) = \left(\boldsymbol{\nabla}\right)^{\mathrm{T}}\mathbf{T}^{\mathrm{T}}\left(\mathbf{k}'\,\mathbf{T}\,\boldsymbol{\nabla}\phi\right) \tag{2.87a}$$

or

$$\mathbf{k} = \mathbf{T}^{T}\mathbf{k}'\,\mathbf{T} \quad \text{or} \quad \mathbf{k}' = \mathbf{T}\,\mathbf{k}\,\mathbf{T}^{\mathrm{T}} \tag{2.87b}$$

Lastly for an isotropic material we can write

$$\mathbf{k} = k\,\mathbf{I} \tag{2.88}$$

where $\mathbf{I}$ is an identity matrix and thus we have also

$$\mathbf{k}' = k\,\mathbf{T}\,\mathbf{I}\mathbf{T}^{\mathrm{T}} = k\,\mathbf{I} \tag{2.89}$$

### 2.3.7 Two-dimensional problems

For two-dimensional plane problems the above equations may be used directly by merely deleting the third row and column of all the arrays. For the axisymmetric problem the gradient of $\phi$ is given by

$$\boldsymbol{\nabla}\phi = \begin{bmatrix} \dfrac{\partial \phi}{\partial r} & \dfrac{\partial \phi}{\partial z} \end{bmatrix}^{\mathrm{T}} \tag{2.90}$$

the flux by the two components

$$\mathbf{q} = [\,q_r \quad q_z\,]^{\mathrm{T}} \tag{2.91}$$

and the continuity requirement by

$$-\left(\frac{\partial q_r}{\partial r}+\frac{q_r}{r}+\frac{\partial q_z}{\partial z}\right)+Q=c\,\frac{\partial \phi}{\partial t} \tag{2.92}$$

We note now that the divergence of the flux is not the same as applying the transpose of the gradient to the flux. This is, of course, due to the presence of curvilinear coordinates instead of the simple Cartesian ones.

Finally, the constitutive behavior is given by

$$\left\{\begin{array}{c} q_r \\ q_z \end{array}\right\}=-\left[\begin{array}{cc} k_{rr} & k_{rz} \\ k_{zr} & k_{zz} \end{array}\right]\left\{\begin{array}{c} \dfrac{\partial \phi}{\partial r} \\ \dfrac{\partial \phi}{\partial z} \end{array}\right\} \tag{2.93}$$

where $k_{rz}=k_{zr}$. These values may be determined from the values in the principal $x'$, $y'$ directions using the coordinate transformation described above.

Assuming an isotropic material and substituting (2.93) into (2.92) yields the governing differential equation

$$k\left(\frac{\partial^2 \phi}{\partial r^2}+\frac{1}{r}\,\frac{\partial \phi}{\partial r}+\frac{\partial^2 \phi}{\partial z^2}\right)+Q=c\,\frac{\partial \phi}{\partial t} \tag{2.94}$$

## 2.4 Concluding remarks

This chapter has presented a summary of the equations that will form a basis for much of the remaining text. The emphasis has been to express the equations in a matrix form that is convenient for the finite element developments presented in subsequent chapters. While the equations are complete for the purposes needed in this volume the properties of their solution have not been described. The reader needs to be well aware of the limitations in the equations and basic properties of their solution. For these aspects the reader is referred to classical works on the subject. For example, some properties of solutions to elasticity problems may be found in the classical works of Love [1], Timoshenko and Goodier [3], or Muskhelishvili [2]. Basic solutions to the quasi-harmonic equation are typically covered for specific topic areas. Thus, for heat conduction problems one may consult the works of Carslaw and Jaeger [13]. A general reference for elliptic and hyperbolic problems is Lebedev and Silverman [14].

## 2.5 Problems

**2.1** For a plane stress problem the displacements have the polynomial expressions

$$u=A\,x^2y+B\,y^3+E\,y+G$$
$$v=C\,xy^2+D\,x^3+F\,x+H$$

**(a)** Compute the expressions for strain in matrix form.
**(b)** If the material is isotropic with properties $E$ and $\nu$ compute the expression for the stresses.
**(c)** Let

$$A = -\frac{P}{2EI},\ B = -\frac{\nu P}{6EI} + \frac{P}{6GI},\ C = \frac{\nu P}{2EI},\ D = \frac{P}{6EI}$$

and compute the expressions for the stresses. What problem can be solved?

**2.2** The axisymmetric solution for a thick-walled cylinder with inner radius $a$ and outer radius $b$ may be expressed as

$$u = A\,r + B\frac{1}{r},\ v = 0$$

**(a)** Obtain the expressions for strains.
**(b)** For an isotropic material with properties $E$ and $\nu$ obtain the expression for stresses assuming $\varepsilon_z = 0$.
**(c)** If the boundary conditions are $u(a) = 0$ and $t_r(b) = 1$ determine the values for $A$ and $B$ and plot the radial and tangential stress distributions.

**2.3** Using Eq. (2.12) show that $e_x + e_y + e_z = 0$ and, thus, does not produce volume change.
**2.4** Deduce a form for the complementary energy $U(\boldsymbol{\sigma})$ that produces Eq. (2.46b) using Eq. (2.49).
**2.5** A long axisymmetric thick-walled cylinder is subjected to a constant temperature $T_a$ at the inner radius $a$ and $T_b$ at the outer radius $b$.

**(a)** Let $\phi = T(r)$ independent of $z$ and determine the steady-state temperature distribution. Assume $Q = 0$ in (2.94).
**(b)** Compute the thermal strains from $T(r)$.
**(c)** Compute the thermal stress assuming a plane strain solution and isotropic material properties $E$, $\nu$, and $\alpha$. Assume the stress-free condition occurs at temperature $T_0$.

## References

[1] A.E.H. Love, A Treatise on the Mathematical Theory of Elasticity, Dover, New York, 2011.
[2] N.I. Muskhelishvili, Some Basic Problems of the Mathematical Theory of Elasticity, third ed., Noordhoff, Groningen, 1953 (English translation by J.R.M. Radok).
[3] S.P. Timoshenko, J.N. Goodier, Theory of Elasticity, third ed., McGraw-Hill, New York, 1970 (Classic Textbook Reissue Series).
[4] S.G. Lekhnitskii, Theory of Elasticity of an Anisotropic Elastic Body, Holden Day, San Francisco, 1963 (Translation from Russian by P. Fern).

[5] I.S. Sokolnikoff, The Mathematical Theory of Elasticity, second ed., McGraw-Hill, New York, 1956.
[6] P.G. Ciarlet, Mathematical Elasticity: Vol. 1, Three-Dimensional Elasticity, North-Holland, Amsterdam, 1988.
[7] O.C. Zienkiewicz, R.L. Taylor, D. Fox, The Finite Element Method for Solid and Structural Mechanics, seventh ed., Elsevier, Oxford, in press.
[8] T. Belytschko, W.K. Liu, B. Moran, Nonlinear Finite Elements for Continua and Structures, John Wiley & Sons, Chichester, 2000.
[9] O.C. Zienkiewicz, F.C. Scott, On the principle of repeatability and its application in analysis of turbine and pump impellers, Int. J. Numer. Methods Eng. 9 (1972) 445–452.
[10] S.P. Timoshenko, J.N. Goodier, Theory of Elasticity, third ed., McGraw-Hill, New York, 1969.
[11] R.F.S. Hearmon, An Introduction to Applied Anisotropic Elasticity, Oxford University Press, Oxford, 1961.
[12] T.C.-T Ting, Anisotropic Elasticity: Theory and Applications, Oxford University Press, New York, 1996.
[13] H.S. Carslaw, J.C. Jaeger, Conduction of Heat in Solids, second ed., Oxford University Press, Oxford, 1986 (Paperback edition).
[14] N.N. Lebedev, R.A. Silverman, Worked Problems in Applied Mathematics, Dover, New York, 2010.

This page is intentionally left blank

CHAPTER

# Weak Forms and Finite Element Approximation: 1-D Problems

3

## 3.1 Weak forms

In the preceding chapter we stated the governing equations for elasticity and field problems in terms of partial differential equations along with their boundary and, where appropriate, initial conditions. These describe the *continuous problem*. General solutions to problems formulated in this manner are seldom possible for situations of interest in science and engineering. Thus, one seeks to find other approaches that may be used. One approach is to recast the problem in an alternate form from which accurate approximate solutions may be achieved. In this chapter we consider one such approach called a *weak form* of the differential equations. In the following chapter we consider a related but alternate approach based on a *variational approach*. Later we show that the two approaches are closely related and often may be used interchangeably.

A weak form to a set of differential equations is obtained using the following steps:

- Multiply each equation by an appropriate *arbitrary* function.
- Integrate this product over the space domain of the problem.
- Use integration by parts to reduce the order of derivatives to a minimum. (3.1)
- Introduce boundary conditions if possible.

In this volume we only integrate over the space domain and, thus, it will be necessary to express time dependency and initial conditions explicitly. It is also possible to consider integrations over the time domain [1–3]; however, this is outside the scope of the present volume. The above steps to construct a weak form are direct and may be used to reduce any set of differential equations to a weak form. An arbitrary function is one that can take any value we can imagine. It can be a polynomial, a trigonometric function, a Heaviside function, a Dirac delta function, or any other function. A special case of an arbitrary function is a virtual displacement which may already be known to the reader.

To illustrate the above process we first consider a weak form for the one-dimensional equilibrium equation of elasticity.

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00003-4

## 3.2 One-dimensional form of elasticity

We first consider a one-dimensional problem in a Cartesian coordinate system with a single component $x$. The equation for linear elasticity may be deduced by writing the $u$-displacement as a function of the $x$-coordinate and time. We assume that our problem is defined in a domain $\Omega$, which is some interval $a < x < b$, and time $t \geq 0$. Accordingly, we write $u(x, t)$ for the displacement and observe that all strain components defined in the previous chapter are zero except

$$\varepsilon_x = \frac{\partial u}{\partial x} \tag{3.2}$$

For simplicity, to describe the stress-strain behavior we assume a Poisson ratio of zero. For this value all components of stress are zero except $\sigma_x$; thus, the equilibrium equations reduce to the single equation

$$\frac{\partial \sigma_x}{\partial x} + b_x = \rho \frac{\partial^2 u}{\partial t^2} \tag{3.3}$$

and the constitutive equation to

$$\sigma_x = E\, \varepsilon_x \tag{3.4}$$

where $E$ is Young's modulus of elasticity. The problem is completed by introducing boundary conditions as

$$u = \bar{u} \quad \text{or} \quad t_x = \bar{t}_x = n_x \sigma_x \quad \text{on } x = a, b \tag{3.5}$$

and initial conditions

$$u(x, 0) = \bar{d}(x) \quad \text{and} \quad \dot{u}(x, 0) = \bar{v}(x) \quad \text{in } \Omega \tag{3.6}$$

In Eq. (3.5) $n_x$ is the unit outward normal that is positive unity at $b$ and negative unity at $a$.

The above equations may be combined by substituting the constitutive equation and strain-displacement equation into the equilibrium equation. This gives the equilibrium equation expressed in terms of displacement as

$$\frac{\partial}{\partial x}\left(E \frac{\partial u}{\partial x}\right) + b_x = \rho \frac{\partial^2 u}{\partial t^2} \tag{3.7}$$

where dependence on $x$ of $E$ is permitted. We note this again is an irreducible form of the differential equation since only a single dependent variable exists.

The set of Eqs. (3.2)–(3.6) is called the *strong form* of the one-dimensional elasticity problem. It can be used to write a set of discrete equations using finite difference methods [4–7]. However, a more powerful approach is to introduce an integral *weak form* of the equations using the steps described in (3.1) from which an alternate approximate method may be introduced.

### 3.2.1 Weak form of equilibrium equation

We start by introducing an arbitrary function $w(x)$ that is defined in the domain $\Omega$ described by the interval $a < x < b$. Multiplying the equilibrium equation (3.3) by this function we may write

$$g(w,\, u,\, \sigma_x) = w(x)\left(\rho\frac{\partial^2 u}{\partial t^2} - b_x - \frac{\partial \sigma_x}{\partial x}\right) = 0 \tag{3.8}$$

It is convenient, though not necessary, that we write a form that is equal to zero. In this way we observe that if $w(x)$ is zero everywhere except at $x = x_c$ then for (3.8) to be zero the differential equation must be satisfied at $x = x_c$. Repeating for all points the expression says the differential equation is satisfied at all points in the domain provided $w(x)$ is arbitrary everywhere. When, in the second step of (3.1), we integrate over the domain we obtain an *integral form* that is also zero and is expressed by

$$\mathrm{G}(w,\, u,\, \sigma_x) = \int_\Omega w(x)\left(\rho\frac{\partial^2 u}{\partial t^2} - b_x - \frac{\partial \sigma_x}{\partial x}\right)\mathrm{d}x = 0 \tag{3.9}$$

This form is a function of functions called a *functional*. Repeating the argument on $w(x)$ being nonzero only at individual points again provides the necessary argument that the differential equation must be zero at all points in the domain. We call this the *fundamental lemma* for an integral form. As a third step, we can integrate the stress term by parts as

$$-\int_\Omega w(x)\frac{\partial \sigma_x}{\partial x}\,\mathrm{d}x = \int_\Omega \frac{\partial w}{\partial x}\,\sigma_x\,\mathrm{d}x - w(x)\,n_x\,\sigma_x\Big|_\Gamma$$

where $\Gamma$ is the boundary of $\Omega$ and $n_x$ is the outward pointing normal to the boundary. The boundary term may be expressed in terms of the traction as

$$\begin{aligned} w(x)\,n_x\,\sigma_x|_\Gamma &= w(b)\,\sigma_x(b) - w(a)\,\sigma_x(a) \\ &= w(b)\,t_x(b) + w(a)\,t_x(a) \end{aligned}$$

where we have noted $n_x(b) = 1$ and $n_x(a) = -1$. We also again introduce the notation that $\Gamma_u$ is a boundary where $u = \bar{u}$, $\Gamma_t$ is a boundary where $t_x = \bar{t}_x$, and the total boundary is $\Gamma = \Gamma_u \cup \Gamma_t$. With this notation we can write the *weak form* for equilibrium as

$$\begin{aligned} \mathrm{G}(w,\, u,\, \sigma_x) = &\int_\Omega w(x)\left(\rho\frac{\partial^2 u}{\partial t^2} - b_x\right)\mathrm{d}x \\ &+ \int_\Omega \frac{\partial w}{\partial x}\,\sigma_x\,\mathrm{d}x - w\,t_x\,\big|_{\Gamma_u} - w\,\bar{t}_x\big|_{\Gamma_t} = 0 \end{aligned} \tag{3.10}$$

Note that the time derivative is not reduced as we have not integrated over time. If we wish to show what equations are satisfied by the weak form, we merely remove all derivatives from the weight function $w$ using integrations by parts and recover the governing differential equation along with some boundary terms. The recovered

differential equation is called the *Euler equation* of the weak form and the recovered boundary conditions are called *natural boundary conditions*. Boundary conditions that are not natural are called *essential*. The reader can verify that (3.10) yields the result

$$\mathrm{G}(w,\, u,\, \sigma_x) = \int_\Omega w(x) \left( \rho \frac{\partial^2 u}{\partial t^2} - \frac{\partial \sigma_x}{\partial x} - b_x \right) \mathrm{d}x - w \left( \bar{t}_x - t_x \right)\big|_{\Gamma_t} = 0 \quad (3.11)$$

From this we conclude that a weak form may be written as a sum of all the governing equations in which appropriate weight functions are introduced. This obviously includes equations both in the domain, $\Omega$, and those on the boundary, $\Gamma$.

We can introduce the constitutive equation and strain displacement equation to obtain a purely *displacement or "irreducible" form* of the equations. Accordingly, we obtain

$$\mathrm{G}(w,\, u) = \int_\Omega w(x) \left( \rho \frac{\partial^2 u}{\partial t^2} - b_x \right) \mathrm{d}x + \int_\Omega \frac{\partial w}{\partial x} E \frac{\partial u}{\partial x} \mathrm{d}x - w \bar{t}_x\big|_{\Gamma_t} = 0 \quad (3.12)$$

For simplicity, where displacements are specified we set $w = 0$ to eliminate the need to consider $t_x$. This form is known as a *displacement model* or an *irreducible form*.[1] A *static* problem is one for which the inertia term is omitted. Integrating (3.12) by parts to recover the Euler differential equations gives

$$\mathrm{G}(w,\, u) = \int_\Omega w(x) \left[ \rho \frac{\partial^2 u}{\partial t^2} - \frac{\partial}{\partial x} \left( E \frac{\partial u}{\partial x} \right) - b_x \right] \mathrm{d}x - w \left( \bar{t}_x - t_x \right)\big|_{\Gamma_t} = 0 \quad (3.13)$$

Thus, we can observe that the traction boundary conditions are natural ones and the displacement ones are essential.

Comparing Eq. (3.12) with Eq. (3.7) or (3.13) we note that the weak form only requires a specification of first derivatives for $u$ and $w$ with respect to the spatial coordinate $x$, whereas the strong form needs second derivatives. The weak form also provides a basis to construct approximate solutions.

#### 3.2.1.1 Adjoint forms

If one considers the static problem without load and boundary terms and continues to integrate the weak form by parts to remove all derivatives on $u$, a differential equation acting on the arbitrary weight function is created. This differential equation is called the *adjoint* to the original equation. For the static problem expressed in (3.12) the adjoint equation is given by

$$\mathrm{G}_{\mathrm{stat}}(w,\, u) = -\int_\Omega u \frac{\partial}{\partial x} \left( E \frac{\partial w}{\partial x} \right) \mathrm{d}x$$

and since it is exactly the same as the original static differential equation on $u$ [viz. (3.13)], the equation is called *self-adjoint*.

[1] When more variables are included in the weak form we shall call the method *mixed*.

Indeed, any linear differential equation $\mathcal{L}(u)$ with weak form written as

$$G_{\text{stat}}(w,\, u) = \int_\Omega w\, \mathcal{L}(u)\, \mathrm{d}x = \int_\Omega u\, \hat{\mathcal{L}}(w)\, \mathrm{d}x \quad \text{with} \quad \mathcal{L} = \hat{\mathcal{L}}$$

is called self-adjoint. Self-adjoint differential equations have useful properties that we will describe later.

## 3.3 Approximation to integral and weak forms: The weighted residual (Galerkin) method

To construct an approximate solution we express the displacement $u(x,\, t)$ in terms of a set of specified functions multiplied by unknown parameters. In a similar way, we write the arbitrary function in terms of an equal number of specified functions and arbitrary parameters. These may be expressed as

$$\begin{aligned} u(x,\, t) \approx \hat{u}(x,\, t) &= \sum_{n=1}^{N} \phi_n(x)\, a_n(t) + u_{\bar{b}}(x,\, t) \\ w(x,\, t) \approx \hat{w}(x) &= \sum_{m=1}^{N} \psi_m(x)\, w_m \end{aligned} \tag{3.14}$$

In the above form we assume that both $\phi_n(x)$ and $\psi_m(x)$ are zero at all locations where the boundary displacement is specified. The function $u_{\bar{b}}(x,\, t)$ is then specified as any function that satisfies the displacement boundary condition. For example, if the displacement must satisfy $u(L,\, t) = \bar{d}(t)$ on the domain $0 < x < L$, this function may be taken as

$$u_{\bar{b}}(x,\, t) = \frac{x}{L}\, \bar{d}(t) = \phi_{\bar{b}}(x)\, \bar{d}(t)$$

There are many ways that approximations may be assumed. These include:

1. *Point collocation* [8]. $\psi_m = \delta(x_m)$ in (3.12), where $\delta(x_m)$ is such that for $x \neq x_m$, $\psi_m = 0$ but $\int_\Omega \psi_m \mathrm{d}x = 1$. This procedure is equivalent to simply making the weak form zero at $N$ points within the domain and integration is "nominal". Finite difference methods are particular cases of this weighting. For point approximation of $w(x)$ the approximation for $u(x,\, t)$ must have continuous derivatives for all terms in the integral form. For (3.13) this requires that second derivatives be continuous at every $x_m$.
2. *Subdomain collocation* [9]. $\psi_m = 1$ in subdomains $x_m < x < x_{m+1}$ and zero elsewhere. This essentially makes the integral of the error zero over the specified subdomains. When used with (3.9) this is one of the many finite volume methods [10]. The requirements for approximations of $u(x,\, t)$ permit use of functions whose highest derivatives in the integral form are piecewise continuous. For (3.13) this implies that the $\phi_n(x)$ have continuous second derivatives.

3. *The Galerkin method* (Bubnov-Galerkin) [9,11]. $\psi_n = \phi_n$. This form allows use of approximating functions with piecewise continuous highest derivatives in the *weak form*. For (3.12) we only need the $\phi_n$ to be continuous. This method, as we shall see, frequently (but by no means always) leads to symmetric matrices and for this and other reasons will be adopted in this volume almost exclusively.

The name "weighted residuals" is clearly much older than that of the "finite element method." The latter uses mainly locally based (element) functions in the expansion of Eq. (3.14) but the general procedures are identical. As the process always leads to equations which, being of integral form, can be obtained by summation of contributions from various subdomains, we choose to embrace all weighted residual approximations under the name *generalized finite element method*. On occasion, simultaneous use of both local and "global" trial functions may be found to be useful.

In the literature the names Petrov and Galerkin [11] are often associated with the use of weighting functions such that $\psi_n \neq \phi_n$. It is important to remark that the well-known *finite difference method* of approximation is a particular case of collocation with locally defined basis functions and is thus a case of a Petrov-Galerkin scheme.

### 3.3.1 Galerkin solution of elasticity equation

An approximate Galerkin solution for the elasticity problem may be obtained by inserting (3.14) into (3.12) to obtain

$$\begin{aligned}\hat{\mathrm{G}}(\hat{w},\,\hat{u}) = &\sum_{m=1}^{N} w_m \sum_{n=1}^{N} \int_\Omega \psi_m\,\rho \left[\phi_n \frac{\mathrm{d}^2 a_n}{\mathrm{d}t^2} + \phi_{\bar{b}} \frac{\mathrm{d}^2 \bar{d}}{\mathrm{d}t^2}\right] \mathrm{d}x \\ &+ \sum_{m=1}^{N} w_m \sum_{n=1}^{N} \int_\Omega \frac{\mathrm{d}\psi_m}{\mathrm{d}x}\,E \left[\frac{\mathrm{d}\phi_n}{\mathrm{d}x}\,a_n(t) + \frac{\mathrm{d}\phi_{\bar{b}}}{\mathrm{d}x}\,\bar{d}(t)\right] \mathrm{d}x \\ &- \sum_{m=1}^{N} w_m \int_\Omega \psi_m\,b_x\,\mathrm{d}x - \sum_{m=1}^{N} w_m\,\psi_m(x)\,\bar{t}_x\big|_{\Gamma_t} = 0\end{aligned} \tag{3.15}$$

Since the functions $\phi_n$, $\psi_n$, and $\phi_{\bar{b}}$ are all known functions, the integrals may be evaluated as

$$\begin{aligned}M_{mn} &= \int_\Omega \psi_m\,\rho\,\phi_n\,\mathrm{d}x \\ K_{mn} &= \int_\Omega \frac{\mathrm{d}\psi_m}{\mathrm{d}x}\,E\,\frac{\mathrm{d}\phi_n}{\mathrm{d}x}\,\mathrm{d}x \\ f_m(t) &= \int_\Omega \psi_m\,b_x\,\mathrm{d}x + \psi_m\,\bar{t}_x\big|_{\Gamma_t} - \int_\Omega \psi_m\,\rho\,\phi_{\bar{b}}\,\mathrm{d}x\,\frac{\mathrm{d}^2\bar{d}}{\mathrm{d}t^2} \\ &\quad - \int_\Omega \frac{\mathrm{d}\psi_m}{\mathrm{d}x}\,E\,\frac{\mathrm{d}\phi_{\bar{b}}}{\mathrm{d}x}\,\mathrm{d}x\,\bar{d}(t)\end{aligned} \tag{3.16}$$

The array $M_{mn}$ is called the *mass matrix*, the array $K_{mn}$ is the *stiffness matrix*, and $f_m$ is the *specified load matrix*. Since the parameters $w_m$ are arbitrary, the expression multiplying each one must be zero. This leads to the set of equations

$$\sum_{n=1}^{N}\left[M_{mn}\,\frac{\mathrm{d}^2 a_n}{\mathrm{d}t^2}+K_{mn}\,a_n(t)\right]=f_m(t),\quad m=1,2,\ldots,N \tag{3.17}$$

The original problem of partial differential equations has been reduced to a set of ordinary differential equations in time. For the present we will consider problems in which inertial effects are not important; transient problems are considered in Section 3.8. Problems without inertia effects are called *static*. The equations are now purely algebraic and given by

$$\sum_{n=1}^{N}K_{mn}\,a_n=f_m,\quad m=1,2,\ldots,N \tag{3.18}$$

where the load matrix reduces to

$$f_m=\int_\Omega \psi_m\,b_x\,\mathrm{d}x+\psi(x)\,\bar{t}_x\big|_{\Gamma_t}-\int_\Omega\frac{\mathrm{d}\psi_m}{\mathrm{d}x}\,E\,\frac{\mathrm{d}\phi_{\bar{b}}}{\mathrm{d}x}\,\mathrm{d}x\,\bar{d} \tag{3.19}$$

Expressing the problem in matrix form

$$\mathbf{K}\,\mathbf{a}=\mathbf{f} \tag{3.20}$$

where

$$\mathbf{K}=\begin{bmatrix}K_{11}&K_{12}&\cdots&K_{1N}\\K_{21}&K_{22}&\cdots&K_{2N}\\\vdots&&&\vdots\\K_{N1}&K_{N2}&\cdots&K_{NN}\end{bmatrix}\quad\text{and}\quad\mathbf{f}=\begin{Bmatrix}f_1\\f_2\\\vdots\\f_N\end{Bmatrix}$$

The formal solution to the problem is now given by

$$\mathbf{a}=\mathbf{K}^{-1}\mathbf{f} \tag{3.21}$$

However, in practical situations the inverse of $\mathbf{K}$ is never determined. Instead the solution is obtained using either a direct solution of (3.20) or by an iterative method (see Appendix D).

Once the parameters $a_n$ are known, the approximation for displacement may be computed from (3.14) and that for stresses from

$$\sigma(x)=E\sum_{n=1}^{N}\frac{\mathrm{d}\phi_n}{\mathrm{d}x}\,a_n$$

**Example 3.1.** Static solution

As an example we consider a static problem with length 10 units and $E=1000$. Figure 3.1 shows the problem to solve. The loading is given by

$$b_x=\begin{cases}10 & \text{for } 0<x<5\\ 0 & \text{for } 5<x<10\end{cases}$$

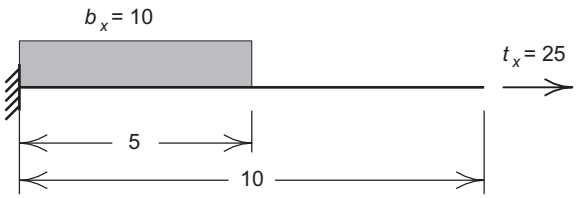



**FIGURE 3.1**
One-dimensional elasticity example.

and the boundary conditions by

$$\bar{t}_x(10) = 25 \quad \text{and} \quad u(0) = 0$$

For this problem the weak form is given by

$$\int_0^{10} \frac{\partial w}{\partial x}\, 1000\, \frac{\partial u}{\partial x}\, \mathrm{d}x - \int_0^5 w(x)\, 10\, \mathrm{d}x - w(10)\, 25 = 0$$

We consider an approximate solution given by

$$\hat{u}(x) = \sum_{n=1}^{N} \left(\frac{x}{10}\right)^n a_n \quad \text{and} \quad \hat{w}(x) = \sum_{m=1}^{N} \left(\frac{x}{10}\right)^m w_m$$

Note that $u_{\bar{b}} = 0$ for this problem. The solution to the weak form is obtained from

$$\sum_{n=1}^{N} \left[\int_0^{10} 10\, mn \left(\frac{x}{10}\right)^{m-1} \left(\frac{x}{10}\right)^{n-1} \mathrm{d}x\right] a_n = \int_0^5 \left(\frac{x}{10}\right)^m 10\, \mathrm{d}x + 25$$

where $m = 1, 2, \ldots, N$. The stiffness matrix and load vector are given by

$$K_{mn} = \int_0^{10} 10\, mn \left(\frac{x}{10}\right)^{m+n-2} \mathrm{d}x = \frac{100\, mn}{m+n-1}$$

and

$$f_m = \int_0^5 \left(\frac{x}{10}\right)^m 10\, \mathrm{d}x + 25 = \frac{100}{m+1}\left(\frac{1}{2}\right)^{m+1} + 25 \tag{3.22}$$

The solution for the parameters using one to five terms in the solution is shown in Table 3.1.

The results from the above problem indicate some important aspects in constructing approximate solutions. First it is noticed that the solutions using three and four terms are identical. This implies that the weak form is able to discard terms that do not contribute to the solution. A second conclusion, however, is that constructing solutions in a global manner does not lead to convergent behavior in the individual parameters. Thus, it is essential to use approximation procedures that have more stable properties. We will observe that finite element methods do possess such good properties. The solution for displacement and stress obtained using one to three terms is

**Table 3.1** Parameters for Bar Example

| *N*-terms | $a_1$ | $a_2$ | $a_3$ | $a_4$ | $a_5$ |
|---|---|---|---|---|---|
| 1 | 0.37500 | | | | |
| 2 | 0.62500 | −0.25000 | | | |
| 3 | 0.78125 | −0.71875 | 0.31250 | | |
| 4 | 0.78125 | −0.71875 | 0.31250 | 0.00000 | |
| 5 | 0.73437 | −0.25000 | −1.09275 | 1.64063 | 0.65625 |

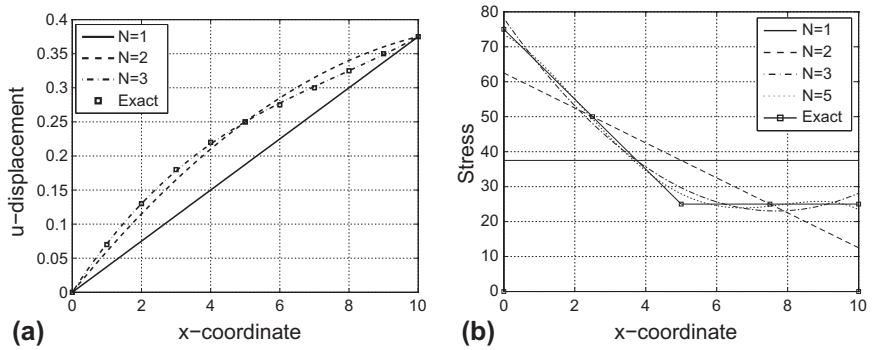



**FIGURE 3.2**
Displacement and stress for bar using *N*-term solutions: (a) displacement and (b) stress.

shown in Fig. 3.2. From this figure we observe that the displacement at the free end is the same no matter how many terms we use. This often happens in one-dimensional static problems but, unfortunately, is seldom true in higher dimensional problems. The solution for stress converges more slowly than that for the displacements; however, once again we observe that some points are more accurate than others. These we shall call *super-convergent* points and these points play an important role in our later discussion on error estimates and adaptive refinement of solutions.

## 3.4 Finite element solution

A more convenient method to construct the approximating functions $\psi_m$ and $\phi_n$ is obtained by dividing the domain to be analyzed into small regular shaped regions. For example, we can divide the one-dimensional region between $a$ and $b$ into a set of "$M$" small finite segments by defining a set of $N$ points $x_i$ such that

$$x_1 = a, \quad x_i < x_{i+1} \quad \text{and} \quad x_N = b$$

For a one-dimensional problem we can let each increment define a *finite element* domain (or more simply, an *element*) and the set of points define the *nodes*.

The division into elements and nodes is a fundamental part of the finite element method and describes what we will refer to as the *finite element mesh* or simply the *mesh* for the problem.

Using the above subdivision, a simple set of continuous polynomial approximating functions may be defined by

$$\phi_i = \begin{cases} 0, & x < x_{i-1} \\ \dfrac{x - x_{i-1}}{x_i - x_{i-1}}, & x_{i-1} \le x \le x_i \\ \dfrac{x_{i+1} - x}{x_{i+1} - x_i}, & x_i < x \le x_{i+1} \\ 0, & x > x_{i+1} \end{cases} \tag{3.23}$$

A plot for these functions and their first derivative is shown in Fig. 3.3. Such functions are called $C_0$ since only the function is continuous in $x$, whereas the first derivative is only piecewise continuous with the discontinuities located at the nodes. Development of appropriate $C_0$ functions for higher spatial dimensions will be considered in subsequent chapters. Here we shall focus on how to describe general forms in *one dimension*.

From the figure we can note that, if required, the end functions can serve as the spatial form for the $\phi_{\bar{b}}(x)$ functions.

If we let $\psi_i = \phi_i$ then we observe that all the integrals defined in the weak form functional can be integrated over the length. Indeed, we usually can evaluate the integrals individually over each element by noting

$$\int_\Omega (\cdot)\, \mathrm{d}x = \sum_{i=1}^{M} \int_{x_i}^{x_{i+1}} (\cdot)\, \mathrm{d}x \equiv \sum_e \int_{\Omega_e} (\cdot)\, \mathrm{d}x \tag{3.24}$$

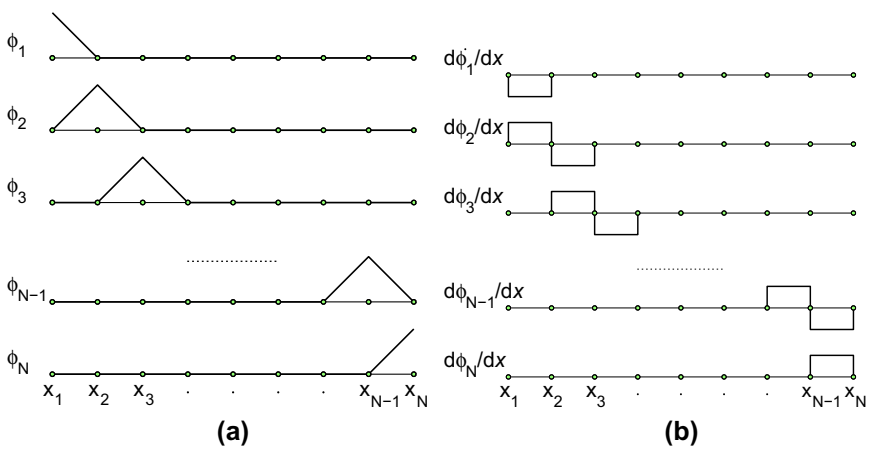



**FIGURE 3.3**
One-dimensional finite element approximation for $\phi_i$: (a) functions and (b) derivatives.

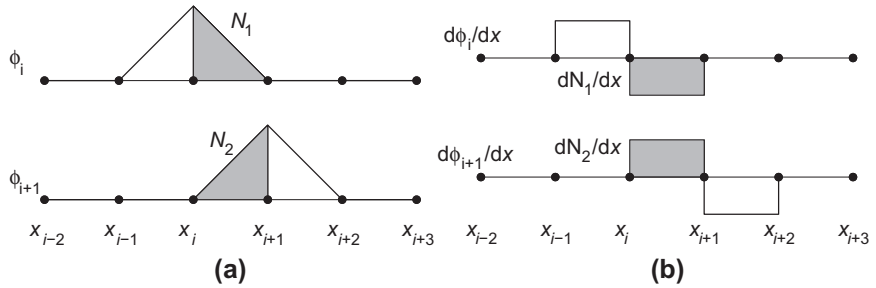



**FIGURE 3.4**
One-dimensional finite element shape functions: (a) functions and (b) derivatives.

**Table 3.2** Local to Global Node Numbering for Two-Node Elements

| Local Node | Element Number | | | | |
|---|---|---|---|---|---|
| | 1 | 2 | 3 | ··· | M |
| 1 | 1 | 2 | 3 | ··· | $N-1$ |
| 2 | 2 | 3 | 4 | ··· | $N$ |

Considering any interval $[x_i,\ x_{i+1}]$ shown in Fig. 3.4 we note that each interval is defined by the same two local functions $N_1$ and $N_2$. We call these the *shape functions* for the element. To simplify the notation we also define local nodal coordinates on each element as $x_1^e$ and $x_2^e$. Then for each element we define the relationship of the local nodes to the global node number as indicated in Table 3.2.

With this notation we can always write the displacements and arbitrary weight function as

$$\begin{aligned} \hat{u}^e &= N_1(x')\,\tilde{u}_1^e + N_2(x')\,\tilde{u}_2^e \\ \hat{w}^e &= N_1(x')\,\tilde{w}_1^e + N_2(x')\,\tilde{w}_2^e \end{aligned} \tag{3.25}$$

If we define a local coordinate system in each element as $x' = x - x_1^e$ and the element length as $h_e = x_2^e - x_1^e$, the shape functions are given by

$$N_1(x') = 1 - \frac{x'}{h_e} \quad \text{and} \quad N_2(x') = \frac{x'}{h_e} \tag{3.26a}$$

and are the same for every element. The derivatives of the shape functions are given by

$$\frac{\mathrm{d}N_1}{\mathrm{d}x} = \frac{\mathrm{d}N_1}{\mathrm{d}x'} = -\frac{1}{h_e} \quad \text{and} \quad \frac{\mathrm{d}N_2}{\mathrm{d}x} = \frac{\mathrm{d}N_2}{\mathrm{d}x'} = \frac{1}{h_e} \tag{3.26b}$$

Using the above we can write the approximation to the weak form as

$$\widehat{\mathrm{G}}(\hat{w},\ \hat{u}) = \widehat{\mathrm{G}}_i(\hat{w},\ \hat{u}) + \widehat{\mathrm{G}}_\sigma(\hat{w},\ \hat{u}) - \widehat{\mathrm{G}}_f(\hat{w},\ \hat{u}) - \hat{w}(x)\bar{t}_x\big|_{\Gamma_t} \tag{3.27a}$$

where $\widehat{\mathrm{G}}_i$ denotes the inertial contribution, $\widehat{\mathrm{G}}_\sigma$ denotes the internal stress contribution, and $\widehat{\mathrm{G}}_f$ denotes the contribution to the force from the body loading. Using (3.12) each

term is defined by

$$\widehat{\mathrm{G}}_i(\hat{w},\,\hat{u}) = \sum_{e=1}^{M} \left[\,\tilde{w}_1^e \quad \tilde{w}_2^e\,\right] \int_0^{h_e} \left\{ \begin{matrix} N_1 \\ N_2 \end{matrix} \right\} \rho_e \left[\,N_1 \quad N_2\,\right] \mathrm{d}x' \left\{ \begin{matrix} \ddot{\tilde{u}}_1^e \\ \ddot{\tilde{u}}_2^e \end{matrix} \right\}$$

$$\widehat{\mathrm{G}}_\sigma(\hat{w},\hat{u}) = \sum_{e=1}^{M} \left[\,\tilde{w}_1^e \quad \tilde{w}_2^e\,\right] \int_0^{h_e} \left\{ \begin{matrix} \dfrac{\mathrm{d}N_1}{\mathrm{d}x'} \\ \dfrac{\mathrm{d}N_2}{\mathrm{d}x'} \end{matrix} \right\} E_e \left[ \begin{matrix} \dfrac{\mathrm{d}N_1}{\mathrm{d}x'} & \dfrac{\mathrm{d}N_2}{\mathrm{d}x'} \end{matrix} \right] \mathrm{d}x' \left\{ \begin{matrix} \tilde{u}_1^e \\ \tilde{u}_2^e \end{matrix} \right\} \tag{3.27b}$$

$$\widehat{\mathrm{G}}_f(\hat{w},\hat{u}) = \sum_{e=1}^{M} \left[\,\tilde{w}_1^e \quad \tilde{w}_2^e\,\right] \int_0^{h_e} \left\{ \begin{matrix} N_1 \\ N_2 \end{matrix} \right\} b_x \,\mathrm{d}x'$$

where $\dot{u} = \mathrm{d}u/\mathrm{d}t$.

Each element can be evaluated as

$$\mathbf{M}^e = \int_0^{h_e} \left\{ \begin{matrix} N_1 \\ N_2 \end{matrix} \right\} \rho_e \left[\,N_1 \quad N_2\,\right] \mathrm{d}x' = \begin{bmatrix} M_{11}^e & M_{12}^e \\ M_{21}^e & M_{22}^e \end{bmatrix}$$

$$\mathbf{K}^e = \int_0^{h_e} \left\{ \begin{matrix} \dfrac{\mathrm{d}N_1}{\mathrm{d}x'} \\ \dfrac{\mathrm{d}N_2}{\mathrm{d}x'} \end{matrix} \right\} E \left[ \begin{matrix} \dfrac{\mathrm{d}N_1}{\mathrm{d}x'} & \dfrac{\mathrm{d}N_2}{\mathrm{d}x'} \end{matrix} \right] \mathrm{d}x' = \begin{bmatrix} K_{11}^e & K_{12}^e \\ K_{21}^e & K_{22}^e \end{bmatrix} \tag{3.28}$$

$$\mathbf{f}^e = \int_0^{h_e} \left\{ \begin{matrix} N_1 \\ N_2 \end{matrix} \right\} b_x \,\mathrm{d}x' = \left\{ \begin{matrix} f_1^e \\ f_2^e \end{matrix} \right\}$$

For the shape functions given in Eq. (3.26a) and assuming $E_e$, $\rho_e$, and $b_x$ are constant within each element the matrices are given by

$$\mathbf{M}^e = \frac{1}{6}\rho_e h_e \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \quad \mathbf{K}^e = \frac{E_e}{h_e} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{f}^e = \frac{1}{2} b_x h_e \left\{ \begin{matrix} 1 \\ 1 \end{matrix} \right\} \tag{3.29}$$

The summation indicated in (3.24) and subsequent expressions leads to a *standard linear problem* with the final equations given by

$$\mathbf{M}\,\ddot{\tilde{\mathbf{u}}} + \mathbf{K}\,\tilde{\mathbf{u}} = \mathbf{f} \tag{3.30}$$

For a static problem the inertia term may be ignored and the expanded form of the matrices is given by

$$\begin{bmatrix} K_{11}^1 & K_{12}^1 & 0 & & \\ K_{21}^1 & (K_{22}^1 + K_{11}^2) & K_{12}^2 & 0 & \\ 0 & K_{21}^2 & (K_{22}^2 + K_{11}^3) & K_{12}^3 & \\ & & & \ddots & \\ & 0 & K_{21}^{M-1} & (K_{22}^{M-1} + K_{11}^M) & K_{12}^M \\ & & 0 & K_{21}^M & K_{22}^M \end{bmatrix} \left\{ \begin{matrix} \tilde{u}_1 \\ \tilde{u}_2 \\ \tilde{u}_3 \\ \vdots \\ \tilde{u}_{N-1} \\ \tilde{u}_N \end{matrix} \right\} = \left\{ \begin{matrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_{N-1} \\ f_N \end{matrix} \right\} \tag{3.31}$$

Up to this point we have not considered boundary conditions other than noting the end functions can serve as the boundary approximation. With the above finite element form it is very simple to impose the displacement boundary conditions since the parameters are now all physical values. That is they obey the property

$$\hat{u}(x_a) \equiv \tilde{u}_a \tag{3.32}$$

Thus, if we wish to impose a displacement condition $u(0) = \bar{u}$, we set $\tilde{u}_1 = \bar{u}$ and rewrite (3.31) as

$$\begin{bmatrix} 1 & 0 & 0 & & \\ 0 & (K^1_{22} + K^2_{11}) & K^2_{12} & 0 & \\ 0 & K^2_{21} & (K^2_{22} + K^3_{11}) & K^3_{12} & \\ & & & \ddots & \\ & 0 & K^{M-1}_{21} & (K^{M-1}_{22} + K^M_{11}) & K^M_{12} \\ & & 0 & K^M_{21} & K^M_{22} \end{bmatrix} \begin{Bmatrix} \tilde{u}_1 \\ \tilde{u}_2 \\ \tilde{u}_3 \\ \vdots \\ \tilde{u}_{N-1} \\ \tilde{u}_N \end{Bmatrix} = \begin{Bmatrix} \bar{u} \\ f_2 - K^1_{21}\bar{u} \\ f_3 \\ \vdots \\ f_{N-1} \\ f_N \end{Bmatrix} \tag{3.33}$$

This is equivalent to setting $\tilde{w}_1 = 0$. Similarly, imposing a traction condition at $x = L$ only requires the modification

$$f_N \to f_N + \bar{t}_x(L) \tag{3.34}$$

These physical properties point out the distinct advantage of using a finite element form to approximate the variables.

**Example 3.2.** Static solution by finite element method

We again consider the solution of the example problem previously considered and shown in Fig. 3.1. We divide the domain into four equal elements as shown in Fig. 3.5 with the solution shown in Fig. 3.6. The solution using eight elements is also shown in the figure. Comparing the two solutions, we note that the stresses are converging more slowly than the displacements (this was true for the first example also) but again have super-convergent points.

### 3.4.1 Requirements for finite element approximations

The construction of finite element approximations for the basis functions, $\phi_i(x)$, is quite easy to express. Indeed there are only three basic requirements:

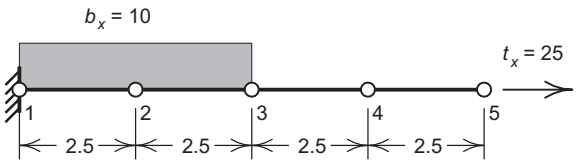



**FIGURE 3.5**
Four-element mesh for one-dimensional elasticity example.

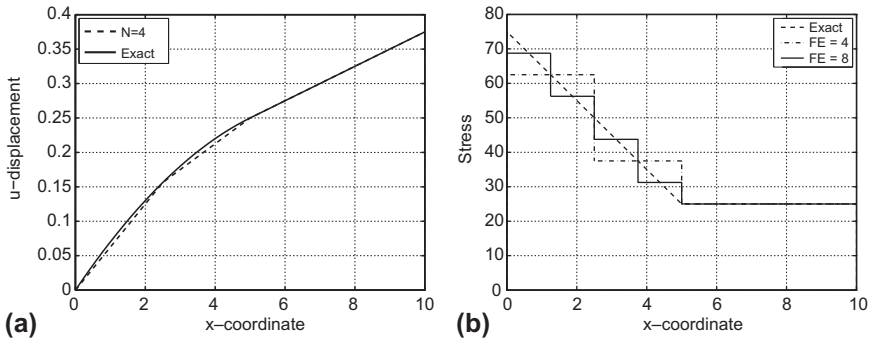



**FIGURE 3.6**
One-dimensional finite element solution to bar problem: (a) displacements and (b) stress.

1. *Continuity:* Each function must be continuous in the problem domain up to one less than the highest derivative in the functional $G(w, u)$.
2. *Completeness:* Adding terms to the solutions (i.e., increasing the number of $\phi_i$ and $\psi_i$) must be such that a suitable choice of the parameters be able to match any bounded solution and its derivative more closely as terms are added. That is,

$$\left|u - \hat{u}|_N\right| < O(h_e^p) \quad \text{and} \quad \left|\frac{\mathrm{d}u}{\mathrm{d}x} - \frac{\mathrm{d}\hat{u}}{\mathrm{d}x}\,|_N\right| < O(h_e^{p-1})$$

   as $N$ is increased.
3. *Linear independence:* The functions must be linearly independent, that is, no combination of some functions may give another of the functions.

It is clear that the simple functions defined in (3.23) satisfy all the above requirements for the one-dimensional elasticity problem. They are continuous and linearly independent as seen in Figs. 3.3 and 3.4. It is also clear that short piecewise continuous straight segments can match any bounded curve as closely as we want by reducing the segment lengths. In particular, linear functions can satisfy the condition $u = const$, which is a rigid translation or rigid body mode of the solution. This condition is crucial to all problems formulated in Cartesian coordinates since any constant displacement causes no strain. If strains were present, we would not be able to match the above

derivative condition. Indeed by using polynomials complete to degree $p$ to define the shape functions we can always satisfy the completeness condition for problems requiring $C_0$ continuity provided $p$ is at least of degree 1.

## 3.5 Isoparametric form

The element displacement function for the two-node element was obtained as

$$\hat{u}^e(x,\, t) = N_1(x')\, \tilde{u}_1^e(t) + N_2(x')\, \tilde{u}_2^e(t)$$

where $x = x_1^e + x'$ and the shape functions are expressed by

$$N_1(x') = 1 - \frac{x'}{h^e} \quad \text{and} \quad N_2(x') = \frac{x'}{h^e}$$

If we let

$$\frac{x'}{h^e} = \xi \quad \text{where} \quad 0 \le \xi \le 1$$

the shape functions may be rewritten as

$$N_1(\xi) = 1 - \xi \quad \text{and} \quad N_2(\xi) = \xi, \quad 0 \le \xi \le 1 \tag{3.35a}$$

This permits the writing of the element coordinate, displacement, and arbitrary function in the parametric form

$$\begin{aligned} x^e &= N_1(\xi) x_1^e + N_2(\xi) x_2^e \\ \hat{u}^e &= N_1(\xi) \tilde{u}_1^e + N_2(\xi) \tilde{u}_2^e \\ \hat{w}^e &= N_1(\xi) \tilde{w}_1^e + N_2(\xi) \tilde{w}_2^e \end{aligned} \tag{3.35b}$$

This is called an *isoparametric* form since all three expressions are identical. In this form the coordinate $\xi$ is a parameter called the *parent coordinate*.

The computation of $x$ derivatives now requires use of the chain rule which for any function $f(\xi)$ may be written as

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial \xi} \frac{\partial \xi}{\partial x} \tag{3.36}$$

If we let $f(\xi) = x$ then we obtain

$$\frac{\partial x}{\partial x} = 1 = \frac{\partial x}{\partial \xi} \frac{\partial \xi}{\partial x}$$

The term $\partial x / \partial \xi$ is called the *Jacobian* and is easily computed from the shape functions as

$$j_e = \frac{\partial x^e}{\partial \xi} = x_2^e - x_1^e = h_e$$

and, thus, for the linear shape functions

$$\frac{\partial \xi}{\partial x} = \frac{1}{h_e}$$

The derivative of the shape functions may now be computed as

$$\frac{\partial N_a}{\partial x} = \frac{\partial N_a}{\partial \xi}\frac{\partial \xi}{\partial x} = \frac{1}{h_e}\frac{\partial N_a}{\partial \xi}, \quad a = 1,\ 2$$

### 3.5.1 Higher order elements: Lagrange interpolation

Using the isoparametric concept the development of higher order approximations for an element now can be easily constructed. In general we shall write the approximation as

$$u^e \approx \hat{u}^e = \sum_a^n N_a(\xi)\,\tilde{u}_a \tag{3.37}$$

where $n$ is the total number of functions used in the element and $\tilde{u}_a$ are the unknown parameters to be determined. The requirement to satisfy Eq. (3.32) is that

$$N_a(\xi_b) = \delta_{ab} = \begin{cases} 1, & \xi_a = \xi_b \\ 0, & \xi_a \neq \xi_b \end{cases} \tag{3.38}$$

where $\xi_a$ is the parent coordinate of node $a$ which has position $x_a$.

A simple construction for any order we desire is the Lagrange interpolation formula given by

$$\begin{aligned} l_a^p(\xi) &= \prod_{\substack{b=1 \\ b\neq a}}^{n} \frac{(\xi - \xi_b)}{(\xi_a - \xi_b)} \\ &= \frac{(\xi - \xi_1)(\xi - \xi_2)(\cdots)(\xi - \xi_{a-1})(\xi - \xi_{a+1})(\cdots)(\xi - \xi_n)}{(\xi_a - \xi_1)(\xi_a - \xi_2)(\cdots)(\xi_a - \xi_{a-1})(\xi_a - \xi_{a+1})(\cdots)(\xi_a - \xi_n)} \end{aligned} \tag{3.39}$$

where the order of the polynomial is $p = n - 1$. In this form we can also pick any range we want for the domain of $\xi$. In the linear example above we set $\xi_1 = 0$ and $\xi_2 = 1$ with $n = 2$. A more convenient choice is to choose $\xi_1 = -1$ and $\xi_2 = 1$. The reason for this choice will become clear later when we describe the use of numerical integration (or quadrature) to evaluate the integrals. Once we choose the location for the end points the internal values of $\xi_a$ may be spaced at uniform increments.

For one-dimensional elements we can set

$$N_a(\xi) = l_a^p(\xi) \tag{3.40}$$

to define the shape functions.

Using an isoparametric form in terms of parent coordinates ensures that the global polynomial $x$ is always available. This may be shown by letting $\tilde{u}_a = \alpha\, x_a$ and then

$$u(\xi) = \sum_a N_a(\xi)\,\alpha\, x_a = \alpha \sum_a N_a(\xi)\, x_a = \alpha\, x \tag{3.41}$$

The completeness condition then requires that $u$ contain any constant $c$ which then yields

$$u(\xi) = \sum_a N_a(\xi)\, c = c$$

or

$$\sum_a N_a(\xi) = 1 \tag{3.42}$$

Thus, for any isoparametric shape functions that we may devise, it is only necessary to check whether (3.42) is satisfied and that the functions are continuous across each point on the element boundary.

#### 3.5.1.1 Linear shape functions

Setting $\xi_1 = -1$ and $\xi_2 = 1$, from (3.39) we obtain for $p = 1$ the two shape functions

$$N_1(\xi) = l_1^1(\xi) = \frac{\xi - 1}{-1 - 1} = \frac{1}{2}(1 - \xi) \quad \text{and} \quad N_2(\xi) = l_2^1(\xi) = \frac{\xi + 1}{1 + 1} = \frac{1}{2}(1 + \xi)$$

The Jacobian for this two-node linear element is given by

$$j_e(\xi) = \frac{\partial x}{\partial \xi} = \frac{1}{2}\,(x_2^e - x_1^e) = \frac{1}{2}\,h_e$$

The derivatives of the shape functions are given by

$$\frac{\partial N_a}{\partial x} = \frac{2}{h_e}\,\frac{\partial N_a}{\partial \xi}, \quad a = 1,\ 2$$

#### 3.5.1.2 Quadratic shape functions

For quadratic shape functions we let the nodes be placed at

$$\xi_1 = -1, \quad \xi_2 = 1, \quad \text{and} \quad \xi_3 = 0$$

and obtain the three shape functions as

$$\begin{aligned} N_1(\xi) = l_1^2(\xi) &= \frac{(\xi - 1)(\xi - 0)}{(-1 - 1)(-1 - 0)} = \frac{1}{2}\xi\,(\xi - 1) \\ N_2(\xi) = l_2^2(\xi) &= \frac{(\xi + 1)(\xi - 0)}{(1 + 1)(1 - 0)} = \frac{1}{2}\xi\,(\xi + 1) \\ N_3(\xi) = l_3^2(\xi) &= \frac{(\xi + 1)(\xi - 1)}{(0 + 1)(0 - 1)} = 1 - \xi^2 \end{aligned} \tag{3.43}$$

Note we have placed the local node 3 in the interior of the element. It is sometimes convenient to let node 1 and node 2 be the boundary nodes of the element with any interior nodes numbered as node 3 and higher.

If we let the coordinates for the element be given by $x_1^e,\ x_2^e,\ x_3^e$, the parametric form for the physical element is given by

$$x = N_1(\xi)\, x_1^e + N_2(\xi)\, x_2^e + N_3(\xi)\, x_3^e$$

The Jacobian is now given by

$$\begin{aligned} j_e(\xi) = \frac{\partial x}{\partial \xi} &= \left(\xi - \frac{1}{2}\right) x_1^e + \left(\xi + \frac{1}{2}\right) x_2^e - 2\,\xi\, x_3^e \\ &= \frac{1}{2} h_e + \xi\, \left(x_1^e + x_2^e - 2\, x_3^e\right) \end{aligned} \tag{3.44}$$

and we note that the Jacobian is not constant unless the coordinate for node 3 is placed at the middle of the element. Furthermore, for a valid isoparametric mapping, the Jacobian must be positive for all values of $\xi$. This restricts the location for the center node to satisfy

$$x_3^e - x_1^e = \alpha\, (x_2^e - x_1^e), \quad \frac{1}{4} < \alpha < \frac{3}{4} \tag{3.45}$$

A general form for the one-dimensional shape function derivative is given by the rational expression

$$\frac{\partial N_a}{\partial x} = \frac{1}{j_e(\xi)} \frac{\partial N_a}{\partial \xi}, \quad a = 1,\ 2,\ \ldots,\ n$$

### 3.5.2 Integrals on the parent element: Numerical integration

Since the shape functions are expressed in terms of the parent coordinate, it also is convenient to express the integrals in terms of the parent coordinate. This is easily obtained upon noting the differential is given in terms of the Jacobian by

$$\mathrm{d}x = \frac{\partial x}{\partial \xi}\, \mathrm{d}\xi = j_e(\xi)\, \mathrm{d}\xi \tag{3.46}$$

Thus, our integrals may be expressed as

$$\int_0^{h_e} f(x)\, \mathrm{d}x = \int_{-1}^{1} \hat{f}(\xi)\, j_e(\xi)\, \mathrm{d}\xi \tag{3.47}$$

where $\hat{f}(\xi) = f(x(\xi))$.

Often the integrals involve rational polynomial forms that are difficult to integrate in closed form. In this case it is expedient to use quadrature (numerical integration) to approximate the integrals. While quadrature may not produce exact results for the integrals, used correctly the integrals are sufficiently accurate for use in a finite element calculation. An efficient formula for polynomial forms is Gauss-Legendre quadrature that is defined on the interval $-1 \le \xi \le 1$ (this is the reason we chose this interval above!). The integral is replaced by the form

$$I = \int_{-1}^{1} f(\xi)\, \mathrm{d}\xi \approx \sum_{j=1}^{n} f(\xi_j)\, w_j \tag{3.48}$$

**Table 3.3** Gaussian Quadrature Points and Weights for $\int_{-1}^{1} f(\xi)\mathrm{d}\xi = \sum_{j=1}^{n} f(\xi_j)\, w_j$

| $n$ | $\xi_j$ | $w_j$ |
|---|---|---|
| 1 | 0 | 2.0 |
| 2 | $\pm 1/\sqrt{3}$ | 1.0 |
| 3 | $\pm\sqrt{0.6}$ | 5/9 |
| | 0.0 | 8/9 |
| 4 | $\dfrac{\pm\sqrt{3+\sqrt{4.8}}}{7}$ | $\dfrac{1}{2} - \dfrac{1}{3\sqrt{4.8}}$ |
| | $\dfrac{\pm\sqrt{3-\sqrt{4.8}}}{7}$ | $\dfrac{1}{2} + \dfrac{1}{3\sqrt{4.8}}$ |

With appropriate choice of the points $\xi_j$ and weights $w_j$ the formula integrates a polynomial in $\xi$ of degree $2n-1$ exactly. Table 3.3 gives the locations of the points and weights for the first four formulas. For $f(\xi)$ that are rational polynomials use of (3.48) results in an error for the integral; however, by using a sufficient number of points the error will be smaller than the error in the approximation of $u$ using polynomial shape functions.

**Example 3.3.** Stiffness for two-node element

The stiffness for the two-node element is computed from

$$\mathbf{K}^e = \int_{-1}^{1} \begin{Bmatrix} \dfrac{\partial N_1}{\partial x} \\ \dfrac{\partial N_2}{\partial x} \end{Bmatrix} E_e \begin{bmatrix} \dfrac{\partial N_1}{\partial x} & \dfrac{\partial N_2}{\partial x} \end{bmatrix} \frac{1}{2} h_e \, \mathrm{d}\xi = \frac{E_e}{h_e} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

which is identical to that obtained previously.

Numerical integration is performed using the one-point quadrature formula since the shape function derivatives are constant. The numerical integral is just

$$\int_{-1}^{1} \mathrm{d}\xi = w_1 = 2$$

which is obviously the correct answer. For this problem quadrature was not needed; however, for higher order elements quadrature greatly simplifies the calculations.

**Example 3.4.** Stiffness for three-node element

The stiffness for the three-node element involves the integral of

$$\mathbf{K}^e = \int_{-1}^{1} \begin{Bmatrix} \dfrac{\partial N_1}{\partial x} \\ \dfrac{\partial N_2}{\partial x} \\ \dfrac{\partial N_3}{\partial x} \end{Bmatrix} E_e \begin{bmatrix} \dfrac{\partial N_1}{\partial x} & \dfrac{\partial N_2}{\partial x} & \dfrac{\partial N_3}{\partial x} \end{bmatrix} j_e(\xi)\, \mathrm{d}\xi$$

where the shape functions are given by (3.43) and the Jacobian by (3.44). For the case where the coordinate for node 3 is centered between nodes 1 and 2 the derivatives are linear in $\xi$ and the Jacobian is constant. In this case the exact stiffness is recovered using a two-point quadrature formula. The reader can verify that the final result is

$$\mathbf{K}^e = \frac{E_e}{3h_e}\begin{bmatrix} 7 & 1 & -8 \\ 1 & 7 & -8 \\ -8 & -8 & 16 \end{bmatrix}$$

If node 3 is not centered the derivatives become rational functions and quadrature yields an approximate answer for the integrals. Numerical experiment shows that two points are still accurate enough to yield good solutions and for convergence to occur as more elements are added.

**Example 3.5.** Mass matrix for three-node element

The element mass matrix for the three-node element is computed from

$$\mathbf{M}^e = \int_{-1}^{1} \begin{Bmatrix} N_1 \\ N_2 \\ N_3 \end{Bmatrix} \rho_e \begin{bmatrix} N_1 & N_2 & N_3 \end{bmatrix} j_e(\xi)\, \mathrm{d}\xi$$

Since each shape function is a quadratic polynomial and the Jacobian can be a linear polynomial, the highest order polynomial is of order 5. Thus, a three-point quadrature is needed to evaluate the matrix exactly. In solution of transient problems the mass matrix must always be positive definite. Use of a lower order quadrature than three points will make the mass matrix indefinite—that is, possess a zero determinant—and should therefore never be used. For the case where node 3 is centered in the element and the density is constant the mass matrix is given by

$$\mathbf{M}^e = \frac{1}{30}\rho_e h_e \begin{bmatrix} 4 & -1 & 2 \\ -1 & 4 & 2 \\ 2 & 2 & 16 \end{bmatrix}$$

## 3.6 Hierarchical interpolation

The essence of the finite element method stated above is in approximating the unknown (displacement) in each element by an expansion given in (3.37). We have explicitly chosen to identify such variables with the values of the unknown function at element nodes, thus making

$$\tilde{u}_a = \hat{u}(x_a) \tag{3.49}$$

The shape functions so defined will be referred to as "standard" ones and are the basis of most finite element programs. If polynomial expansions are used and the element satisfies the completeness convergence condition (which specifies that rigid

body displacements cause no strain), it is clear that a constant value of $\tilde{u}_a$ specified at all nodes must result in a constant value of $\hat{u}$,

$$\hat{u} = \left( \sum_{a=1}^{n} N_a \right) u_0 = u_0 \tag{3.50}$$

when $\tilde{u}_a = u_0$. It follows that

$$\sum_{a=1}^{n} N_a = 1 \tag{3.51}$$

at all points of the domain. This important property is known as a *partition of unity* [12].

A drawback exists with "standard" functions; when element refinement occurs, totally new shape functions have to be generated and hence all calculations repeated. It is sometimes advantageous to avoid this difficulty by considering the expression (3.37) as a *series* in which the shape function $N_a$ does not depend on the number of nodes in the mesh $n$. This indeed is achieved with *hierarchic shape functions*.

The hierarchic concept is well illustrated by the one-dimensional (elastic bar) problem of Fig. 3.7. Here for simplicity the elastic property is taken as constant ($E$) and the body force $b$ is assumed to vary in such a manner as to produce the exact solution shown in the figure (with zero displacements at both ends).

Two meshes are shown and a linear interpolation between nodal points assumed. For both standard and hierarchic forms the coarse mesh gives

$$K_{11}^{c}\, \tilde{u}_1^{c} = f_1 \tag{3.52}$$

For a finer mesh two additional nodes are added and with the standard shape function the equations requiring solution are

$$\begin{bmatrix} K_{11}^{F} & K_{12}^{F} & 0 \\ K_{21}^{F} & K_{22}^{F} & K_{23}^{F} \\ 0 & K_{32}^{F} & K_{33}^{F} \end{bmatrix} \begin{Bmatrix} \tilde{u}_1 \\ \tilde{u}_2 \\ \tilde{u}_3 \end{Bmatrix} = \begin{Bmatrix} f_1 \\ f_2 \\ f_3 \end{Bmatrix} \tag{3.53}$$

In this form the zero matrices have been automatically inserted due to element interconnection which is here obvious, and we note that as no coefficients are the same, the new equations have to be resolved.

With the "hierarchic" form using the shape functions shown, a similar form of equation arises and an identical approximation is achieved (being simply given by a series of straight segments). The *final* solution is identical but the meaning of the parameters $\tilde{u}_a^{\star}$ is now different, as shown in Fig. 3.7.

Quite generally,

$$K_{11}^{F} = K_{11}^{c} \tag{3.54}$$

as an identical shape function is used for the first variable. Further, in this particular case the off-diagonal coefficients are zero and the final equations become, for the

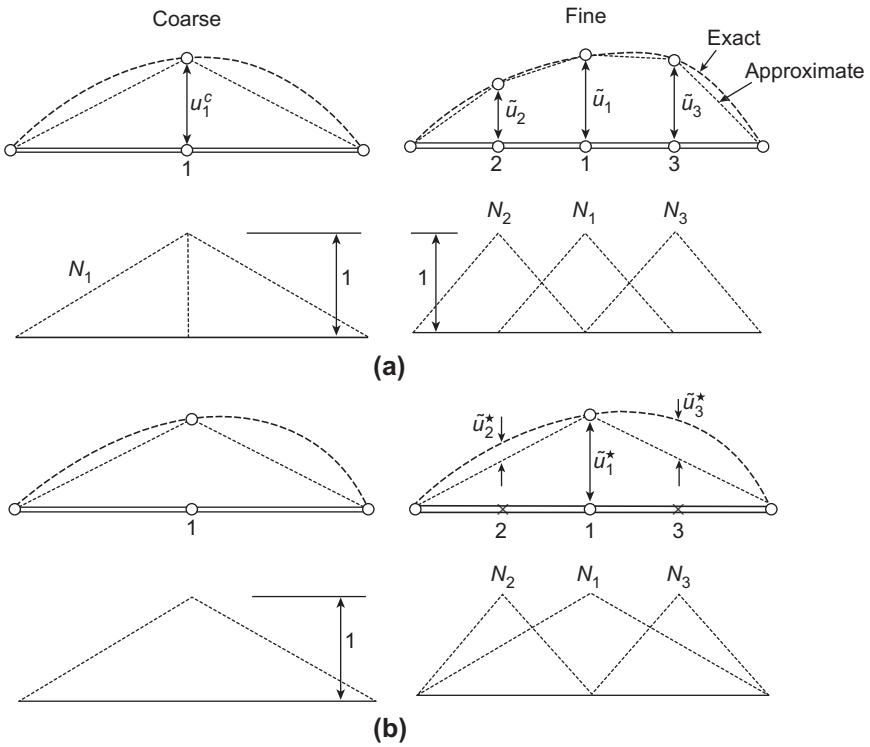



**FIGURE 3.7**
A one-dimensional problem of stretching of a uniform elastic bar by prescribed body forces.

fine mesh,

$$\begin{bmatrix} K_{11}^c & 0 & 0 \\ 0 & K_{22}^F & 0 \\ 0 & 0 & K_{33}^F \end{bmatrix} \begin{Bmatrix} \tilde{u}_1^\star \\ \tilde{u}_2^\star \\ \tilde{u}_3^\star \end{Bmatrix} = \begin{Bmatrix} f_1 \\ f_2 \\ f_3 \end{Bmatrix} \tag{3.55}$$

The "diagonality" feature is only true in the one-dimensional problem, but in higher dimensions it will be found that the matrices obtained using hierarchic shape functions are more nearly diagonal and hence usually imply better conditioning than those with standard shape functions. Improved conditioning is advantageous when iterative methods are used to solve the assembled stiffness equations.

Although the variables are now not subject to the obvious interpretation (as local displacement values), they can be easily transformed to those if desired. Though it is not usual to use hierarchic forms in linearly interpolated elements, their derivation in polynomial form is simple and very advantageous.

The reader should note that with hierarchic forms it is convenient to consider the finer mesh as still using the same, coarse, elements but now adding additional refining functions.

Hierarchic forms provide a link with other approximate (orthogonal) series solutions. Many problems solved in classical literature by trigonometric Fourier series expansion are indeed particular examples of this approach.

**Example 3.6.** Stiffness for three-node hierarchical element

For a three-node element with hierarchical shape functions

$$N_1(\xi) = l_1^1(\xi) = \frac{1}{2}(1-\xi)$$
$$N_2(\xi) = l_2^1(\xi) = \frac{1}{2}(1+\xi)$$
$$N_3(\xi) = l_3^2(\xi) = 1-\xi^2$$

the stiffness matrix for an element with node 3 placed at the center of the element is given by

$$\mathbf{K}^e = \frac{E_e}{3h_e}\begin{bmatrix} 3 & -3 & 0 \\ -3 & 3 & 0 \\ 0 & 0 & 16 \end{bmatrix}$$

**Example 3.7.** Mass for three-node hierarchical element

Using the shape functions in the previous example the mass matrix for the three-node element with the mid-side node centered is given by

$$\mathbf{M}^e = \frac{1}{30}\rho_e h_e \begin{bmatrix} 10 & 5 & 10 \\ 5 & 10 & 10 \\ 10 & 10 & 16 \end{bmatrix}$$

Thus, while the hierarchical formulation helps in defining the stiffness matrix it does not have the same advantage in describing the mass behavior.

## 3.7 Axisymmetric one-dimensional problem

The axisymmetric elasticity problem described in Chapter 2 also can be reduced to a one-dimensional problem in terms of the radial direction. Here we shall assume an infinitely long cylinder in which the displacement field is given by

$$u(r,\, z) = u(r) \quad \text{and} \quad v(r,\, z) = 0 \tag{3.56}$$

The nonzero strains for this case may be determined from (2.10) as

$$\boldsymbol{\varepsilon} = \begin{Bmatrix} \varepsilon_r \\ \varepsilon_\theta \end{Bmatrix} = \begin{Bmatrix} \dfrac{\partial u}{\partial r} \\ \dfrac{u}{r} \end{Bmatrix} \tag{3.57}$$

The equilibrium equation given in (2.23) simplifies to the single form

$$\frac{\partial \sigma_r}{\partial r} + \frac{\sigma_r - \sigma_\theta}{r} + b_r = \rho \frac{\partial^2 u}{\partial t^2} \tag{3.58}$$

For an isotropic material the stress-strain relations, now including temperature effects, are given by

$$\begin{Bmatrix} \sigma_r \\ \sigma_\theta \end{Bmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} (1-\nu) & \nu \\ \nu & (1-\nu) \end{bmatrix} \begin{Bmatrix} \varepsilon_r - \alpha\, \Delta T \\ \varepsilon_\theta - \alpha\, \Delta T \end{Bmatrix} \tag{3.59a}$$

and

$$\sigma_z = \nu\,(\sigma_r + \sigma_\theta) - E\,\alpha\,\Delta T \tag{3.59b}$$

where $\alpha\,\Delta T$ is the thermal strain effect.

### 3.7.1 Weak form for axisymmetric problem

The construction of the weak form for the axisymmetric problem proceeds in an identical manner as described in Section 3.1. We always use Eq. (3.1) to do the construction. The first step in the construction gives

$$g(w,\, u,\, \sigma_r,\, \sigma_\theta) = w(r) \left( \rho \frac{\partial^2 u}{\partial t^2} - b_r - \frac{\partial \sigma_r}{\partial r} - \frac{\sigma_r - \sigma_\theta}{r} \right) = 0$$

Next we integrate over the *volume* of the domain. In the case of axisymmetry, assuming a unit thickness in the $z$ direction, the differential of the one-dimensional domain volume is given by $2\pi r\, \mathrm{d}r$. Thus, for the case where $a \le r \le b$ the weak form becomes

$$\begin{aligned} \mathrm{G}(w,\, u,\, \sigma_r,\, \sigma_\theta) = {} & 2\,\pi \int_a^b w(r) \left( \rho \frac{\partial^2 u}{\partial t^2} - b_r \right) r\, \mathrm{d}r \\ & - 2\,\pi \int_a^b w(r) \left( \frac{\partial \sigma_r}{\partial r} + \frac{\sigma_r - \sigma_\theta}{r} \right) r\, \mathrm{d}r = 0 \end{aligned}$$

The only spatial term to integrate by parts is

$$\begin{aligned} -\int_a^b w(r) \frac{\partial \sigma_r}{\partial r}\, r\, \mathrm{d}r &= \int_a^b \frac{\partial}{\partial r} \left[ w(r)\, r \right] \sigma_r\, \mathrm{d}r - w(r)\, r\, \sigma_r\, n_r \Big|_a^b \\ &= \int_a^b \left( \frac{\partial w}{\partial r}\, r + w \right) \sigma_r\, \mathrm{d}r - w(r)\, r\, t_r \Big|_a^b \end{aligned}$$

where $t_r = \sigma_r\, n_r$ is the radial traction to the boundary. Inserting the above into the weak form yields

$$\begin{aligned} \mathrm{G}(w,\, u,\, \sigma_r,\, \sigma_\theta) = {} & 2\,\pi \int_a^b w(r) \left( \rho \frac{\partial^2 u}{\partial t^2} - b_r \right) r\, \mathrm{d}r \\ & + 2\,\pi \int_a^b \left( \frac{\partial w}{\partial r}\, \sigma_r + \frac{w}{r}\, \sigma_\theta \right) r\, \mathrm{d}r - 2\,\pi w(r)\, r\, t_r \Big|_a^b = 0 \end{aligned} \tag{3.60}$$

We note that the factor $2\pi$ appears in all terms and can be omitted if desired. The boundary term may be split into the traction and displacement parts as

$$w(r)\, r\, t_r|_a^b = w\, r\, t_r|_{\Gamma_u} + w\, r\, \bar{t}_r\big|_{\Gamma_t}$$

and again for simplicity we will let $w = 0$ and $u = \bar{u}$ on $\Gamma_u$ to write the final weak form as

$$\begin{aligned} \mathrm{G}(w,\, u,\, \sigma_r,\, \sigma_\theta) = 2\pi \int_a^b w(r) \left(\rho \frac{\partial^2 u}{\partial t^2} - b_r\right) r\, \mathrm{d}r \\ + 2\pi \int_a^b \left(\frac{\partial w}{\partial r}\sigma_r + \frac{w}{r}\sigma_\theta\right) r\, \mathrm{d}r - 2\pi w(r)\, r\, \bar{t}_r\big|_{\Gamma_t} = 0 \end{aligned} \tag{3.61}$$

### 3.7.2 A variational notation

By now the reader should have observed that the multiplier of each stress has an identical form as that for the associated strain. Indeed, if we replace the arbitrary function $w$ by a *variational* term $\delta u$ (read as "variation of $u$"), the association is complete. Readers familiar with "virtual work" concepts will observe that the variational notation for displacements is identical to "virtual displacement" and we will use the terms interchangeably in this text. Inserting this variational form the axisymmetric variational strains (or virtual strains) are given by

$$\begin{aligned} \frac{\partial w}{\partial r} &\to \frac{\partial \delta u}{\partial r} = \delta\varepsilon_r \\ \frac{w}{r} &\to \frac{\delta u}{r} = \delta\varepsilon_\theta \end{aligned}$$

and we can write (3.61) as

$$\begin{aligned} \mathrm{G}(\delta u, u, \sigma_r, \sigma_\theta) = 2\pi \int_a^b \delta u(r) \left(\rho \frac{\partial^2 u}{\partial t^2} - b_r\right) r\, \mathrm{d}r \\ + 2\pi \int_a^b \left(\delta\varepsilon_r\, \sigma_r + \delta\varepsilon_\theta\, \sigma_\theta\right) r\, \mathrm{d}r - 2\pi\delta u(r)\, r\, \bar{t}_r\big|_{\Gamma_t} = 0 \end{aligned} \tag{3.62}$$

This form also suggests that there is no need to look for two functions $\phi_n$ and $\psi_m$ to do the approximation—the same spatial functions may be used for both. Indeed, in the preceding examples we have already used the same functions with success. This property is not true for all classes of problems. For example, if we consider the ordinary differential equation

$$a \frac{\mathrm{d}^2 u}{\mathrm{d}x^2} + b \frac{\mathrm{d}u}{\mathrm{d}x} + cu = f$$

we will find that for large values of $b$ use of the same functions for $u$ and $\delta u$ leads to poor performance. The reader is referred to the companion text on fluids for more information on this class of problems [13].

### 3.7.3 Irreducible form for axisymmetric problem

If we substitute the constitutive equation (3.59a) into the weak form (3.62) we obtain

$$\begin{aligned} \mathrm{G}(\delta u, u) = {} & 2\pi \int_a^b \delta u(r) \left( \rho \frac{\partial^2 u}{\partial t^2} - b_r \right) r \, \mathrm{d}r \\ & + 2\pi \int_a^b \begin{bmatrix} \delta\varepsilon_r & \delta\varepsilon_\theta \end{bmatrix} \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \begin{Bmatrix} \varepsilon_r - \alpha\Delta T \\ \varepsilon_\theta - \alpha\Delta T \end{Bmatrix} r \, \mathrm{d}r \\ & - 2\pi \, \delta u(r) \, r \, \bar{t}_r \big|_{\Gamma_t} = 0 \end{aligned} \tag{3.63}$$

where all strains are expressed in terms of the displacement $u$ or $\delta u$ and the $D_{ij}$ are defined by (3.59a). This is obviously an irreducible or displacement form for the axisymmetric problem.

### 3.7.4 Finite element solution

Again we divide the region of a cylinder defined by $a \le r \le b$ into a set of $M$ elements defined by nodes $r_i$ and express the integrals as

$$2\pi \int_a^b (\cdot) \, r \, \mathrm{d}r = 2\pi \sum_{i=1}^{M} \int_{r_i}^{r_{i+1}} (\cdot) \, r \, \mathrm{d}r = 2\pi \sum_{i=1}^{M} \int_{-1}^{1} (\cdot) \, r j_e \, \mathrm{d}\xi \tag{3.64a}$$

The strain terms in the weak form (3.63) may be expressed in a matrix form as

$$\boldsymbol{\varepsilon} = \begin{Bmatrix} \varepsilon_r \\ \varepsilon_\theta \end{Bmatrix} = \begin{Bmatrix} \dfrac{\partial u}{\partial r} \\ \dfrac{u}{r} \end{Bmatrix} = \begin{bmatrix} \dfrac{\partial N_1}{\partial r} & \dfrac{\partial N_2}{\partial r} \\ \dfrac{N_1}{r} & \dfrac{N_2}{r} \end{bmatrix} \begin{Bmatrix} \tilde{u}_i \\ \tilde{u}_{i+1} \end{Bmatrix} = \begin{bmatrix} \mathbf{B}_1 & \mathbf{B}_2 \end{bmatrix} \tilde{\mathbf{u}}_e \tag{3.64b}$$

where

$$\mathbf{B}_i = \begin{Bmatrix} \dfrac{\partial N_i}{\partial r} \\ \dfrac{N_i}{r} \end{Bmatrix}, \quad i = 1, 2 \tag{3.64c}$$

The moduli and shape functions are grouped as

$$\begin{aligned} \tilde{\mathbf{u}}_e &= \begin{Bmatrix} \tilde{u}_i \\ \tilde{u}_{i+1} \end{Bmatrix} \\ \mathbf{D} &= \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \\ \mathbf{N} &= \begin{bmatrix} N_1 & N_2 \end{bmatrix} \quad \text{and} \\ \mathbf{B} &= \begin{bmatrix} \mathbf{B}_1 & \mathbf{B}_2 \end{bmatrix} \end{aligned} \tag{3.64d}$$

This permits the weak form to be expressed by

$$\begin{aligned} \mathrm{G}(\delta\mathbf{u},\,\mathbf{u}) = 2\,\pi \sum_{i=1}^{M} \delta\tilde{\mathbf{u}}_e^{\mathrm{T}} \Bigg( & \int_{r_i}^{r_{i+1}} \mathbf{N}^{\mathrm{T}}\,\rho\mathbf{N}\,r\,\mathrm{d}r\,\ddot{\tilde{\mathbf{u}}}_e \\ & + \int_{r_i}^{r_{i+1}} \mathbf{B}^{\mathrm{T}}\,\mathbf{D}\mathbf{B}\,r\,\mathrm{d}r\,\tilde{\mathbf{u}}_e - \int_{r_i}^{r_{i+1}} \mathbf{B}^{\mathrm{T}}\,\mathbf{D}\,\boldsymbol{\alpha}\Delta T\,r\,\mathrm{d}r \\ & - \int_{r_i}^{r_{i+1}} \mathbf{N}^{\mathrm{T}}\,b_r\,r\,\mathrm{d}r \Bigg) - 2\pi\delta u(r)\,\bar{t}_r\,r\big|_{\Gamma_t} = 0 \end{aligned} \tag{3.65}$$

where $\boldsymbol{\alpha}^{\mathrm{T}} = [\alpha\;\alpha]$. In matrix notation the discretized weak form may be written as

$$\mathrm{G}(\delta\mathbf{u},\,\mathbf{u}) = \sum_{e} \delta\tilde{\mathbf{u}}_e^{\mathrm{T}} \left[ \mathbf{M}^e\ddot{\tilde{\mathbf{u}}}_e + \mathbf{K}^e\tilde{\mathbf{u}} - \mathbf{f}^e \right] = 0 \tag{3.66}$$

where

$$\begin{aligned} \mathbf{M}^e &= 2\pi \int_{r_i}^{r_{i+1}} \mathbf{N}^{\mathrm{T}}\rho\mathbf{N}\,r\,\mathrm{d}r \\ \mathbf{K}^e &= 2\pi \int_{r_i}^{r_{i+1}} \mathbf{B}^{\mathrm{T}}\mathbf{D}\,\mathbf{B}\,r\,\mathrm{d}r \\ \mathbf{f}^e &= 2\pi \int_{r_i}^{r_{i+1}} \mathbf{N}^{\mathrm{T}} b_r\,r\,\mathrm{d}r + 2\pi \int_{r_i}^{r_{i+1}} \mathbf{B}^{\mathrm{T}}\,\mathbf{D}\,\boldsymbol{\alpha}\Delta T\,r\,\mathrm{d}r \\ &\quad + 2\pi\delta u(r)\,\bar{t}_r\,r\big|_{\Gamma_t} \end{aligned} \tag{3.67}$$

The contribution of the last term of $\mathbf{f}^e$ applies only to those elements at the boundary of the solution domain. After summing over all elements we obtain the assembled form as

$$\mathrm{G}(\delta\mathbf{u},\mathbf{u}) = \delta\tilde{\mathbf{u}}^{\mathrm{T}} \left( \mathbf{M}\,\ddot{\tilde{\mathbf{u}}} + \mathbf{K}\,\tilde{\mathbf{u}} - \mathbf{f} \right) = 0$$

Since the $\delta\tilde{\mathbf{u}}$ is arbitrary, the final set of equations is again in the form of the standard linear problem given in (3.30).

**Example 3.8.** Thick-walled cylinder subjected to internal pressure

As an example of *static* axisymmetric problem we consider a thick-walled cylinder subjected to an internal pressure and a traction-free external pressure. For the axisymmetric one-dimensional problem there are no rigid body modes. Hence, no essential displacement boundary conditions are needed. We ignore temperature and intertia effects. The properties of the cylinder are given as

$$a = 5,\;\; b = 10,\;\; b_r = 0,\;\; \bar{t}_r(a) = -2,\;\; E = 10{,}000 \text{ and } \nu = 0.3$$

The weak form for the above static finite element problem is given by

$$\mathrm{G}(\delta\mathbf{u},\,\mathbf{u}) = \sum_{i=1}^{M} \delta\tilde{\mathbf{u}}_e^{\mathrm{T}} \int_{r_i}^{r_{i+1}} \mathbf{B}^{\mathrm{T}}\,\mathbf{D}\mathbf{B}\,r\,\mathrm{d}r\,\tilde{\mathbf{u}}_e - \delta u(r)\,\bar{t}_r\,r\big|_{\Gamma_t} = 0 \tag{3.68}$$

where we also drop the $2\pi$ factor. The stiffness matrix for each element may be computed in the parent domain as

$$\mathbf{K}^e = \int_{-1}^{1} \mathbf{B}^{\mathrm{T}}\mathbf{D}\mathbf{B}\, r\; j_e(r)\, \mathrm{d}\xi$$

and is approximated by two-point Gauss-Legendre quadrature. An eight-element problem of linear, two-node elements is considered with element coordinates and displacements given by

$$r^e = N_1(\xi)\, r_1 + N_2(\xi)\, r_2 \quad \text{and} \quad \hat{u}^e = N_1^e(\xi)\, \tilde{u}_1^e + N_2(\xi)\, \tilde{u}_2^e$$

The nodes for the problem are placed at $r_1 = 5$ and $r_9 = 10$ with the other nodes equally spaced in between. The nodal force from the internal boundary traction is given by

$$f_1 = a\, \bar{t}_r(a) = -2 \cdot 5 = -10$$

The results for the displacement and stresses are shown in Fig 3.8. The reader should note the unusual behavior for $\sigma_r$ but again also note that super-convergent points exist for all stress components.

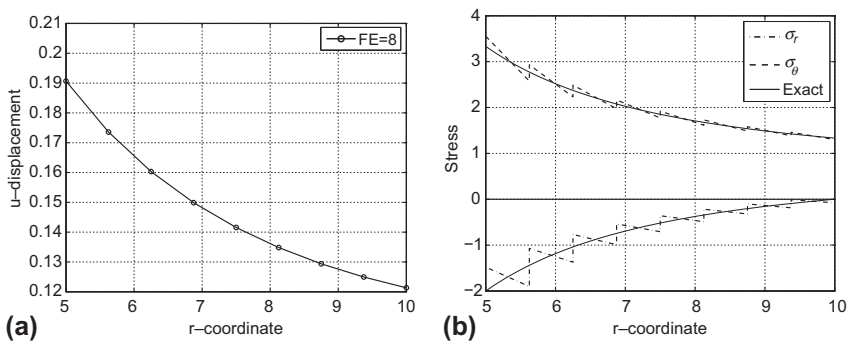



**FIGURE 3.8**

One-dimensional finite element solution to axisymmetric problem: (a) displacements and (b) stresses.

## 3.8 Transient problems

When loading is applied with time variation such that the inertial term is significant the full weak form given in (3.30) must be considered. Two methods of solution to the problem may be considered: discrete approximation in time and semi-discretization in which an explicit form of the time behavior is assumed. Both methods reduce the problem to one of algebraic form. Here we summarize the two approaches when applied to the finite element equations.

### 3.8.1 Discrete time methods

In discrete time methods the transient solution for the time duration of interest is described at a set of discrete time points $t_n$ and approximations to the solution are determined at these points. Eq. (3.30) contains time derivatives up to second degree; thus, we need three different discrete time values which we denote as

$$\left(\tilde{\mathbf{u}}(t_n),\ \dot{\tilde{\mathbf{u}}}(t_n),\ \ddot{\tilde{\mathbf{u}}}(t_n)\right) \approx \left(\tilde{\mathbf{u}}_n,\ \tilde{\mathbf{v}}_n,\ \tilde{\mathbf{a}}_n\right) \tag{3.69}$$

There are many different discrete forms; one which is frequently used is the *Newmark method* [14]. The Newmark method is a *one step* procedure that describes how a solution known at time $t_n$ is determined at time $t_{n+1}$. Since there are three discrete quantities given in (3.30) we need three independent time equations to obtain a solution at each discrete time. Two of the equations are the Newmark formulas and the third is the discrete equilibrium equation. These are written as

$$\begin{aligned}
\tilde{\mathbf{u}}_{n+1} &= \tilde{\mathbf{u}}_n + \Delta t\, \tilde{\mathbf{v}}_n + \left(\frac{1}{2} - \beta\right) \Delta t^2 \tilde{\mathbf{a}}_n + \beta\, \Delta t^2 \tilde{\mathbf{a}}_{n+1} \\
\tilde{\mathbf{v}}_{n+1} &= \tilde{\mathbf{v}}_n + (1-\gamma)\, \Delta t\, \tilde{\mathbf{a}}_n + \gamma\, \Delta t\, \tilde{\mathbf{a}}_{n+1} \\
\mathbf{R}_{n+1} &= \mathbf{f}_{n+1} - \mathbf{M}\, \tilde{\mathbf{a}}_{n+1} - \mathbf{K}\, \tilde{\mathbf{u}}_{n+1} = \mathbf{0}
\end{aligned} \tag{3.70}$$

where $\Delta t = t_{n+1} - t_n$. The quantity $\mathbf{R}_{n+1}$ is called the *residual* of the equilibrium equation. The result $\mathbf{R}_{n+1} = \mathbf{0}$ describes the requirement that the discrete solution must be satisfied at each time $t_{n+1}$. In these equations the two parameters $\beta$ and $\gamma$ control the overall numerical response for *stability* and *numerical dissipation*. A stable solution is one in which the response after a load is removed does not become large at later discrete times. Numerical dissipation in the above equations implies the solution approaches zero at later discrete times after the loading is removed.

To solve (3.70) we may select any one of the discrete time quantities as the primary dependent time variable and solve for the other two from the Newmark formulas. The simplest form is to select $\tilde{\mathbf{a}}_{n+1}$ as the primary variable and directly substitute the two Newmark formulas into the residual equation, $\mathbf{R}_{n+1}$. The result gives

$$\left[\mathbf{M} + \beta\, \Delta t^2 \mathbf{K}\right] \tilde{\mathbf{a}}_{n+1} = \mathbf{f}_{n+1} - \mathbf{K}\, \widehat{\mathbf{u}}_{n+1} \tag{3.71}$$

where

$$\widehat{\mathbf{u}}_{n+1} = \tilde{\mathbf{u}}_n + \Delta t\, \tilde{\mathbf{v}}_n + \left(\frac{1}{2} - \beta\right) \Delta t^2 \tilde{\mathbf{a}}_n$$

collects all the values at $t_n$ together. The acceleration parameters are given formally as

$$\tilde{\mathbf{a}}_{n+1} = \left[\mathbf{M} + \beta\, \Delta t^2 \mathbf{K}\right]^{-1} \left(\mathbf{f}_{n+1} - \mathbf{K}\, \widehat{\mathbf{u}}_{n+1}\right) \tag{3.72}$$

and we note solving the transient problem involves a repeated solution of the type indicated in (3.21) but with a modified stiffness matrix. Two types of methods exist: (a) $\beta > 0$ which is called an *implicit* method and (b) $\beta = 0$ which is called an *explicit* method. Explicit methods are always conditionally stable, meaning a limit on the size of $\Delta t$ exists beyond which the solution grows in an uncontrolled manner. If the mass matrix $\mathbf{M}$ is diagonal (see Appendix H), explicit methods are very cheap for each time step and are used extensibly in crash analysis of vehicles. Their main drawback is the small time step which must be used to remain stable. Implicit methods for linear problems can be unconditionally stable. Using $\beta = 1/4$ and $\gamma = 1/2$ results in an unconditionally stable solution without any numerical dissipation for our linear problem [15,16]. The Newmark procedure with these parameters is often called a *trapezoidal rule* or a *constant acceleration* method.

After the acceleration is determined from (3.72) the displacement and velocity at $t_{n+1}$ may be determined from the two Newmark formulas and, thus, all the values at $t_{n+1}$ are known and may be used to advance the solution to the next time. Repeating the process provides a discrete solution to the problem for any specified loading history, $\mathbf{f}_{n+1}$. We note that the solution for the time response is obtained from the solution of a set of *algebraic equations* in the same manner as for static problems.

#### 3.8.1.1 Stability and dissipation

It was noted that the parameters $\beta$ and $\gamma$ control the stability and numerical dissipation in the algorithm. To observe how this may be assessed we consider the case where $\beta = 1/4$ and $\gamma = 1/2$. For this case we can write the two Newmark formulas as

$$\tilde{\mathbf{u}}_{n+1} - \tilde{\mathbf{u}}_n = \frac{1}{2}\, \Delta t\, \left(\tilde{\mathbf{v}}_n + \tilde{\mathbf{v}}_{n+1}\right)$$
$$\tilde{\mathbf{v}}_{n+1} - \tilde{\mathbf{v}}_n = \frac{1}{2} \Delta t\, \left(\tilde{\mathbf{a}}_n + \tilde{\mathbf{a}}_{n+1}\right)$$

Adding together the equilibrium equations evaluated at $t_n$ and $t_{n+1}$ and multiplying by the transpose of $(\tilde{\mathbf{u}}_{n+1} - \tilde{\mathbf{u}}_n)/2$ gives

$$\frac{1}{2} \left(\tilde{\mathbf{u}}_{n+1} - \tilde{\mathbf{u}}_n\right)^{\mathrm{T}} \mathbf{M} \left(\tilde{\mathbf{a}}_{n+1} + \tilde{\mathbf{a}}_n\right) + \frac{1}{2} \left(\tilde{\mathbf{u}}_{n+1} - \tilde{\mathbf{u}}_n\right)^{\mathrm{T}} \mathbf{K} \left(\tilde{\mathbf{u}}_{n+1} + \tilde{\mathbf{u}}_n\right)$$
$$= \frac{1}{2} \left(\tilde{\mathbf{u}}_{n+1} - \tilde{\mathbf{u}}_n\right)^{\mathrm{T}} \left(\mathbf{f}_{n+1} + \mathbf{f}_n\right)$$

Using the Newmark formulas on the mass term we obtain (after dropping the common factor $\Delta t$)

$$\begin{aligned}\frac{1}{2}\left(\tilde{\mathbf{u}}_{n+1}-\tilde{\mathbf{u}}_n\right)^{\mathrm{T}}\mathbf{M}\left(\tilde{\mathbf{a}}_{n+1}+\tilde{\mathbf{a}}_n\right) &= \frac{1}{2}\left(\tilde{\mathbf{v}}_{n+1}+\tilde{\mathbf{v}}_n\right)^{\mathrm{T}}\mathbf{M}\left(\tilde{\mathbf{v}}_{n+1}-\tilde{\mathbf{v}}_n\right)\\ &= \frac{1}{2}\tilde{\mathbf{v}}_{n+1}^{\mathrm{T}}\mathbf{M}\,\tilde{\mathbf{v}}_{n+1}-\frac{1}{2}\tilde{\mathbf{v}}_n^{\mathrm{T}}\mathbf{M}\,\tilde{\mathbf{v}}_n\\ &= \mathcal{K}_{n+1}-\mathcal{K}_n\end{aligned}$$

where $\mathcal{K}_n$ is the kinetic energy at time $t_n$. Similarly, for the stiffness term we recover

$$\begin{aligned}\frac{1}{2}\left(\tilde{\mathbf{u}}_{n+1}-\tilde{\mathbf{u}}_n\right)^{\mathrm{T}}\mathbf{K}\left(\tilde{\mathbf{u}}_{n+1}+\tilde{\mathbf{u}}_n\right) &= \frac{1}{2}\tilde{\mathbf{u}}_{n+1}^{\mathrm{T}}\mathbf{K}\,\tilde{\mathbf{u}}_{n+1}-\frac{1}{2}\tilde{\mathbf{u}}_n^{\mathrm{T}}\mathbf{K}\,\tilde{\mathbf{u}}_n\\ &= \mathcal{U}_{n+1}-\mathcal{U}_n\end{aligned}$$

where $\mathcal{U}_n$ is the elastic potential energy at time $t_n$. In the above we assume $\mathbf{K}$ and $\mathbf{M}$ are symmetric. Denoting the incremental work of the applied loads as

$$\frac{1}{2}\left(\tilde{\mathbf{u}}_{n+1}-\tilde{\mathbf{u}}_n\right)^{\mathrm{T}}\left(\mathbf{f}_{n+1}+\mathbf{f}_n\right)=\Delta\mathcal{W}$$

and collecting all the terms together gives the energy equation

$$\begin{aligned}\mathcal{K}_{n+1}+\mathcal{U}_{n+1} &= \mathcal{K}_n+\mathcal{U}_n+\Delta\mathcal{W} \quad \text{or}\\ \mathcal{E}_{n+1} &= \mathcal{E}_n+\Delta\mathcal{W}\end{aligned}$$

where $\mathcal{E}_n$ is the total energy at time $t_n$. For times when no incremental work is added or removed we observe that the *energy is conserved*. Indeed, the energy only changes if forces are applied to the system and in these cases the amount of energy change is a finite quantity. Solution methods with these properties are termed *unconditionally stable*. Of course the solution may not be accurate if the time increments are too large; thus, additional assessments are needed.

### 3.8.2 Semi-discretization of the problem

An alternative approach to discrete time stepping uses a specified form for the time behavior. For linear ordinary differential equations with constant coefficients the exact time behavior is given in terms of exponentials in time. For the second-order problem we can substitute the form

$$\tilde{\mathbf{u}}=\bar{\mathbf{u}}\exp(i\omega t) \quad \text{and} \quad \mathbf{f}=\bar{\mathbf{f}}\exp(i\omega t) \quad \text{where } i=\sqrt{-1} \tag{3.73}$$

into (3.30) and divide by the exponential term. The governing equation becomes

$$\left(\mathbf{K}-\omega^2\mathbf{M}\right)\bar{\mathbf{u}}=\bar{\mathbf{f}} \tag{3.74}$$

In this form the amplitude of the solution $\bar{\mathbf{u}}$ depends upon the frequency of the excitation $\omega$; however, the solution procedure is again one with an algebraic equation that is identical in form to that for a static problem. Considering all the frequencies we will find that near certain frequencies the amplitude becomes very large. To understand

this phenomenon we consider a simple one degree of freedom problem given by

$$\left(k - \omega^2 m\right) \bar{u} = \bar{f}$$

The solution to the problem is given by

$$\bar{u} = \frac{\bar{f}}{\left(k - \omega^2 m\right)}$$

We can immediately observe that the frequency $\omega = \sqrt{k/m}$ will result in an infinite value of $\bar{u}$. Such frequencies are called the *natural frequencies* of the system. In real physical systems an infinite solution does not occur due to either material dissipation or large displacement effects. However, the determination of the natural frequencies of an elastic body can provide useful information about the properties of linear systems. It is also used as a solution procedure for linear problems. The natural frequencies are independent of the loading and may be computed from the homogeneous equation

$$\left(\mathbf{K} - \omega_i^2 \mathbf{M}\right) \bar{\mathbf{u}}_i = \mathbf{0}$$

The subscript is added to emphasize that a solution to the homogeneous equations is possible only at a finite number of natural frequencies, $\omega_i$. Thus, in general we will need to solve the problem

$$\mathbf{K}\,\bar{\mathbf{U}} = \mathbf{M}\,\bar{\mathbf{U}}\,\boldsymbol{\Lambda} \tag{3.75}$$

where

$$\bar{\mathbf{U}} = \left[\begin{array}{cccc} \bar{\mathbf{u}}_1 & \bar{\mathbf{u}}_2 & \cdots & \bar{\mathbf{u}}_n \end{array}\right]^{\mathrm{T}} \quad \text{and} \quad \boldsymbol{\Lambda} = \begin{bmatrix} \omega_1^2 & 0 & \cdots & 0 \\ 0 & \omega_2^2 & 0 & \cdots \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & \omega_n^2 \end{bmatrix}$$

The above problem is an *eigenvalue problem* (see Appendix A) that has the solution pairs $\bar{\mathbf{u}}_i$, $\omega_i^2$ for $i = 1, 2, \ldots, n$. The $\bar{\mathbf{u}}_i$ is the *eigenvector* and $\omega_i$ is the associated eigenvalue.

For each mode $i$ of the problem we can compute a scalar mass and stiffness from

$$m_i = \bar{\mathbf{u}}_i^{\mathrm{T}} \mathbf{M} \bar{\mathbf{u}}_i = 1 \quad \text{and} \quad k_i = \bar{\mathbf{u}}_i^{\mathrm{T}} \mathbf{K} \bar{\mathbf{u}}_i = \omega_i^2$$

where the unity condition on $m_i$ is enforced to scale the eigenvector. Next, considering two separate pairs we can write

$$\mathbf{K}\,\bar{\mathbf{u}}_i = \mathbf{M}\,\bar{\mathbf{u}}_i\,\omega_i^2 \quad \text{and} \quad \mathbf{K}\,\bar{\mathbf{u}}_j = \mathbf{M}\,\bar{\mathbf{u}}_j\,\omega_j^2$$

Premultiplying the first by $\bar{\mathbf{u}}_j$, the second by $\bar{\mathbf{u}}_i$, noting symmetry for $\mathbf{K}$ and $\mathbf{M}$, and taking the difference gives

$$\bar{\mathbf{u}}_j^{\mathrm{T}} \mathbf{M} \bar{\mathbf{u}}_i \left(\omega_i^2 - \omega_j^2\right) = 0$$

For situations where $\omega_i \neq \omega_j$ we then obtain the result

$$\bar{\mathbf{u}}_j^{\mathrm{T}} \mathbf{M} \bar{\mathbf{u}}_i = 0 \tag{3.76}$$

For cases in which $\omega_i = \omega_j$ we will enforce (3.76) and, thus, the eigenvectors are constructed to be *mass orthonormal*.

A solution to (3.30) may now be constructed by assuming

$$\mathbf{u}(t) = \sum_{i=1}^{m} \bar{\mathbf{u}}_i \, y_i(t) \tag{3.77}$$

and obtaining

$$\sum_{i=1}^{m} \left( \mathbf{M} \bar{\mathbf{u}}_i \, \ddot{y}_i + \mathbf{K} \, \bar{\mathbf{u}}_i \, y_i \right) = \mathbf{f}$$

Premultiplying by the transpose of eigenvector $\bar{\mathbf{u}}_j$ gives $m$ scalar equations

$$\ddot{y}_j + \omega_j^2 \, y_j = f_j, \quad j = 1, 2, \ldots, m \tag{3.78}$$

where

$$f_j = \bar{\mathbf{u}}_j^{\mathrm{T}} \mathbf{f}$$

The reader can easily solve this equation for simple forms of loading. However, for general time-varying loading a numerical approach is needed. We can, of course, apply the Newmark method to numerically solve each equation and simply add the solution together as indicated in (3.77).

### 3.8.2.1 Stability of modes

The stability of a discrete time method may also be established from the eigenmodes of the semi-discrete problem. For example letting $y_j = u$ in (3.78) and substituting into a scalar form of the Newmark formulas (3.70) we obtain

$$\begin{aligned} u_{n+1} &= u_n + \Delta t v_n + \left( \frac{1}{2} - \beta \right) \Delta t^2 [f_n - \omega^2 u_n] + \beta \Delta t^2 [f_{n+1} - \omega^2 u_{n+1}] \\ v_{n+1} &= v_n + (1 - \gamma) \Delta t [f_n - \omega^2 u_n] + \gamma \Delta t [f_{n+1} - \omega^2 u_{n+1}] \end{aligned} \tag{3.79}$$

Multiplying the second equation by $\Delta t$ and collecting terms the above may be written in matrix form as

$$\begin{aligned} \begin{bmatrix} (1 + \beta \Omega^2) & 0 \\ \gamma \Omega^2 & 1 \end{bmatrix} \begin{Bmatrix} u_{n+1} \\ \Delta t v_{n+1} \end{Bmatrix} &= \begin{bmatrix} \left( 1 - \left( \frac{1}{2} - \beta \right) \Omega^2 \right) & 1 \\ -(1 - \gamma) \Omega^2 & 1 \end{bmatrix} \begin{Bmatrix} u_n \\ \Delta t v_n \end{Bmatrix} \\ &+ \begin{Bmatrix} \left( \frac{1}{2} - \beta \right) g_n + \beta g_{n+1} \\ (1 - \gamma) g_n + \gamma g_{n+1} \end{Bmatrix} \end{aligned} \tag{3.80}$$

where $\Omega = \omega \Delta t$ and $g_n = \Delta t^2 f_n$. Solving for the solution at $t_{n+1}$ gives

$$\begin{Bmatrix} u_{n+1} \\ \Delta t v_{n+1} \end{Bmatrix} = \frac{1}{1+\beta\Omega^2} \begin{bmatrix} 1 & 0 \\ -\gamma\Omega^2 & (1+\beta\Omega^2) \end{bmatrix} \begin{bmatrix} \left(1 - \left(\frac{1}{2} - \beta\right)\Omega^2\right) & 1 \\ -(1-\gamma)\Omega^2 & 1 \end{bmatrix} \begin{Bmatrix} u_n \\ \Delta t v_n \end{Bmatrix} + \frac{1}{1+\beta\Omega^2} \begin{bmatrix} 1 & 0 \\ -\gamma\Omega^2 & (1+\beta\Omega^2) \end{bmatrix} \begin{Bmatrix} \left(\frac{1}{2} - \beta\right) g_n + \beta g_{n+1} \\ (1-\gamma) g_n + \gamma g_{n+1} \end{Bmatrix} \tag{3.81}$$

The stability of the solution is evaluated from the homogeneous equation by assuming $u_{n+1} = \lambda u_n$ and $v_{n+1} = \lambda v_n$. This yields the algebraic eigenproblem

$$\begin{bmatrix} \left[1 - \left(\frac{1}{2} - \beta\right)\Omega^2 - \bar{\lambda}\right] & 1 \\ \left[-\Omega^2 + \left(\frac{1}{2}\gamma - \beta\right)\Omega^4\right] & [1 + (\beta - \gamma)\Omega^2 - \bar{\lambda}] \end{bmatrix} \begin{Bmatrix} u_n \\ \Delta t v_n \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \tag{3.82}$$

where $\bar{\lambda} = (1 + \beta\Omega^2)\lambda$. A full analysis for all values of $\beta$ and $\gamma$ may be found in Refs. [15,17]. Here we consider only the special cases of $\gamma = 1/2$ with $\beta = 0$ or $\beta = 1/4$.

**Example 3.9.** Implicit integration: $\beta = 1/4$

We first consider the implicit problem with $\gamma = 1/2$ and $\beta = 1/4$. For this case (3.82) simplifies to

$$\begin{bmatrix} \left[\left(1 - \frac{1}{4}\Omega^2\right) - \bar{\lambda}\right] & 1 \\ -\Omega^2 & \left[\left(1 - \frac{1}{4}\Omega^2\right) - \bar{\lambda}\right] \end{bmatrix} \begin{Bmatrix} u_n \\ \Delta t v_n \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}$$

where $\bar{\lambda} = \left(1 + \frac{1}{4}\Omega^2\right)\lambda$. The two values for the $\bar{\lambda}$ are obtained from the characteristic polynomial

$$\left[\left(1 - \frac{1}{4}\Omega^2\right) - \bar{\lambda}\right]^2 + \Omega^2 = 0$$

and yield the result

$$\lambda_j = \frac{\left(1 - \frac{1}{4}\Omega^2\right) \pm i\,\Omega}{\left(1 + \frac{1}{4}\Omega^2\right)} \quad \text{for } j = 1,\ 2$$

The solution is complex and has a modulus value $\rho = \max(|\lambda_1|, |\lambda_2|) = 1$ independent of the value of $\Omega$ (and $\Delta t$). Thus, again we have shown that the use of these parameters yields an unconditionally stable method for second-order, linear differential equations.

**Example 3.10.** Explicit integration: $\beta = 0$

We next consider the explicit problem with $\gamma = 1/2$ and $\beta = 0$. For this case (3.82) simplifies to

$$\begin{bmatrix} \left[\left(1 - \frac{1}{2}\Omega^2\right) - \lambda\right] & 1 \\ -\Omega^2 + \frac{1}{4}\Omega^4 & \left[\left(1 - \frac{1}{2}\Omega^2\right) - \lambda\right] \end{bmatrix} \begin{Bmatrix} u_n \\ \Delta t\, v_n \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}$$

The two values of $\lambda$ may be determined from

$$\left[\left(1 - \frac{1}{2}\Omega^2\right) - \lambda\right]^2 + \Omega^2 - \frac{1}{4}\Omega^4 = 0$$

For this case two possibilities for roots exist depending on the value of $\Omega$:

$$\lambda_j = \begin{cases} 1 - \frac{1}{2}\Omega^2 \pm \left[\frac{1}{4}\Omega^4 - \Omega^2\right]^{1/2} & \text{for } \Omega > 2 \\ 1 - \frac{1}{2}\Omega^2 \pm i\left[\Omega^2 - \frac{1}{4}\Omega^4\right]^{1/2} & \text{for } \Omega < 2 \end{cases}$$

When $\Omega = 2$ the roots are both unity and, thus, are at a limit of stability. For $\Omega > 2$ the value of $\rho$ defined above will be greater than unity and the method is unstable. Thus for stability we must always have

$$\Delta t < \frac{2}{\omega}$$

where $\omega$ is the *largest eigenvalue* from the problem being solved. This value may be estimated by computing the maximum frequency of individual elements [18]. Alternatively, the critical time step can be estimated by a few iterations of a Rayleigh quotient [19].

## 3.9 Weak form for one-dimensional quasi-harmonic equation

In the previous sections we have considered a procedure to solve one-dimensional elasticity problems using a weak form of the governing equations. Here we repeat the process for a one-dimensional quasi-harmonic equation. We consider a problem in Cartesian coordinates where

$$q_x = q_x(x) \quad \text{and} \quad q_y = q_z = 0$$

With this assumption the quasi-harmonic equation is given by

$$-\frac{\partial q_x}{\partial x} + Q = c\,\frac{\partial \phi}{\partial t} \tag{3.83}$$

The one-dimensional form of the material law (2.83) is given by

$$q_x = -k \frac{\partial \phi}{\partial x} \tag{3.84}$$

and the boundary and initial conditions are taken as one-dimensional forms of (2.81a), (2.81b), and (2.82).

### 3.9.1 Weak form

Following the rules given in (3.1) for constructing a weak form we may write the first step as

$$g(q_x,\ \phi,\ \delta\phi) = \delta\phi \left( c \frac{\partial \phi}{\partial t} - Q + \frac{\partial q_x}{\partial x} \right) = 0$$

Integrating over the domain $0 < x < L$ we obtain the integral form

$$\mathrm{G}(q_x,\ \phi,\ \delta\phi) = \int_\Omega \delta\phi \left( c \frac{\partial \phi}{\partial t} - Q + \frac{\partial q_x}{\partial x} \right) \mathrm{d}x = 0$$

Upon integrating the last term by parts the expression becomes

$$\mathrm{G}(q_x,\ \phi,\ \delta\phi) = \int_\Omega \left[ \delta\phi \left( c \frac{\partial \phi}{\partial t} - Q \right) - \frac{\partial \delta\phi}{\partial x} q_x \right] \mathrm{d}x + \delta\phi q_x n_x|_\Gamma = 0$$

Finally, considering the boundary conditions and letting $\phi = \bar{\phi}$ and $\delta\phi = 0$ on $\Gamma_\phi$ and $q_x n_x = \bar{q}_n + H\,(\phi - \phi_0)$ on $\Gamma_q$ we obtain

$$\begin{aligned} \mathrm{G}(q_x,\ \phi,\ \delta\phi) = \int_\Omega \left[ \delta\phi \left( c \frac{\partial \phi}{\partial t} - Q \right) - \frac{\partial \delta\phi}{\partial x} q_x \right] \mathrm{d}x \\ + \delta\phi \left[ \bar{q}_n + H\,(\phi - \phi_0) \right]\big|_{\Gamma_q} = 0 \end{aligned} \tag{3.85}$$

An *irreducible form* is obtained by inserting the one-dimensional form (3.84) into (3.85) to obtain

$$\begin{aligned} \mathrm{G}(\phi,\ \delta\phi) = \int_\Omega \left[ \delta\phi \left( c \frac{\partial \phi}{\partial t} - Q \right) + \frac{\partial \delta\phi}{\partial x} k \frac{\partial \phi}{\partial x} \right] \mathrm{d}x \\ + \delta\phi \left[ \bar{q}_n + H\,(\phi - \phi_0) \right]\big|_{\Gamma_q} = 0 \end{aligned} \tag{3.86}$$

### 3.9.2 Finite element solution of quasi-harmonic problem

A finite element solution to the irreducible one-dimensional quasi-harmonic problem may be constructed in an identical manner to that performed for the elasticity problem. To this end we define a set of nodes $x_i$ on the domain and, similarly, a set of elements using the nodes. Using the isoparametric concept we can define approximations on each element using shape functions. Since only first derivatives in space are contained

in the weak form we may use any of the $C_0$ shape functions defined above for the elasticity problem. Accordingly, we can write for each element

$$x^e = \sum_a N_a(\xi)\, x_a^e,\ \phi^e = \sum_a N_a(\xi)\, \tilde{\phi}_a^e \quad \text{and} \quad \delta\phi^e = \sum_a N_a(\xi)\, \delta\tilde{\phi}_a^e \tag{3.87}$$

Dividing the domain into a set of elements the weak form (3.86) may be written as

$$\mathrm{G}(\phi,\ \delta\phi) = \sum_e \mathrm{G}^e(\phi,\ \delta\phi) + \mathrm{G}^b(\phi,\ \delta\phi) \tag{3.88}$$

which gives the element expression

$$\mathrm{G}^e(\phi,\ \delta\phi) = \sum_a \delta\tilde{\phi}_a^e \left[ \int_{-1}^{1} N_a \left( c \sum_b N_b \frac{\partial \tilde{\phi}_b^e}{\partial t} - Q \right) + \frac{\partial N_a}{\partial x}\, k \sum_b \frac{\partial N_b}{\partial x}\, \tilde{\phi}_b^e \right] j_e \,\mathrm{d}\xi \tag{3.89a}$$

Similarly, the boundary term becomes

$$\mathrm{G}^b(\phi,\ \delta\phi) = \delta\tilde{\phi}_a N_b(x_a) \left[ \bar{q}_n + H\,(N_b(x_a)\tilde{\phi}_b - \phi_0) \right]\Big|_{x_a \in \Gamma_q} \tag{3.89b}$$

The integrals may now be evaluated to give the element matrix expressions

$$\begin{aligned} C_{ab}^e &= \int_{-1}^{1} N_a\, c\, N_b\, j_e \,\mathrm{d}\xi \\ H_{ab}^e &= \int_{-1}^{1} \frac{\partial N_a}{\partial x}\, k\, \frac{\partial N_b}{\partial x}\, j_e \,\mathrm{d}\xi \quad \text{and} \\ s_a^e &= \int_{-1}^{1} N_a\, Q\, j_e \,\mathrm{d}\xi \end{aligned} \tag{3.90}$$

Similarly, considering the boundary term we obtain

$$\begin{aligned} H_{ab}^b &= N_a(x_a)\, H\, N_b(x_a)|_{x_a \in \Gamma_q} \quad \text{and} \\ s_a^b &= N_a(x_a)\left[ -\bar{q}_n + H\,\phi_0 \right]\Big|_{x_a \in \Gamma_q} \end{aligned} \tag{3.91}$$

where we note the boundary radiation term contributes to both the coefficient matrix and the loading term. Inserting the matrices into (3.89a) and (3.89b) gives

$$\begin{aligned} &\mathrm{G}^e(\phi,\ \delta\phi) + \mathrm{G}^b(\phi,\ \delta\phi) \\ &\quad = \sum_a \delta\phi_a^e \left[ \sum_b \left( C_{ab}^e \frac{\partial \tilde{\phi}_b^e}{\partial t} + \left( H_{ab}^e + H_{ab}^b \right) \tilde{\phi}_b^e \right) - \left( s_a^e + s_a^b \right) \right] \end{aligned} \tag{3.92}$$

Performing the summation indicated in (3.88) yields

$$\mathrm{G}(\boldsymbol{\phi},\ \delta\boldsymbol{\phi}) = \delta\tilde{\boldsymbol{\phi}}^{\mathrm{T}} \left[ \mathbf{C} \frac{\partial \tilde{\boldsymbol{\phi}}}{\partial t} + \mathbf{H}\,\tilde{\boldsymbol{\phi}} - \mathbf{s} \right] = 0 \tag{3.93}$$

in which $\mathbf{s}$ and $\mathbf{H}$ contain also the contribution from the boundary terms on $\Gamma_q$ and modification for any $\Gamma_\phi$ conditions are also imposed. Since the variation $\delta\tilde{\boldsymbol{\phi}}$ is arbitrary the above expression again leads to a standard linear problem

$$\mathbf{C}\frac{\partial\tilde{\boldsymbol{\phi}}}{\partial t} + \mathbf{H}\,\tilde{\boldsymbol{\phi}} = \mathbf{s} \tag{3.94}$$

If the loading $\mathbf{s}$ is applied very slowly, the transient term may be neglected and a *steady-state* solution evaluated from

$$\mathbf{H}\,\tilde{\boldsymbol{\phi}} = \mathbf{s} \tag{3.95}$$

Once a solution is known the flux in each element may be determined from (3.84).

### 3.9.3 Transient problems

Transient problems may be solved by discretizing the time derivative in (3.94) in a manner similar to that for the elasticity problem. Here, however, only a first derivative in time exists. An approximate solution at each discrete time $t_n$ between $t = 0$ and $t = T$ is denoted by

$$(\tilde{\boldsymbol{\phi}}(t_n),\ \dot{\tilde{\boldsymbol{\phi}}}(t_n)) \approx (\tilde{\boldsymbol{\phi}}_n,\ \dot{\tilde{\boldsymbol{\phi}}}_n)$$

Similar to the Newmark formulas we consider the approximation

$$\tilde{\boldsymbol{\phi}}_{n+1} = \tilde{\boldsymbol{\phi}}_n + (1-\theta)\,\Delta t\,\dot{\tilde{\boldsymbol{\phi}}}_n + \theta\,\Delta t\,\dot{\tilde{\boldsymbol{\phi}}}_{n+1} \tag{3.96}$$

where $\theta$ is a parameter that may range between 0 and 1. A discrete approximation to (3.94) is given by

$$\mathbf{C}\,\dot{\tilde{\boldsymbol{\phi}}}_{n+1} + \mathbf{H}\,\tilde{\boldsymbol{\phi}}_{n+1} = \mathbf{s}_{n+1} \tag{3.97}$$

Substituting (3.96) into (3.97) yields the algebraic equations

$$\left[\mathbf{C} + \theta\,\Delta t\,\mathbf{H}\right]\dot{\tilde{\boldsymbol{\phi}}}_{n+1} = \mathbf{s}_{n+1} - \mathbf{H}\,\widehat{\boldsymbol{\phi}}_{n+1} \tag{3.98}$$

where

$$\widehat{\boldsymbol{\phi}}_{n+1} = \tilde{\boldsymbol{\phi}}_n + (1-\theta)\,\Delta t\,\dot{\tilde{\boldsymbol{\phi}}}_n$$

#### 3.9.3.1 *Stability*

The stability of the discrete approximation may be determined from a single homogeneous scalar equation

$$c\dot{\phi} + h\phi = 0$$

Evaluating this equation at $t_n$ and $t_{n+1}$ and substituting into (3.96) gives the recurrence

$$\phi_{n+1} = \frac{[1-(1-\theta)\,\lambda\Delta t]}{[1+\theta\,\lambda\Delta t]}\,\phi_n$$

where $\lambda = h/c$. The exact solution to the scalar equation is $\exp(-\lambda t)$. Thus, the approximate solution should always produce solutions where $|\phi_{n+1}| < |\phi_n|$. A simple calculation reveals that this is always true for any $\Delta t$ if $\theta > 0.5$ and thus the solution

is unconditionally stable. For values of $\theta < 0.5$ the solution is only stable for time increments less than a critical value. For example, if $\theta = 0$ we obtain

$$\phi_{n+1} = [1 - \lambda\, \Delta t]\, \phi_n$$

and $\Delta t_{cr} = 2/\lambda$.

For the multi-degree-of-freedom problem it is necessary to compute the *largest* $\lambda$ from the eigenproblem

$$\mathbf{C\Phi} = \mathbf{H\Phi}\Lambda$$

Generally, however, we will find it advantageous to use unconditionally stable forms with $\theta \geq 1/2$ when solving quasi-harmonic equations forms.

**Example 3.11.** One-dimensional steady-state heat conduction

The problem here will be a one-dimensional representation of the heat conduction equation [Eq. (3.94)] with unit conductivity (this problem could equally well represent many other physical situations, e.g., deflection of a loaded string with unit tension). Here we let (see Fig. 3.9)

$$-\frac{\mathrm{d}^2\phi}{\mathrm{d}x^2} + Q(x) = 0 \quad (0 < x < L)$$

with $Q(x)$ given by

$$Q(x) = \begin{cases} 0 & 0 < x \leq L/2 \\ -2\, Q_0\, (x/L - 1/2) & L/2 < x < L \end{cases}$$

The boundary conditions assumed will be simply $\phi = 0$ at $x = 0$ and $x = L$.

The problem is first solved using two-node isoparametric elements to form $\mathbf{H}$ and $\mathbf{s}$. Assembly of four equal size elements results, after inserting the boundary conditions $\tilde{\phi}_1 = \tilde{\phi}_5 = 0$, in the equation set

$$\frac{4}{L}\begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix} \begin{Bmatrix} \tilde{\phi}_2 \\ \tilde{\phi}_3 \\ \tilde{\phi}_4 \end{Bmatrix} = \frac{Q_0 L}{48} \begin{Bmatrix} 0 \\ 1 \\ 6 \end{Bmatrix}$$

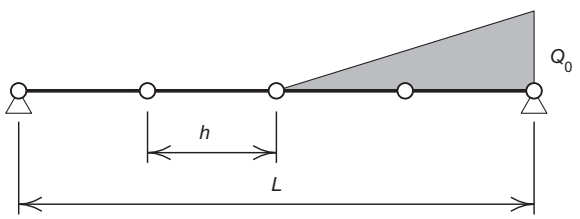



**FIGURE 3.9**
Problem description and loading for 1-D heat conduction example.

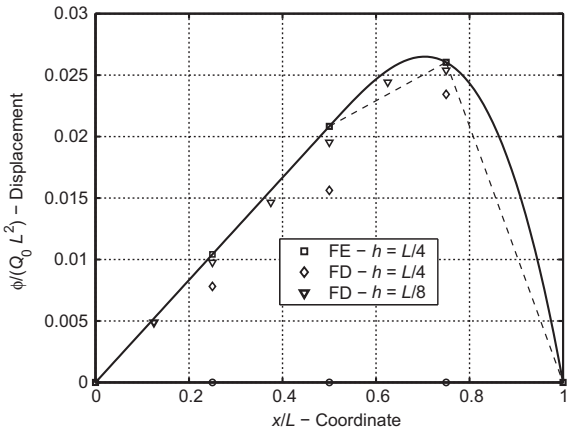



**FIGURE 3.10**

One-dimensional heat conduction. Solution by finite element method with linear elements and $h = L/4$; finite difference method with $h = L/4$ and $h = L/8$.

The solution using two-node elements is shown in Fig. 3.10 along with the exact solution to the problem. For comparison purposes we also show a finite difference solution in which simple collocation is used in a weighted residual equation together with the approximation for the second derivative given by a Taylor expansion

$$\left.\frac{\mathrm{d}^2\phi}{\mathrm{d}x^2}\right|_{x_a} \approx \frac{1}{h^2}\left(\tilde{\phi}_{a-1} - 2\tilde{\phi}_a + \tilde{\phi}_{a+1}\right)$$

which yields the approximation for each node point

$$\frac{1}{h^2}\left(-\tilde{\phi}_{a-1} + 2\tilde{\phi}_a - \tilde{\phi}_{a+1}\right) + Q_a = 0$$

After including the boundary conditions a set of three equations for the points 2, 3, and 4 is expressed as

$$\frac{16}{L^2}\begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}\begin{Bmatrix} \tilde{\phi}_2 \\ \tilde{\phi}_3 \\ \tilde{\phi}_4 \end{Bmatrix} = Q_0 \begin{Bmatrix} 0 \\ 0 \\ 1/2 \end{Bmatrix}$$

The reader will note that the coefficient matrix for the finite element and finite difference methods differs by only a constant multiplier (for the boundary conditions assumed in this one-dimensional problem); however, the right-hand sides differ significantly. Here we note that the nodal results for the finite element method are *exact* whereas those for the finite difference solution are all in error (although convergence can be observed for the finer subdivision). The nodal exactness is a property of the particular equation being solved and unfortunately does not carry over to general problems [20] (see also Appendix G). However, based on the above result

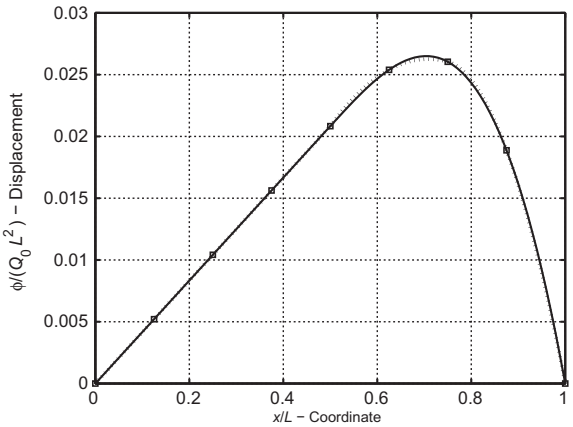



**FIGURE 3.11**
One-dimensional heat conduction. Solution by finite element method with quadratic elements and $h = L/4$.

and other experiences we can say that the finite element method always achieves (the same or) better results than classical finite difference methods. In addition, the finite element method permits an approximation of the solution *at all points in the domain* as indicated by the dashed lines in Fig. 3.10 for the one-dimensional problem.

The problem is repeated using 4-quadratic order finite elements and the results are shown in Fig. 3.11. It is evident that the use of quadratic order greatly increases the accuracy of the results obtained. Indeed, if cubic order elements were used results would be exact, since for linear varying $Q$ the solution over the loaded portion will only contain polynomials up to cubic order.

## 3.10 Concluding remarks

In this chapter we have presented all the steps necessary to perform a finite element analysis of any problem for which a differential equation is known. The presentation has been simplified to equations with one coordinate for simplicity. However, as we shall observe in the later chapters, the basic steps used are general and apply equally to higher dimensional problems.

We also include in the development the possibility of considering transient problems. Using weak forms as the basis for a finite element discretization no additional complexity arises and we can consider both transient and static (steady-state) applications. The solution to transient problems may be addressed using discrete time methods, such as the Newmark method, or for linear forms by methods based on semi-discretization using modal methods.

The list of problem types that can be considered based on the presentation in this chapter is large and the examples presented cover only a small portion of possibilities.

In the next chapter we present an alternative to weak forms which may also be used with finite element approximation. Once both of these chapters are fully understood, the reader should be able to consider many additional classes of problems. In subsequent chapters we concentrate on problems in solid and structural mechanics and develop the methodology to construct the necessary finite element approximations for a wide class of applications.

## 3.11 Problems

**3.1** Write weak forms for the following differential equations and boundary conditions. For each form state appropriate continuity conditions for approximations to the dependent variable $u$ and the weighting function $v$. The domain for each one-dimensional differential equation is $0 < x < 1$.

**(a)**

$$a\frac{\mathrm{d}u}{\mathrm{d}x} + cu + q = 0, \quad u(0) = \bar{g}$$

**(b)**

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(a\frac{\mathrm{d}u}{\mathrm{d}x}\right) + q = 0, \quad u(0) = \bar{g} \quad \text{and} \quad a\frac{\mathrm{d}u}{\mathrm{d}x} + ku = \bar{g}, \ \text{at } x = 1$$

**(c)**

$$-\frac{\mathrm{d}}{\mathrm{d}x}\left(a\frac{\mathrm{d}u}{\mathrm{d}x}\right) + b\frac{\mathrm{d}u}{\mathrm{d}x} + q = 0, \quad u(0) = \bar{g}_0, \quad u(1) = \bar{g}_2$$

**(d)**

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(a\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right) + f = 0,$$
$$u(0) = \bar{g}_0, \ \left.\frac{\mathrm{d}u}{\mathrm{d}x}\right|_{x=0} = \bar{h}_0 \quad \text{and} \quad u(1) = \bar{g}_1$$

**3.2** The differential equations for bending of a beam are given by

$$(1)\ \frac{\mathrm{d}V}{\mathrm{d}x} + q = 0 \qquad (2)\ \frac{\mathrm{d}M}{\mathrm{d}x} + V = 0$$
$$(3)\ \frac{\mathrm{d}\theta}{\mathrm{d}x} - \frac{M}{EI} = 0 \quad (4)\ \frac{\mathrm{d}w}{\mathrm{d}x} - \theta - \frac{V}{GA} = 0$$

in which $V$ is shear force, $M$ is moment, $\theta$ is section rotation, $w$ is displacement, $EI$ is bending stiffness, $GA$ is shear stiffness, and $q$ is load as shown in Fig. 3.12.

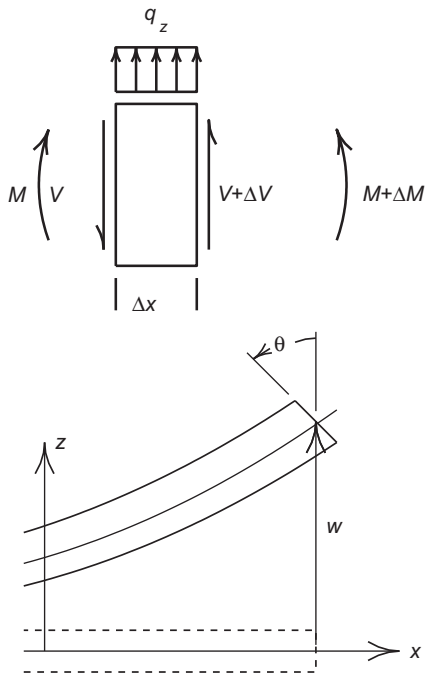



**FIGURE 3.12**
Beam bending description.

Boundary conditions are given by

$$(1)\ V = \overline{V} \quad \text{or } w = \bar{w}$$
$$(2)\ M = \overline{M} \quad \text{or } \theta = \bar{\theta}$$

Construct a weak form for the beam equations by multiplying (1) by $\pm\delta w$, (2) by $\pm\delta\theta$, (3) by $\delta M$, and (4) by $\delta V$.
Choose the correct sign for $\delta w$ and $\delta\theta$ to give symmetry.

**3.3** Add all boundary conditions to the weak form obtained in Problem 3.2.

**3.4** For $GA = \infty$ (no shear deformation) deduce the irreducible differential equation in terms of $w$. Express all boundary conditions in terms of $w$.

**3.5** Construct a weak form for Problem 3.4. What is the required continuity of the dependent variable needed for approximation by a finite element method? What are the natural and essential boundary conditions for the weak form?

**3.6** For $GA = \infty$ (no shear deformation) deduce the differential equations in terms of $w$ and $M$. Express all boundary conditions in terms of these variables.

**3.7** Deduce a weak form for Problem 3.6 that permits approximation using $C_0$ functions to approximate $w$ and $M$. Let

$$w = \sum_{a=1}^{2} N_a \tilde{w}_a \quad \text{and} \quad M = \sum_{a=1}^{2} N_a \tilde{M}_a$$

where $N_a$ are given by (3.46). Ensure your weak form gives a symmetric coefficient matrix for these approximations.
Compute typical element matrices $\mathbf{K}$ and $\mathbf{f}$ for an element of length $h$ with constant $EI$ and $q$ in the element.

**3.8** For a simply supported beam of length 10 and constant cross-section $EI = 3$ compute the solution for a uniform load of $q = 1$. The boundary conditions at each end of the beam for a simple support are $w = M = 0$. Obtain a solution using 2, 4, and 8 elements. It is recommended that a small computer program be written using a high level language, e.g. MATLAB [21] or GNU Octave [22], to perform the numerical calculations. Compare your results to an exact solution.

**3.9** The transient heat equation in one-dimension is given by

$$-\frac{\partial}{\partial x}\left(k\,\frac{\partial \phi}{\partial x}\right) + Q + c\,\frac{\partial \phi}{\partial t} = 0$$

where $\phi$ is temperature, $k$ is thermal conductivity, $Q$ is heat generation per unit length, and $c$ is specific heat.
Boundary conditions may be given as

$$\phi = \bar{\phi} \quad \text{on} \quad \Gamma_1 \quad \text{or} \quad q = -k\,\frac{\partial \phi}{\partial x} = \bar{q} \quad \text{on} \quad \Gamma_2$$

where $q$ is the heat flux and $\bar{\phi},\ \bar{q}$ are specified values. Initial conditions are given as $\phi(x,\ 0) = \bar{\phi}_0(x)$.

**(a)** Construct a weak form for the problem.

**(b)** Using the shape functions given in Eq. (3.26a) and the approximation

$$u^e = N_1(x)\tilde{u}_1(t) + N_2(x)\tilde{u}_2(t)$$
$$\delta u^e = N_1(x)\delta\tilde{u}_1 + N_2(x)\delta\tilde{u}_2$$

construct the semi-discrete form for a typical element of length $h$.

**(c)** Consider a region of length 10, with properties $k = 5, c = 1, Q = 0$. Divide the region into four equal length elements and establish the set of global semi-discrete equations.

**(d)** Consider a set of discrete times $t_n$. Approximate time derivatives of nodal values by $\mathrm{d}\phi/\mathrm{d}t(t_n) \approx (\phi_n - \phi_{n-1})/\Delta t$ where $\phi_n$ is the approximation to $\phi(t_n)$ and $\Delta t = t_n - t_{n-1}$ and write the fully discrete equations.
Write a computer program (e.g., using MATLAB or GNU Octave) to solve the problem. Assume the initial temperature of the region is zero and boundary conditions $\phi(0) = 0$ and $\phi(10) = 1$ are applied at time zero and held constant. Solve the problem using 10 steps with $\Delta t = 0.01$, followed by 9 steps with $\Delta t = 0.1$ and finally 9 steps with $\Delta t = 1$. Plot the finite element solution for $\phi$ vs. $x$ at times 0.01, 0.1, 1.0, and 10.0.
Replace the element matrix associated with $c$ by a diagonal (lumped) form with $c\,h/2$ on each diagonal ($h = x_2^e - x_1^e$). Repeat the above solution and compare results with the consistent form for the matrix.

## References

[1] O.C. Zienkiewicz, W.L. Wood, N.W. Hine, R.L. Taylor, A unified set of single-step algorithms. Part 1: general formulation and applications, International Journal for Numerical Methods in Engineering 20 (1984) 1529–1552.
[2] W.L. Wood, A unified set of single-step algorithms. Part 2: theory, International Journal for Numerical Methods in Engineering 20 (1984) 2302–2309.
[3] M. Katona, O.C. Zienkiewicz, A unified set of single-step algorithms. Part 3: the beta-m method, a generalization of the Newmark scheme, International Journal for Numerical Methods in Engineering 21 (1985) 1345–1359.
[4] G.E. Forsythe, W.R. Wasow, Finite Difference Methods for Partial Differential Equations, John Wiley & Sons, New York, 1960.
[5] F.B. Hildebrand, Finite Difference Equations and Simulations, Prentice-Hall, Englewood Cliffs, N.J., 1968.
[6] F.B. Hildebrand. Introduction to Numerical Analysis, Dover Publishers, New York, second ed., 1987.
[7] A.R. Mitchell, D. Griffiths, The Finite Difference Method in Partial Differential Equations, John Wiley & Sons, London, 1980.
[8] R.A. Frazer, W.P. Jones, and S.W. Sken. Approximations to functions and to the solution of differential equations, Technical Report 1799, Aero. Research Committee, Report, 1937.
[9] C.B. Biezeno, R. Grammel, Technishe Dynamik, Springer-Verlag, Berlin, 1933, p. 142.
[10] O.C. Zienkiewicz, E. Oñate, Finite elements versus finite volumes. Is there a choice? in: P Wriggers, W Wagner (Eds.), Nonlinear Computational Mechanics, State of the Art, Springer, Berlin, 1991.
[11] B.G. Galerkin, Series solution of some problems in elastic equilibrium of rods and plates, Vestnik Inzhenerov i Tekhnikov 19 (1915) 897–908.
[12] W. Rudin, Principles of Mathematical Analysis, McGraw-Hill, New York, third ed., 1976.
[13] O.C. Zienkiewicz, R.L. Taylor, P. Nithiarasu, The Finite Element Method for Fluid Dynamics, Elsevier, Oxford, seventh ed., 2013.
[14] N. Newmark, A method of computation for structural dynamics, Journal of Engineering Mechanics ASCE 85 (1959) 67–94.
[15] G.L. Goudreau, R.L. Taylor, Evaluation of numerical integration methods in elastodynamics, Computer Methods in Applied Mechanics and Engineering 2 (1972) 69–97.
[16] T.J.R. Hughes, The Finite Element Method: Linear Static and Dynamic Analysis, Dover Publications, New York, 2000.
[17] T.J.R. Hughes, Analysis of transient algorithms with particular reference to stability behavior, in: T Belytschko, TJR Hughes (Eds.), Computational Methods for Transient Analysis, North-Holland, Amsterdam, 1983.

[18] T.J.R. Hughes, K.S. Pister, R.L. Taylor, Implicit-explicit finite elements in nonlinear transient analysis, Computer Methods in Applied Mechanics and Engineering 17–18 (1979) 159–182.
[19] D.J. Benson, Stable time step estimation for multi-material Eulerian hydrocodes, Computer Methods in Applied Mechanics and Engineering 167 (1998) 191–205.
[20] P. Tong, Exact solution of certain problems by the finite element method, Journal of AIAA 7 (1969) 179–180.
[21] MATLAB, 2012. <www.mathworks.com>.
[22] GNU Octave, 2012. <www.gnu.org/software/octave>.

CHAPTER

# 4 Variational Forms and Finite Element Approximation: 1-D Problems

## 4.1 Variational principles

In the previous chapter we described a weak form as an integral expression of a differential equation. For problems in which time derivative terms are not present an alternative integral expression known as a *variational principle* often exists. Variational principles can also be used as a basis to construct finite element solutions.

First, a definition: A "variational principle" specifies a scalar quantity (a functional) $\Pi$, which for the one-dimensional problems considered here is defined by an integral form

$$\Pi = \int_{\Omega} F\left(u, \frac{\mathrm{d}u}{\mathrm{d}x}, \dots\right)\mathrm{d}x + E\left(u, \frac{\mathrm{d}u}{\mathrm{d}x}, \dots\right)\Big|_{\Gamma} \tag{4.1}$$

in which $u$ is the unknown function and $F$ and $E$ are specified differential operators. The solution to the continuous problem is a function $u$ which makes $\Pi$ *stationary* with respect to arbitrary changes $\delta u$. Thus, for a solution to the continuous problem, the "variation" is

$$\delta\Pi = 0 \tag{4.2}$$

for any $\delta u$, which defines the condition of stationarity [1].

If a "variational principle" can be found, then means are immediately established for obtaining approximate solutions in the standard, integral form suitable for finite element analysis.

Dividing the domain into elements and assuming a trial function expansion in the usual form [Eq. (3.37)]

$$u^e \approx \hat{u}^e = \sum_{a=1}^{n} N_a \tilde{u}_a = \mathbf{N}\tilde{\mathbf{u}}$$

we can insert this into (4.1) and write

$$\begin{aligned} \delta\Pi &= \sum_{e=1}^{M} \delta\Pi^e = 0 \\ \delta\Pi^e &= \frac{\partial\Pi^e}{\partial\tilde{u}_1}\delta\tilde{u}_1 + \frac{\partial\Pi^e}{\partial\tilde{u}_2}\delta\tilde{u}_2 + \cdots + \frac{\partial\Pi^e}{\partial\tilde{u}_n}\delta\tilde{u}_n \end{aligned} \tag{4.3}$$

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00004-6

This being true for any variations $\delta\tilde{\mathbf{u}}$ yields a set of equations for the system

$$\frac{\partial \Pi}{\partial \tilde{\mathbf{u}}} = \begin{Bmatrix} \dfrac{\partial \Pi}{\partial \tilde{u}_1} \\ \vdots \\ \dfrac{\partial \Pi}{\partial \tilde{u}_n} \end{Bmatrix} = \mathbf{0} \tag{4.4}$$

from which parameters $\tilde{u}_a$ are found. The equations are of an integral form necessary for the finite element approximation as the original specification of $\Pi$ was given in terms of domain and boundary integrals.

The process of finding stationarity with respect to trial function parameters $\tilde{\mathbf{u}}$ is an old one and is associated with the names of Rayleigh [2] and Ritz [3].

If the functional $\Pi$ is "quadratic," i.e., if the function $u$ and its derivatives occur in powers not exceeding 2, then (4.4) reduces to a standard linear form similar to (3.20), i.e.,

$$\frac{\partial \Pi}{\partial \tilde{\mathbf{u}}} \equiv \mathbf{K}\tilde{\mathbf{u}} - \mathbf{f} = \mathbf{0} \tag{4.5}$$

It is easy to show that the matrix $\mathbf{K}$ will now always be symmetric. To do this let us consider a linearization of the vector $\partial\Pi/\partial\tilde{\mathbf{u}}$. This we can write as

$$\Delta\left(\frac{\partial \Pi}{\partial \tilde{\mathbf{u}}}\right) = \begin{Bmatrix} \frac{\partial}{\partial \tilde{u}_1}\left(\frac{\partial \Pi}{\partial \tilde{u}_1}\right)\Delta\tilde{u}_1 + \frac{\partial}{\partial \tilde{u}_2}\left(\frac{\partial \Pi}{\partial \tilde{u}_1}\right)\Delta\tilde{u}_2 + \cdots \\ \vdots \end{Bmatrix} \equiv \mathbf{K}_{\mathrm{T}}\Delta\tilde{\mathbf{u}} \tag{4.6}$$

in which $\mathbf{K}_{\mathrm{T}}$ is generally known as the tangent matrix, of significance in nonlinear analysis, and $\Delta\tilde{\mathbf{u}}$ are small incremental changes to $\tilde{\mathbf{u}}$. Now it is easy to see that

$$K_{\mathrm{T}ab} = \frac{\partial^2 \Pi}{\partial \tilde{u}_a \partial \tilde{u}_b} = K_{\mathrm{T}ba} \tag{4.7}$$

Hence $\mathbf{K}_{\mathrm{T}}$ is symmetric.

For a quadratic functional we have, from (4.5), $\mathbf{K}_{\mathrm{T}} = \mathbf{K}$, a constant matrix.

The fact that *symmetric matrices will arise whenever a variational principle exists is one of the most important merits of variational approaches for discretization*. However, symmetric forms will frequently arise directly from the Galerkin process. In such cases we simply conclude that the variational principle exists but we shall not need to use it directly since then it is automatically known that

$$\delta\Pi(u, \delta u) \equiv \mathrm{G}(u, \delta u) = 0 \tag{4.8}$$

Further, the discovery of symmetry from a weighted residual process leads directly to variational principles [4–6].

We note that frequently the physical aspects of a problem can be stated directly in a variational principle form. Theorems such as minimization of total potential energy

to achieve equilibrium in mechanical systems, least energy dissipation principles in viscous flow, etc., may be known to the reader and are considered by many as the basis of the formulation. Variational principles of this kind are called "natural" ones but unfortunately they do not exist for all problems for which well-defined differential equations can be formulated.

There is another category of variational principles which we may call "contrived" and these can always be constructed for any specified problem, either by extending the number of unknown functions $u$ using additional variables known as Lagrange multipliers, or by procedures imposing a higher degree requirement on continuity, such as in least squares problems. In subsequent sections we shall discuss some "natural" and "contrived" variational principles.

Before proceeding further it is worth noting that, in addition to symmetry occurring in equations derived by variational means, sometimes further motivation arises. When "natural" variational principles exist the quantity $\Pi$ may be of specific interest itself. If this arises a variational approach possesses the merit of easy evaluation of this quantity.

The reader will observe that if the functional is "quadratic" and yields (4.5), then we can write the approximate "functional" $\Pi$ simply as

$$\Pi = \tfrac{1}{2}\tilde{\mathbf{u}}^{\mathrm{T}}\mathbf{K}\tilde{\mathbf{u}} - \tilde{\mathbf{u}}^{\mathrm{T}}\mathbf{f} \tag{4.9}$$

Hence, upon inserting the solution

$$\mathbf{K}\tilde{\mathbf{u}} - \mathbf{f} = \mathbf{0}$$

into (4.9) we obtain

$$\Pi = -\frac{1}{2}\tilde{\mathbf{u}}^{\mathrm{T}}\mathbf{f} = -\frac{1}{2}\tilde{\mathbf{u}}^{\mathrm{T}}\mathbf{K}\tilde{\mathbf{u}}$$

## 4.2 "Natural" variational principles and their relation to governing differential equations

### 4.2.1 Euler equations

If we consider the definitions of (4.1) and (4.2) we observe that for stationarity we can write, after performing some differentiations and integrations by parts,

$$\delta\Pi = \int_{\Omega} \delta u \mathcal{A}(u)\mathrm{d}x + \delta u \mathcal{B}(u)|_{\Gamma} = 0 \tag{4.10}$$

As the above has to be true for any variations $\delta u$, we must have

$$\mathcal{A}(u) = 0 \quad \text{in } \Omega \quad \text{and} \quad \mathcal{B}(u) = 0 \quad \text{on } \Gamma \tag{4.11}$$

If $\mathcal{A}$ corresponds precisely to the differential equations governing a problem of interest and $\mathcal{B}$ to its boundary conditions, then the variational principle is a *natural* one. Similar

to the result from a weak form, (4.11) are known as the *Euler equations* corresponding to the variational principle requiring the stationarity of $\Pi$. It is easy to show that for any variational principle a corresponding set of Euler equations can be established. The reverse is unfortunately not true, i.e., only certain forms of differential equations are Euler equations of a variational functional. In the next section we shall consider the conditions necessary for the existence of variational principles and give a prescription for establishing the $\Pi$ from a set of suitable linear differential equations. In this section we shall continue to assume that the form of the variational principle is known.

**Example 4.1.** Heat equation in one dimension

To illustrate the process let us now consider a specific example. Suppose we specify a problem by requiring the stationarity of a functional

$$\Pi = \int_\Omega \left[ \frac{1}{2} k \left( \frac{\mathrm{d}\phi}{\mathrm{d}x} \right)^2 - Q\phi \right] \mathrm{d}x + \left[ \bar{q}_n \phi + H \left( \frac{1}{2}\phi^2 - \phi_0 \phi \right) \right] \Big|_{\Gamma_q} \tag{4.12}$$

in which $k$ and $Q$ depend only on position and we assume $\phi = \bar{\phi}$ is satisfied on $\Gamma_\phi$. We now perform the variation. This can be written following the rules of differentiation as [1]

$$\delta\Pi = \int_\Omega \left[ k \frac{\mathrm{d}\phi}{\mathrm{d}x} \delta \left( \frac{\mathrm{d}\phi}{\mathrm{d}x} \right) - Q\delta\phi \right] \mathrm{d}x + \delta\phi \left[ \bar{q}_n + H(\phi - \phi_0) \right] \Big|_{\Gamma_q} = 0 \tag{4.13}$$

As

$$\delta \left( \frac{\mathrm{d}\phi}{\mathrm{d}x} \right) = \frac{\mathrm{d}}{\mathrm{d}x} (\delta\phi)$$

we can integrate by parts (as in Section 3.9) and, since $\delta\phi = 0$ on $\Gamma_\phi$, obtain

$$\delta\Pi = \int_\Omega \delta\phi \left[ -\frac{\mathrm{d}}{\mathrm{d}x} \left( k \frac{\mathrm{d}\phi}{\mathrm{d}x} \right) - Q \right] \mathrm{d}x + \delta\phi \left( k \frac{\mathrm{d}\phi}{\mathrm{d}n} + \bar{q}_n + H(\phi - \phi_0) \right) \Big|_{\Gamma_q} = 0 \tag{4.14a}$$

This is of the form of (4.10) and we immediately observe that the Euler equations are

$$\begin{aligned} \mathcal{A}(\phi) &= -\frac{\mathrm{d}}{\mathrm{d}x} \left( k \frac{\mathrm{d}\phi}{\mathrm{d}x} \right) - Q = 0 \quad \text{in } \Omega \\ \mathcal{B}(\phi) &= k \frac{\mathrm{d}\phi}{\mathrm{d}n} + \bar{q}_n + H(\phi - \phi_0) = 0 \quad \text{on } \Gamma_q \end{aligned} \tag{4.14b}$$

If $\phi$ is prescribed so that $\phi = \bar{\phi}$ on $\Gamma_\phi$ and $\delta\phi = 0$ on that boundary, then the problem is precisely the one we have already discussed in Section 3.9 and the functional (4.12) specifies the one-dimensional *quasi-harmonic form* for the heat conduction problem in an alternative way.

In this case we have "guessed" the functional but the reader will observe that the variation operation could have been carried out for any functional specified and corresponding *Euler* equations could have been established.

**Example 4.2.** Variational finite element solution

Let us continue the process to obtain an approximate solution of the linear heat conduction problem for the case where the convection coefficient $H$ is zero. Taking, as usual,

$$\phi^e \approx \hat{\phi}^e = \sum_a N_a \tilde{\phi}_a = \mathbf{N}\tilde{\boldsymbol{\phi}} \tag{4.15}$$

we substitute this approximation into the expression for the functional $\Pi$ [Eq. (4.12)] and obtain

$$\Pi^e = \int_{\Omega_e} \frac{1}{2} k \left( \sum_a \frac{\mathrm{d}N_a}{\mathrm{d}x} \tilde{\phi}_a \right)^2 \mathrm{d}x - \int_{\Omega_e} Q \sum_a N_a \tilde{\phi}_a \mathrm{d}x + \bar{q}_n \sum_a N_a \tilde{\phi}_a \Big|_{\Gamma_{qe}} \tag{4.16}$$

On differentiation with respect to a typical parameter $\tilde{\phi}_b$ we have

$$\frac{\mathrm{d}\Pi^e}{\mathrm{d}\tilde{\phi}_b} = \int_{\Omega_e} k \left( \sum_a \frac{\mathrm{d}N_a}{\mathrm{d}x} \tilde{\phi}_a \right) \frac{\mathrm{d}N_b}{\mathrm{d}x} \mathrm{d}x - \int_{\Omega_e} Q N_b \mathrm{d}x + \bar{q}_n N_b |_{\Gamma_{qe}} \tag{4.17}$$

where after summing over all the elements and setting the $\delta\Pi = 0$ we obtain the system of equations for the solution of the problem

$$\mathbf{H}\tilde{\boldsymbol{\phi}} - \mathbf{s} = \mathbf{0} \tag{4.18}$$

with element arrays given by

$$\begin{aligned} H^e_{ab} &= \int_{\Omega_e} k \frac{\mathrm{d}N_a}{\mathrm{d}x} \frac{\mathrm{d}N_b}{\mathrm{d}x} \mathrm{d}x = H^e_{ba} \\ s^e_b &= \int_{\Omega_e} N_b Q \mathrm{d}x - N_b \bar{q}_n |_{\Gamma_{qe}} \end{aligned} \tag{4.19}$$

The reader will observe that the approximation equations are here identical to those obtained in Section 3.9.2 for the same problem using the Galerkin process. No special advantage accrues to the variational formulation here, and indeed we can predict now that *Galerkin and variational procedures must give the same answer for cases where natural variational principles exist.*

## 4.3 Establishment of natural variational principles for linear, self-adjoint differential equations

General rules for deriving natural variational principles from nonlinear differential equations are complicated and even the tests necessary to establish the existence of such variational principles are not simple. Much mathematical work has been done in this context by Vainberg [7], Tonti [6], Oden [8,9], and others.

For linear differential equations the situation is much simpler and a study is available in the works of Mikhlin [4,5], and in this section a brief presentation of such rules is given.

We shall consider here only the establishment of variational principles for a linear system of equations with *essential* (forced) boundary conditions, implying only variation of functions which yield $\delta\mathbf{u} = \mathbf{0}$ on their boundaries.

Writing a linear system of differential equations as

$$\mathcal{A}(u) \equiv \mathcal{L}u + b = 0 \tag{4.20}$$

in which $\mathcal{L}$ is a linear differential operator, natural variational principles require that the operator $\mathcal{L}$ be such that

$$\int_\Omega \psi(\mathcal{L}\gamma)\mathrm{d}x = \int_\Omega \gamma(\mathcal{L}\psi)\mathrm{d}x + \text{b.t.} \tag{4.21}$$

for any two function sets $\psi$ and $\gamma$. In the above, "b.t." stands for boundary terms which we disregard in the present context. The property required in the above operator is called *self-adjointness* or *symmetry*, which we mentioned in Section 3.2.1.

If the operator $\mathcal{L}$ is self-adjoint, the variational principle can be written immediately as

$$\Pi = \int_\Omega \left[\frac{1}{2}u(\mathcal{L}u) + ub\right]\mathrm{d}x + \text{b.t.} \tag{4.22}$$

To prove the veracity of the last statement a variation needs to be considered. We thus write (omitting boundary terms)

$$\delta\Pi = \int_\Omega \left[\frac{1}{2}\delta u(\mathcal{L}u) + \frac{1}{2}u\delta(\mathcal{L}u) + \delta ub\right]\mathrm{d}x = 0 \tag{4.23}$$

Noting that for any linear operator

$$\delta(\mathcal{L}u) \equiv \mathcal{L}\delta u \tag{4.24}$$

and that $u$ and $\delta u$ can be treated as the independent functions, by identity (4.21) we can write (4.23) as

$$\delta\Pi = \int_\Omega \delta u[\mathcal{L}u + b]\mathrm{d}x = 0 \tag{4.25}$$

We observe immediately that the term in the brackets, i.e., the Euler equation of the functional, is identical with the original equation postulated, and therefore the variational principle is verified.

The above gives a very simple test and a prescription for the establishment of natural variational principles for differential equations of the problem.

**Example 4.3.** Helmholtz problem in one-dimension

A Helmholtz problem is governed by a differential equation similar to the heat conduction equation, e.g.,

$$\frac{\mathrm{d}^2\phi}{\mathrm{d}x^2} + c\phi + Q = 0 \tag{4.26}$$

with $c$ and $Q$ being dependent on position only. Boundary conditions can be of the form

$$\phi = \bar{\phi} \quad \text{on } \Gamma_\phi \quad \text{or} \quad q_n = -\frac{\mathrm{d}\phi}{\mathrm{d}n} \quad \text{on} \quad \Gamma_q \tag{4.27}$$

The above can be written in the general form of (4.20), with

$$\mathcal{L} = \left[\frac{\mathrm{d}^2}{\mathrm{d}x^2} + c\right], \quad b = Q \quad \text{and} \quad u = \phi \tag{4.28}$$

Verifying that self-adjointness applies (which we leave to the reader as an exercise), we immediately have a variational principle

$$\Pi = \int_\Omega \left[\frac{1}{2}\phi\left(\frac{\mathrm{d}^2\phi}{\mathrm{d}x^2} + c\phi\right) + \phi Q\right]\mathrm{d}x \tag{4.29}$$

with $\phi$ satisfying the forced boundary condition, i.e., $\phi = \bar{\phi}$ on $\Gamma_\phi$. Integrating by parts of the first term results in

$$\Pi = -\int_\Omega \left[\frac{1}{2}\left(\frac{\mathrm{d}\phi}{\mathrm{d}x}\right)^2 - \frac{1}{2}c\phi^2 - \phi Q\right]\mathrm{d}x - q_n\phi|_{\Gamma_q} \tag{4.30}$$

We note that the boundary term for prescribed $\phi$ does not alter the principle but is a constraint that must be satisfied by approximations to $\phi$.

In the next example we show that the above process applies to sets of differential equations as well as to single ones. In the case of sets of equations we generalize (4.10) to

$$\delta\Pi = \int_\Omega \delta\mathbf{u}^{\mathrm{T}}\boldsymbol{\mathcal{A}}(\mathbf{u})\mathrm{d}x + \delta\mathbf{u}^{\mathrm{T}}\boldsymbol{\mathcal{B}}(\mathbf{u})\Big|_\Gamma = 0 \tag{4.31}$$

**Example 4.4.** First-order form of heat equation

This problem concerns the one-dimensional heat conduction equation (Section 3.9.2), written in first-order form as

$$\boldsymbol{\mathcal{A}}(\mathbf{u}) = \begin{Bmatrix} q + \frac{\mathrm{d}\phi}{\mathrm{d}x} \\ -\frac{\mathrm{d}q}{\mathrm{d}x} + Q \end{Bmatrix} = \mathbf{0}$$

or using (4.20) generalized as

$$\boldsymbol{\mathcal{L}} \equiv \begin{bmatrix} 1 & \frac{\mathrm{d}}{\mathrm{d}x} \\ -\frac{\mathrm{d}}{\mathrm{d}x} & 0 \end{bmatrix}, \quad \mathbf{b} = \begin{Bmatrix} 0 \\ Q \end{Bmatrix} \quad \text{and} \quad \mathbf{u} = \begin{Bmatrix} q \\ \phi \end{Bmatrix}$$

Again self-adjointness of the operator can be tested and found to be satisfied. We now write a functional of *mixed* type as

$$\begin{aligned}\Pi &= \int_\Omega \left[ \tfrac{1}{2} \begin{Bmatrix} q \\ \phi \end{Bmatrix}^{\mathrm{T}} \left( \begin{bmatrix} 1 & \frac{\mathrm{d}}{\mathrm{d}x} \\ -\frac{\mathrm{d}}{\mathrm{d}x} & 0 \end{bmatrix} \begin{Bmatrix} q \\ \phi \end{Bmatrix} \right) + \begin{Bmatrix} q \\ \phi \end{Bmatrix}^{\mathrm{T}} \begin{Bmatrix} 0 \\ Q \end{Bmatrix} \right] \mathrm{d}x \\ &= \int_\Omega \left[ \tfrac{1}{2} \left( q^2 + q \tfrac{\mathrm{d}\phi}{\mathrm{d}x} - \phi \tfrac{\mathrm{d}q}{\mathrm{d}x} \right) + \phi Q \right] \mathrm{d}x \end{aligned} \tag{4.32}$$

The verification of the correctness of the above, by executing a variation, is left to the reader. This is called a *mixed variational form* since more than one variable type exists.

These two examples illustrate the simplicity of application of the general expressions. The reader will observe that self-adjointness of the operator will generally exist if even orders of differentiation are present. For odd orders self-adjointness is only possible if the operator is a "skew"-symmetric matrix such as occurs in the second example.

## 4.4 Maximum, minimum, or a saddle point?

In discussing variational principles so far we have assumed simply that at the solution point $\delta\Pi = 0$ and the functional is *stationary*. It is often desirable to know whether $\Pi$ is at a maximum, minimum, or simply at a "saddle point." If a maximum or a minimum is involved, then the approximation to $\Pi$ will always be "bounded," i.e., will provide approximate values of $\Pi$ which are either smaller or larger than the correct ones.[1] The bound in itself may be of practical significance in some problems.

When, in elementary calculus, we consider a stationary point of a function $\Pi$ of one variable $u$, we investigate the rate of change of $\mathrm{d}\Pi$ with $\mathrm{d}u$ and write

$$\mathrm{d}(\mathrm{d}\Pi) = \mathrm{d}\left( \frac{\partial \Pi}{\partial u} \mathrm{d}u \right) = \frac{\partial^2 \Pi}{\partial u^2} (\mathrm{d}u)^2 \tag{4.33}$$

The sign of the second derivative determines whether $\Pi$ is a minimum, maximum, or simply stationary (saddle point), as shown in Fig. 4.1. By analogy in the calculus of variations we shall consider changes of $\delta\Pi$. Noting the general form of this quantity given by (4.3) and the notion of the second derivative of (4.6) we can write, in terms of discrete parameters,

$$\delta(\delta\Pi) \equiv \delta \left( \frac{\partial \Pi}{\partial \tilde{\mathbf{u}}} \right)^{\mathrm{T}} \delta\tilde{\mathbf{u}} = \delta\tilde{\mathbf{u}}^{\mathrm{T}} \delta \left( \frac{\partial \Pi}{\partial \tilde{\mathbf{u}}} \right) = \delta\tilde{\mathbf{u}}^{\mathrm{T}} \left( \frac{\partial^2 \Pi}{\partial \tilde{\mathbf{u}} \partial \tilde{\mathbf{u}}} \delta\tilde{\mathbf{u}} \right) = \delta\tilde{\mathbf{u}}^{\mathrm{T}} \mathbf{K}_{\mathrm{T}} \delta\tilde{\mathbf{u}} \tag{4.34}$$

If, in the above, $\delta(\delta\Pi)$ is always negative then $\Pi$ is obviously reaching a maximum, if it is always positive then $\Pi$ is a minimum, but if the sign is indeterminate this shows only the existence of a saddle point.

[1] Provided all integrals are exactly evaluated.

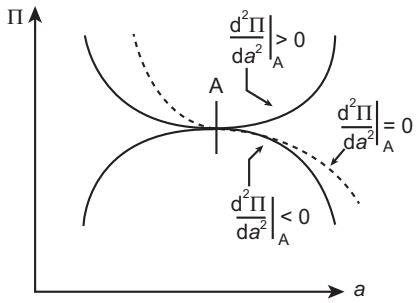



**FIGURE 4.1**
Maximum, minimum, and a "saddle" point for a functional Π of one variable.

As $\delta\tilde{\mathbf{u}}$ is an arbitrary vector this statement is equivalent to requiring the matrix $\mathbf{K}_T$ to be negative definite for a maximum *or* positive definite for a minimum. The form of the matrix $\mathbf{K}_T$ (or in linear problems of $\mathbf{K}$ which is identical to it) is thus of importance in the solution of variational problems. A matrix is positive definite if all its eigenvalues are positive. If the eigenvalues are positive or zero it is called positive semi-definite.

## 4.5 Constrained variational principles

### 4.5.1 Lagrange multipliers

Consider the problem of making a functional Π stationary, subject to the unknown $u$ obeying an additional relationship

$$C(u) = 0 \quad \text{in } \Omega \tag{4.35}$$

We can introduce this constraint by forming another functional

$$\bar{\Pi}(u, \lambda) = \Pi(u) + \int_\Omega \lambda C(u) \mathrm{d}x \tag{4.36}$$

in which $\lambda$ is some function of $x$ known as a *Lagrange multiplier*. The variation of the new functional is now

$$\delta\bar{\Pi} = \delta\Pi + \int_\Omega \lambda \delta C(u) \mathrm{d}x + \int_\Omega \delta\lambda C(u) \mathrm{d}x = 0 \tag{4.37}$$

which immediately gives $C(u) = 0$ and, simultaneously, an added contribution to the original $\delta\Pi$ involving $\lambda$. We note that the above process could also be introduced into a weak form directly by merely appending the last two terms.

In a similar way, constraints can be introduced at some points of the domain. For instance, if we require that $u$ obey

$$E(u) = 0 \quad \text{at } x = x_e \tag{4.38}$$

we would add to the original functional the term

$$\lambda E(u)|_{x=x_e} \tag{4.39}$$

with $\lambda$ now being an unknown parameter defined only at $x = x_e$.

It appears, therefore, possible to always introduce a number of additional functions $\lambda$ and modify a functional to include any prescribed constraint. In the "discretization" process we shall now have to use trial functions to describe both $u$ and $\lambda$. Writing, for instance,

$$\hat{u}(x,t) = \sum_a \mathrm{N}_a \tilde{u}_a = \mathbf{N}(x)\tilde{\mathbf{u}}(t); \quad \hat{\lambda}(x,t) = \sum_b \bar{N}_b \tilde{\lambda}_b = \bar{\mathbf{N}}(x)\tilde{\boldsymbol{\lambda}}(t) \tag{4.40}$$

we shall obtain a set of equations

$$\frac{\partial \Pi}{\partial \mathbf{w}} = \begin{Bmatrix} \dfrac{\partial \Pi}{\partial \tilde{\mathbf{u}}} \\ \dfrac{\partial \Pi}{\partial \tilde{\boldsymbol{\lambda}}} \end{Bmatrix} = \mathbf{0} \quad \text{where} \quad \mathbf{w} = \begin{Bmatrix} \tilde{\mathbf{u}} \\ \tilde{\boldsymbol{\lambda}} \end{Bmatrix} \tag{4.41}$$

from which both sets of parameters $\tilde{\mathbf{u}}$ and $\tilde{\boldsymbol{\lambda}}$ can be obtained. It is somewhat paradoxical that the "constrained" problem has resulted in a larger number of unknown parameters than the original one and, indeed, has complicated the solution. We shall, nevertheless, find practical use for Lagrange multipliers in formulating some physical variational principles, and will make use of these in a more general context in Chapter 9.

Before proceeding further it is of interest to investigate the form of equations resulting from the modified functional $\Pi$ of (4.36). If the original functional $\Pi$ gave as its Euler equations a system

$$\boldsymbol{\mathcal{A}}(\mathbf{u}) = \mathbf{0} \tag{4.42}$$

then we have (omitting the boundary terms) for a system of constraints

$$\delta \bar{\Pi} = \int_\Omega \delta \mathbf{u}^{\mathrm{T}} \boldsymbol{\mathcal{A}}(\mathbf{u}) \mathrm{d}x + \int_\Omega \delta \mathbf{C}^{\mathrm{T}} \boldsymbol{\lambda} \mathrm{d}x + \int_\Omega \delta \boldsymbol{\lambda}^{\mathrm{T}} \mathbf{C}(\mathbf{u}) \mathrm{d}x = 0 \tag{4.43}$$

Substituting the trial functions (4.40) we can write for a linear set of constraints

$$\mathbf{C}(\mathbf{u}) = \boldsymbol{\mathcal{L}}_1 \mathbf{u} + \mathbf{C}_1$$

that

$$\begin{aligned} \delta \bar{\Pi} = \delta \tilde{\mathbf{u}}^{\mathrm{T}} \left[ \int_\Omega \mathbf{N}^{\mathrm{T}} \boldsymbol{\mathcal{A}}(\hat{\mathbf{u}}) \mathrm{d}x + \int_\Omega (\boldsymbol{\mathcal{L}}_1 \mathbf{N})^{\mathrm{T}} \hat{\boldsymbol{\lambda}} \, \mathrm{d}x \right] \\ + \delta \tilde{\boldsymbol{\lambda}}^{\mathrm{T}} \int_\Omega \bar{\mathbf{N}}^{\mathrm{T}} (\boldsymbol{\mathcal{L}}_1 \hat{\mathbf{u}} + \mathbf{C}_1) \mathrm{d}x = 0 \end{aligned} \tag{4.44}$$

As this has to be true for all variations $\delta\tilde{\mathbf{u}}$ and $\delta\tilde{\boldsymbol{\lambda}}$, we have a system of equations

$$\begin{aligned}\int_\Omega \mathbf{N}^{\mathrm{T}}\boldsymbol{\mathcal{A}}(\hat{\mathbf{u}})\mathrm{d}x + \int_\Omega (\boldsymbol{\mathcal{L}}_1\mathbf{N})^{\mathrm{T}}\hat{\boldsymbol{\lambda}}\,\mathrm{d}x = \mathbf{0}\\ \int_\Omega \bar{\mathbf{N}}^{\mathrm{T}}(\boldsymbol{\mathcal{L}}_1\hat{\mathbf{u}} + \mathbf{C}_1)\mathrm{d}x = \mathbf{0}\end{aligned} \tag{4.45}$$

For linear equations $\boldsymbol{\mathcal{A}}$, the first term of the first equation is precisely the ordinary, unconstrained, variational approximation

$$\mathbf{K}_{uu}\tilde{\mathbf{u}} + \mathbf{f}_u \tag{4.46}$$

and inserting again the trial functions (4.40) we can write the approximated (4.45) as a linear system:

$$\mathbf{K}_w\mathbf{w} = \begin{bmatrix}\mathbf{K}_{uu} & \mathbf{K}_{u\lambda}\\ \mathbf{K}_{u\lambda}^{\mathrm{T}} & \mathbf{0}\end{bmatrix}\begin{Bmatrix}\tilde{\mathbf{u}}\\ \tilde{\boldsymbol{\lambda}}\end{Bmatrix} + \begin{Bmatrix}\mathbf{f}_u\\ \mathbf{f}_\lambda\end{Bmatrix} = \mathbf{0} \tag{4.47}$$

with

$$\mathbf{K}_{u\lambda}^{\mathrm{T}} = \int_\Omega \bar{\mathbf{N}}^{\mathrm{T}}(\boldsymbol{\mathcal{L}}_1\mathbf{N})\mathrm{d}x, \quad \mathbf{f}_\lambda = \int_\Omega \bar{\mathbf{N}}^{\mathrm{T}}\mathbf{C}_1\mathrm{d}x \tag{4.48}$$

Clearly the system of equations is symmetric but now possesses zeros on the diagonal, and therefore the variational principle $\Pi$ is merely stationary. Further, computational difficulties may be encountered unless the solution process allows for zero diagonal terms.

### 4.5.2 Identification of Lagrange multipliers: Forced boundary conditions and modified variational principles

Although the Lagrange multipliers were introduced as a mathematical concept necessary for the enforcement of certain external constraints required to satisfy the original variational principle, we shall find that in many situations they can be identified with certain physical quantities of importance to the original mathematical model. Such an identification will follow immediately from the definition of the variational principle established in (4.36) and through the first of the Euler equations in (4.45) corresponding to it. The variation $\delta\bar{\Pi}$, written in (4.37), supplies through its third term the constraint equation. The first two terms can always be rewritten as

$$\int_\Omega \delta\mathbf{C}(\mathbf{u})^{\mathrm{T}}\boldsymbol{\lambda}\,\mathrm{d}x + \int_\Omega \delta\mathbf{u}^{\mathrm{T}}\boldsymbol{\mathcal{A}}(\mathbf{u})\,\mathrm{d}x = \mathbf{0} \tag{4.49a}$$

and/or

$$\int_\Gamma \delta\mathbf{E}(\mathbf{u})^{\mathrm{T}}\boldsymbol{\lambda}\,\mathrm{d}\Gamma + \int_\Gamma \delta\mathbf{u}^{\mathrm{T}}\boldsymbol{\mathcal{B}}(\mathbf{u})\,\mathrm{d}\Gamma = \mathbf{0} \tag{4.49b}$$

This supplies the identification of $\boldsymbol{\lambda}$.

In the literature of variational calculation such identification arises frequently and the reader is referred to the book by Washizu [10] for numerous examples.

**Example 4.5.** Identification of Lagrange multiplier for boundary condition

Here we shall introduce this identification by means of the example considered in Section 4.2.1. As we have noted, the variational principle of (4.12) established the governing equation and the natural boundary conditions of the heat conduction problem providing the essential boundary condition

$$E(\phi) = \phi - \bar{\phi} = 0 \tag{4.50}$$

was satisfied on $\Gamma_\phi$ in the choice of the trial function for $\phi$.

The above forced boundary condition can, however, be considered as a constraint on the original problem. We can write the constrained variational principle as

$$\bar{\Pi} = \Pi + \lambda(\phi - \bar{\phi})\big|_{\Gamma_\phi} \tag{4.51}$$

where $\Pi$ is given by (4.12).

Performing the variation we have

$$\delta\bar{\Pi} = \delta\Pi + \delta\phi\lambda|_{\Gamma_\phi} + \delta\lambda(\phi - \bar{\phi})\big|_{\Gamma_\phi} = 0 \tag{4.52}$$

$\delta\Pi$ is now given by expression (4.14a) augmented by a term

$$\delta\phi k \left.\frac{\mathrm{d}\phi}{\mathrm{d}n}\right|_{\Gamma_\phi} \tag{4.53}$$

which was previously disregarded (as we had assumed that $\delta\phi = 0$ on $\Gamma_\phi$). In addition to the conditions of (4.14b), we now require that

$$\delta\lambda(\phi - \bar{\phi})\big|_{\Gamma_\phi} + \delta\phi\left(\lambda + k\frac{\mathrm{d}\phi}{\mathrm{d}n}\right)\Bigg|_{\Gamma_\phi} = 0 \tag{4.54}$$

which must be true for all variations $\delta\lambda$ and $\delta\phi$. The first simply reiterates the constraint

$$\phi - \bar{\phi} = 0 \quad \text{on } \Gamma_\phi \tag{4.55}$$

The second *defines* $\lambda$ as

$$\lambda = -k\frac{\mathrm{d}\phi}{\mathrm{d}n} \tag{4.56}$$

Noting that $k(\partial\phi/\partial n)$ is the negative to the flux $q_n$ on the boundary $\Gamma_\phi$, the physical identification of the multiplier has been achieved—that is, $\lambda \equiv q_n$.

## 4.6 Constrained variational principles: Penalty function and perturbed Lagrangian methods

In the previous section we have seen how the process of introducing Lagrange multipliers allows constrained variational principles to be obtained at the expense of increasing the total number of unknowns. Further, we have shown that even in linear problems the algebraic equations which have to be solved are now complicated by having zero diagonal terms. In this section we shall consider alternative procedures of introducing constraints which do not possess these drawbacks.

### 4.6.1 Penalty functions

Considering once again the problem of obtaining stationarity of $\Pi$ with a set of constraint equations $\mathbf{C}(\mathbf{u}) = \mathbf{0}$ in domain $\Omega$, we note that the product

$$\mathbf{C}^{\mathrm{T}}\mathbf{C} = C_1^2 + C_2^2 + \cdots \tag{4.57}$$

where $\mathbf{C}^{\mathrm{T}} = [C_1, C_2, \ldots]$ must always be a quantity which is positive or zero. Clearly, the latter value is found when the constraints are satisfied and clearly the variation

$$\delta(\mathbf{C}^{\mathrm{T}}\mathbf{C}) = 0 \tag{4.58}$$

as the product reaches that minimum.

We can now write a new functional

$$\bar{\bar{\Pi}} = \Pi + \frac{1}{2}\alpha \int_\Omega \mathbf{C}^{\mathrm{T}}(\mathbf{u})\mathbf{C}(\mathbf{u})\mathrm{d}x \tag{4.59}$$

in which $\alpha$ is a "penalty number" and then require stationarity for the constrained solution. If $\Pi$ is itself a minimum of the solution then $\alpha$ should be a positive number. The solution obtained by the stationarity of the functional $\bar{\bar{\Pi}}$ will satisfy the constraints only approximately. The larger the value of $\alpha$ the better will be the constraints achieved. Further, it seems obvious that the process is best suited to cases where $\Pi$ is a minimum (or maximum) principle, but success can be obtained even with purely saddle point problems. The process is equally applicable to constraints applied on boundaries or simple discrete constraints. In this latter case integration is dropped.

**Example 4.6.** Linear constraint

Consider the linear constraint

$$C(u) = C_0 u + C_1$$

with the approximation for $u$ given by (4.40). The added term to the variational theorem is given by

$$\int_\Omega [C(u)]^2 \mathrm{d}x = \tilde{\mathbf{u}}^{\mathrm{T}} \left[ \int_\Omega \mathbf{N}^{\mathrm{T}} C_0^2 \mathbf{N}\, \mathrm{d}x\, \tilde{\mathbf{u}} + \int_\Omega \mathbf{N}^{\mathrm{T}} 2C_0 C_1\, \mathrm{d}x \right] + \int_\Omega C_1^2\, \mathrm{d}x$$

Upon variation the last term vanishes.

### 4.6.2 Perturbed Lagrangian

As an alternative to a penalty method we consider once again the problem of obtaining stationarity of $\Pi$ with a constraint equation $C(u) = 0$ in domain $\Omega$. The Lagrange multiplier form to embed the constraint is given in (4.36). Here we modify the expression by appending a quadratic term of the form $\lambda^2$ scaled by a parameter $\alpha$. The form of the final equation is given by

$$\breve{\Pi}(u, \lambda) = \Pi(u) + \int_\Omega \lambda C(u)\mathrm{d}x - \frac{1}{2\alpha}\int_\Omega \lambda^2 \mathrm{d}x \tag{4.60}$$

We note that as the parameter $\alpha$ tends toward infinity the form approaches a Lagrange multiplier form. Accordingly, this form is called a *perturbed Lagrangian* functional. Taking the variation we obtain the result

$$\delta\breve{\Pi} = \delta\Pi + \int_\Omega \lambda\delta C(u)\mathrm{d}x + \int_\Omega \delta\lambda C(u)\mathrm{d}x - \frac{1}{\alpha}\int_\Omega \delta\lambda\lambda\,\mathrm{d}x = 0 \tag{4.61}$$

If the constraints are a linear form given by

$$C(u) = C_0\,u + C_1 = 0$$

we can introduce the approximations (4.40) into (4.61) along with the constant $\lambda = \tilde{\lambda}$ to obtain the set of equations

$$\begin{bmatrix} \mathbf{K}_{uu} & \mathbf{K}_{u\lambda} \\ \mathbf{K}_{\lambda u} & -\frac{1}{\alpha}K_{\lambda\lambda} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}} \\ \tilde{\lambda} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ f_\lambda \end{Bmatrix} \tag{4.62}$$

where $\mathbf{K}_{uu}$ is the coefficient array from $\delta\Pi$ and

$$\mathbf{K}_{u\lambda} = \int_\Omega \mathbf{N}^{\mathrm{T}} C_0 \mathrm{d}x = \mathbf{K}_{\lambda u}^{\mathrm{T}}, \quad K_{\lambda\lambda} = \int_\Omega \mathrm{d}x \quad \text{and} \quad f_\lambda = -\int_\Omega C_1\,\mathrm{d}x$$

The second equation of (4.62) may be solved for $\tilde{\lambda}$ in terms of $\tilde{\mathbf{u}}$ and substituted into the first equation to obtain

$$\bar{\mathbf{K}}_{uu}\tilde{\mathbf{u}} = \left[\mathbf{K}_{uu} + \alpha\mathbf{K}_{u\lambda}K_{\lambda\lambda}^{-1}\mathbf{K}_{\lambda u}\right]\tilde{\mathbf{u}} = \mathbf{f} + \alpha\mathbf{K}_{u\lambda}K_{\lambda\lambda}^{-1}f_\lambda$$

It is now apparent that the perturbed Lagrangian and penalty forms are closely related. The perturbed Lagrangian uses

$$\mathbf{K}_{u\lambda}K_{\lambda\lambda}^{-1}\mathbf{K}_{\lambda u}$$

to impose the constraint whereas the penalty method uses

$$\int_\Omega \mathbf{N}^{\mathrm{T}} C_0^2 \mathbf{N}\,\mathrm{d}x$$

When the constraint is a simple scalar relation at a single point the two methods are identical. However, when general forms $\mathbf{C}(\mathbf{u})$ are considered the methods will yield different approximations unless the shape functions for the set of $\boldsymbol{\lambda}$ include all the terms contained in $\delta\mathbf{C}(\mathbf{u})$.

In practical applications the method of penalty functions has proved to be quite effective [11], and indeed is often introduced intuitively.

**Example 4.7.** Boundary condition by penalty method

In the example presented next the forced boundary conditions are not introduced *a priori* and the problem gives, on assembly, a singular system of equations

$$\mathbf{K}\tilde{\mathbf{u}} - \mathbf{f} = \mathbf{0} \tag{4.63}$$

which can be obtained from the functional (providing $\mathbf{K}$ is symmetric)

$$\Pi = \frac{1}{2}\tilde{\mathbf{u}}^{\mathrm{T}}\mathbf{K}\tilde{\mathbf{u}} - \tilde{\mathbf{u}}^{\mathrm{T}}\mathbf{f} \tag{4.64}$$

Introducing a prescribed value of $u_1$, i.e., writing

$$C(u_1) = u_1 - \bar{u}_1 = 0 \tag{4.65}$$

the functional can be modified to

$$\bar{\bar{\Pi}} = \Pi + \frac{1}{2}\alpha(u_1 - \bar{u}_1)^2 \tag{4.66}$$

yielding

$$\bar{\bar{K}}_{11} = K_{11} + \alpha; \quad \bar{\bar{f}}_1 = f_1 - \alpha\bar{u}_1 \tag{4.67}$$

and giving no change in any of the other matrix coefficients.

The use of the penalty function in the finite element context presents certain difficulties.

*Firstly*, the constrained functional of (4.59) leads to equations of the form

$$(\mathbf{K}_1 + \alpha\mathbf{K}_2)\tilde{\mathbf{u}} - \bar{\bar{\mathbf{f}}} = \mathbf{0} \tag{4.68}$$

where $\mathbf{K}_1$ derives from the original functional and $\mathbf{K}_2$ from the constraints. As $\alpha$ increases $\mathbf{K}_1$ becomes insignificant compared to $\alpha\mathbf{K}_2$ and the above equation degenerates to

$$\mathbf{K}_2\tilde{\mathbf{u}} = \mathbf{f}/\alpha \to \mathbf{0}$$

and $\tilde{\mathbf{u}} = \mathbf{0}$ unless the matrix $\mathbf{K}_2$ is singular. The phenomenon where $\tilde{\mathbf{u}} \Rightarrow \mathbf{0}$ is known as *locking* and has often been encountered by researchers who failed to recognize its source. This singularity in the equations does not always arise and we shall discuss means of its introduction in Chapters 9 and 10.

*Secondly*, with large but finite values of $\alpha$ numerical difficulties will be encountered. Noting that discretization errors can be of comparable magnitude to those due to not *satisfying* the constraint, we can make

$$\alpha = \text{constant}(1/h)^n$$

(where $h$ is an element size parameter) ensuring a limiting convergence to the correct answer. Fried [12,13] discusses this problem in detail.

A more general discussion of the whole topic is given in Ref. [14] and in Chapter 10 where the relationship between Lagrange constraints and penalty forms is made clear.

## 4.7 Least squares approximations

A general variational principle also may be constructed if the constraints described in the previous section are simply the governing equations of the problem

$$\mathbf{C}(\mathbf{u}) = \boldsymbol{\mathcal{A}}(\mathbf{u}) \tag{4.69}$$

Obviously the same procedure can be used in the context of the penalty function approach by setting $\Pi = 0$ in (4.59). We can thus write a "variational principle"

$$\bar{\bar{\Pi}} = \frac{1}{2}\int_\Omega \left(A_1^2 + A_2^2 + \cdots\right) \mathrm{d}x = \frac{1}{2}\int_\Omega \boldsymbol{\mathcal{A}}^{\mathrm{T}}(\mathbf{u})\boldsymbol{\mathcal{A}}(\mathbf{u})\mathrm{d}x \tag{4.70}$$

for any set of differential equations. In the above equation the boundary conditions are assumed to be satisfied by $\mathbf{u}$ (forced boundary condition) and the parameter $\alpha$ is dropped as it becomes a multiplier.

Clearly, the above statement is a requirement that the sum of the squares of the residuals of the differential equations should be a minimum at the correct solution. This minimum is obviously zero at that point, and the process is simply the well-known *least squares method* of approximation.

It is equally obvious that we could obtain the correct solution by minimizing any functional of the form

$$\bar{\bar{\Pi}} = \frac{1}{2}\int_\Omega \left(p_1 A_1^2 + p_2 A_2^2 + \cdots\right) \mathrm{d}x = \frac{1}{2}\int_\Omega \boldsymbol{\mathcal{A}}^{\mathrm{T}}(\mathbf{u})\mathbf{p}\boldsymbol{\mathcal{A}}(\mathbf{u})\mathrm{d}x \tag{4.71}$$

in which $p_1$, $p_2$, . . ., etc., are positive valued weighting functions or constants and $\mathbf{p}$ is a diagonal matrix:

$$\mathbf{p} = \begin{bmatrix} p_1 & & & 0 \\ & p_2 & & \\ & & p_3 & \\ 0 & & & \ddots \end{bmatrix} \tag{4.72}$$

The above alternative form is sometimes convenient as it places a different importance on the satisfaction of individual components of the equation set and allows additional freedom in the choice of the approximate solution. Once again this weighting function could be chosen so as to ensure a constant ratio of terms contributed by various equations.

A least squares method of the kind shown above is a very powerful alternative procedure for obtaining integral forms from which an approximate solution can be started, and has been used with considerable success [15–18]. As a least squares variational principle can be written for *any* set of differential equations without introducing additional variables, we may well inquire as to what the difference is between these and the *natural variational principles* discussed previously. On performing a variation in a specific case the reader will find that the Euler equations which are obtained

no longer give the original differential equations but give higher order derivatives of these. Thus, higher order continuity of trial functions is now generally needed. This may be a serious drawback but frequently can be bypassed by stating the original problem as a set of lower order equations. The appearance of higher order derivatives in the Euler equations also introduces the possibility of spurious solutions if incorrect boundary conditions are used.

**Example 4.8.** Least squares solution for Helmholtz equation

To illustrate the use of a least squares approach consider the Helmholtz problem governed by (4.26), for which we have already obtained a *natural* variational principle [Eq. (4.30)] in which only first derivatives were involved, requiring $C_0$ continuity for $\phi$. Now, if we use the operator $\mathcal{L}$ and term $b$ defined by (4.28), we have a set of approximating equations with

$$\begin{aligned} K_{ab} &= \int_\Omega \left(\frac{\mathrm{d}^2 N_a}{\mathrm{d}x^2} + cN_a\right)\left(\frac{\mathrm{d}^2 N_b}{\mathrm{d}x^2} + cN_b\right)\mathrm{d}x \\ f_a &= \int_\Omega \left(\frac{\mathrm{d}^2 N_a}{\mathrm{d}x^2} + cN_a\right) Q\,\mathrm{d}x \end{aligned} \tag{4.73}$$

The reader will observe that due to the presence of second derivatives $C_1$ continuity is now needed for the trial functions $\mathbf{N}$.

An alternative, avoiding the requirement of $C_1$ functions, is to write (4.26) as a first-order system. This can be written as

$$\mathcal{A}(\mathbf{u}) = \left\{ \begin{array}{c} q + \dfrac{\mathrm{d}\phi}{\mathrm{d}x} \\ -\dfrac{\mathrm{d}q}{\mathrm{d}x} + c\phi + Q \end{array} \right\} = 0 \tag{4.74}$$

or, introducing the vector $\mathbf{u}$,

$$\mathbf{u} = \left\{ \begin{array}{c} q \\ \phi \end{array} \right\} = (\mathbf{N}\tilde{\mathbf{u}}) \tag{4.75}$$

as the unknown we can write an approximation as

$$\mathbf{u} \approx \hat{\mathbf{u}} = \begin{bmatrix} \mathbf{N}_q & \mathbf{0} \\ \mathbf{0} & \mathbf{N}_\phi \end{bmatrix} \left\{ \begin{array}{c} \tilde{\mathbf{q}} \\ \tilde{\boldsymbol{\phi}} \end{array} \right\} = \mathbf{N}\tilde{\mathbf{u}} \tag{4.76}$$

where $\mathbf{N}_q$ and $\mathbf{N}_\phi$ are $C_0$ shape functions for the $q$ and $\phi$ variables, respectively. The least squares approximation is now given by

$$\delta \bar{\bar{\Pi}} = \delta \tilde{\mathbf{u}}^{\mathrm{T}} \int_\Omega (\mathcal{L}\mathbf{N})^{\mathrm{T}} [(\mathcal{L}\mathbf{N})\tilde{\mathbf{u}} + \mathbf{b}] \mathrm{d}x = 0 \tag{4.77a}$$

where

$$\mathcal{L}\mathbf{N} = \begin{bmatrix} \mathbf{N}_q & \dfrac{\mathrm{d}\mathbf{N}_\phi}{\mathrm{d}x} \\ -\dfrac{\mathrm{d}\mathbf{N}_q}{\mathrm{d}x} & c\mathbf{N}_\phi \end{bmatrix}; \quad \mathbf{b} = \left\{ \begin{array}{c} 0 \\ Q \end{array} \right\} \tag{4.77b}$$

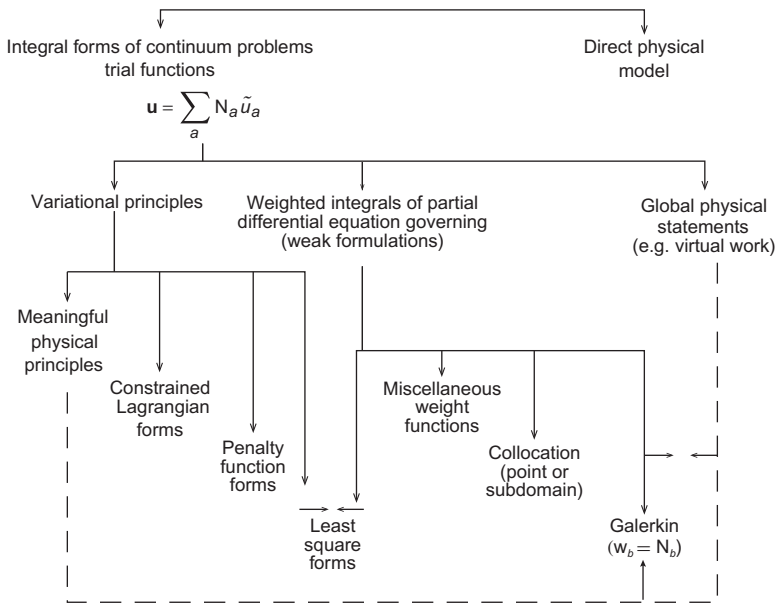
The reader can now perform the final steps to obtain the **K** and **f** matrices. The approximation equations in a form requiring only $C_0$ continuity are obtained, however, at the expense of additional variables. Use of such forms has been made extensively in the finite element context [15–22].

## 4.8 Concluding remarks: Finite difference and boundary methods

This chapter along with Chapter 3 present the general possibilities of using the finite element process in almost any mathematical or mathematically modeled physical problem. The essential approximation processes mostly have been given in a simple one-dimensional form. In the chapters that follow we shall apply many of these methods to multidimensional problems in mechanics and other fields. In some we shall show, however, that certain extensions of the process are possible. For example, in Chapter 8 we show how a violation of some of the rules here expounded can be accepted.

The numerous approximation procedures discussed fall into several categories. To remind the reader of these, we present in Table 4.1 a comprehensive catalog of the methods used here and in Chapter 3. The only aspect of the finite element process mentioned in this table that has not been discussed here is that of a *direct physical method*. In such models an "atomic" rather than continuum concept is the starting

**Table 4.1** Finite Element Approximation



point. While much interest exists in the possibilities offered by such models, their discussion is outside the scope of this book.

In all the continuum processes discussed the first step is always the choice of suitable shape or trial functions. A few simple one-dimensional forms of such functions have been introduced as needs demanded and many new forms will be introduced in the next two chapters. Indeed, the reader who has mastered the essence of the last two chapters will have little difficulty in applying the finite element method to any suitably defined physical problem once appropriate shape functions are available.

The methods listed do not include specifically two well-known techniques, i.e., *finite difference* methods and *boundary solution* methods (sometimes known as boundary elements). In the general sense these belong to the category of a *generalized finite element* method [23]. Boundary solution methods choose the trial functions such that the governing equation is automatically satisfied in the domain $\Omega$. In this class of problems only boundary terms remain to be satisfied. Finite difference procedures can be interpreted as an approximation based on local, discontinuous, shape functions with collocation weighting applied (although usually the derivation of the approximation algorithm is based on a Taylor expansion).

Many textbooks deal exclusively with these types of approximations. References [24–27] discuss finite difference approximation and Refs. [28–31] relate to boundary methods.

## 4.9 Problems

**4.1** Deduce the Euler differential equation and boundary conditions for the variational principle expressed as

$$\Pi(u) = \int_a^b \left[ EI \left( \frac{\mathrm{d}u}{\mathrm{d}x} \right)^2 - Pu^2 \right] \mathrm{d}x - ug \Big|_{x=b}, \quad u(a) = 0$$

Classify $\Pi$ as a minimum, maximum, or saddle point form.

**4.2** Deduce the Euler differential equation and boundary conditions for the variational principle expressed as

$$\Pi(u) = \int_a^b \left[ EA \left( \frac{\mathrm{d}u}{\mathrm{d}x} \right)^2 + ku^2 - 2qu \right] \mathrm{d}x + \alpha[(u(a))^2 + (u(b))^2]$$

where $EA$ and $k$ are constant parameters and $\alpha$ is a penalty parameter.

**4.3** Deduce the Euler equations and boundary conditions for the variational principle expressed as

$$\Pi(u, \lambda_a, \lambda_b) = \int_a^b \left[ EA \left( \frac{\mathrm{d}u}{\mathrm{d}x} \right)^2 + ku^2 - 2qu \right] \mathrm{d}x + \lambda_a u(a) + \lambda_b u(b)$$

where $EA$, $k$, and $q$ are constant parameters and $\alpha$ is a penalty parameter.

**4.4** Construct a variational theorem which gives the weak form obtained in Problems 3.2 and 3.3 as the first variation.

**4.5** Construct a variational theorem which has the irreducible differential equation for the beam problem described by Problem 3.4 as its first variation.

**4.6** Solve the one-dimensional heat equation given in Example 3.11 by enforcing the boundary conditions by the penalty formulation described by Example 4.7. How large must each penalty parameter be taken to make the boundary error less than $10^{-6}|\phi_{\max}|$?

## References

[1] F.B. Hildebrand, Methods of Applied Mathematics, second ed., Prentice-Hall, 1965 (reprinted by Dover Publishers, 1992).

[2] Lord Rayleigh (J.W. Strutt), On the theory of resonance, Trans. Roy. Soc. (Lond.) A161 (1870) 77–118.

[3] W. Ritz, Über eine neue Methode zur Lösung gewisser variationsproblem der mathematischen physik, J. für die reine und angewandte Mathematik 135 (1908) 1–61.

[4] S.C. Mikhlin, Variational Methods in Mathematical Physics, Macmillan, New York, 1964.

[5] S.C. Mikhlin, The Problem of the Minimum of a Quadratic Functional, Holden-Day, San Francisco, 1966.

[6] E. Tonti, Variational formulation of non-linear differential equations, Bull. Acad. Roy. Belg. Classe Sci, 55 (1969) 137–165, 263–278.

[7] M.M. Vainberg, Variational Methods for the Study of Nonlinear Operators, Holden-Day Inc., San Francisco, CA, 1964.

[8] J.T. Oden, A general theory of finite elements. Part I. Topological considerations, Int. J. Numer. Methods Eng. 1 (1969) 205–246.

[9] J.T. Oden, A general theory of finite elements. Part II. Applications, Int. J. Numer. Methods Eng. 1 (1969) 247–254.

[10] K. Washizu, Variational Methods in Elasticity and Plasticity, third ed., Pergamon Press, New York, 1982.

[11] O.C. Zienkiewicz, Constrained variational principles and penalty function methods in finite element analysis, in: Lecture Notes in Mathematics, No. 363, Springer-Verlag, Berlin, 1974, pp. 207–214.

[12] I. Fried, Shear in $c^0$ and $c^1$ bending finite elements, Int. J. Solids Struct. 9 (1973) 449–460.

[13] I. Fried, Finite element analysis of incompressible materials by residual energy balancing, Int. J. Solids Struct. 10 (1974) 993–1002.

[14] O.C. Zienkiewicz, E. Hinton, Reduced integration, function smoothing and non-conformity in finite element analysis, J. Franklin Inst. 302 (1976) 443–461.

[15] P.P. Lynn, S.K. Arya, Finite elements formulation by the weighted discrete least squares method, Int. J. Numer. Methods Eng. 8 (1974) 71–90.

[16] O.C. Zienkiewicz, D.R.J. Owen, K.N. Lee, Least square finite element for elasto-static problems—use of reduced integration, Int. J. Numer. Methods Eng. 8 (1974) 341–358.
[17] B.-N. Jiang, Least Squares Finite Element Method: Theory and Applications in Computational Fluid Dynamics and Electromagnetics, Springer, New York, 1998.
[18] P.B. Bochev, M.D. Gunzburger, Least-Squares Finite Element Methods, Springer, New York, 2009.
[19] B.-N. Jiang, Optimal least-squares finite element method for elliptic problems, Comput. Methods Appl. Mechanics Eng. 102 (1993) 199–212.
[20] B.-N. Jiang, On the least-squares method, Computer Methods in Appl. Mech. Eng. 152 (1998) 239–257.
[21] B.-N. Jiang, The least-squares finite element method in elasticity. I. Plane stress or strain with drilling degrees of freedom, Int. J. Numer. Methods Eng. 53 (2002) 621–636.
[22] B.-N. Jiang, The least-squares finite element method in elasticity. II. Bending of thin plates, Int. J. Numer. Methods Eng. 54 (2002) 1459–1475.
[23] O.C. Zienkiewicz, K. Morgan, Finite Elements and Approximation, John Wiley & Sons, London, 1983.
[24] R.V. Southwell, Relaxation Methods in Theoretical Physics, Clarendon Press, Oxford, 1946.
[25] D.N. de G. Allen, Relaxation Methods, McGraw-Hill, London, 1955.
[26] F.B. Hildebrand, Introduction to Numerical Analysis, second ed., Dover Publishers, 1987.
[27] A.R. Mitchell, D. Griffiths, The Finite Difference Method in Partial Differential Equations, John Wiley & Sons, London, 1980.
[28] P.K. Banerjee, The Boundary Element Methods in Engineering, McGraw-Hill, London, 1994.
[29] Prem K. Kythe, An Introduction to Boundary Element Methods, CRC Press, 1995.
[30] G. Beer, J.O. Watson, Programming the Boundary Element Method: An Introduction for Engineers, John Wiley & Sons, Chichester, 2001.
[31] L. Gaul, Boundary Element Methods for Engineers and Scientists, Springer, Berlin, 2003.

This page is intentionally left blank

CHAPTER

# Field Problems: A Multidimensional Finite Element Method

5

## 5.1 Field problems: Quasi-harmonic equation

The general procedures discussed in the previous two chapters can be applied to a variety of physical problems. Here we shall deal with situations governed by the general "quasi-harmonic" equation described in Chapter 2 and solved by finite element methods in Chapters 3 and 4 for the one-dimensional case. Particular cases of the equation are the well-known Laplace, Poisson, and Helmholtz equations [1–6]. The range of physical problems falling into this category is large. To list but a few frequently encountered in engineering practice we have:

- Heat conduction
- Seepage through porous media
- Irrotational flow of ideal fluids
- Distribution of electrical (or magnetic) potential
- Torsion of prismatic shafts
- Lubrication of pad bearings, etc.

The formulation developed in this chapter is equally applicable to all, and hence only limited reference will be made to the actual physical quantities. In all the above classes of problems, the behavior can be represented in terms of a scalar variable for which we will generally use the symbol $\phi$ as the physical variable describing the behavior. However, in discussing heat conduction applications $\phi$ becomes the temperature which we will denote by the symbol $T$. Similar changes will be made when discussing other specific physical problems.

In general Cartesian coordinates we recall that the balance equation for the flux $\mathbf{q}$ is given by [viz. (2.78)]

$$-\left(\frac{\partial q_x}{\partial x}+\frac{\partial q_y}{\partial y}+\frac{\partial q_z}{\partial z}\right)+Q=c\frac{\partial \phi}{\partial t} \tag{5.1}$$

Following the steps described in (3.1) a weak form for the equation may be written as

$$\mathrm{G}(\delta\phi,\phi,\mathbf{q})=\int_\Omega \delta\phi\left[c\frac{\partial \phi}{\partial t}+\left(\frac{\partial q_x}{\partial x}+\frac{\partial q_y}{\partial y}+\frac{\partial q_z}{\partial z}\right)-Q\right]\mathrm{d}\Omega=0 \tag{5.2}$$

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00005-8

After integration by parts the weak form becomes

$$\begin{aligned} \mathrm{G}(\delta\phi, \phi, \mathbf{q}) = &\int_\Omega \delta\phi \left[ c\frac{\partial \phi}{\partial t} - Q \right] \mathrm{d}\Omega + \int_{\Gamma_q} \delta\phi\, \bar{q}_n \,\mathrm{d}\Gamma \\ &- \int_\Omega \begin{bmatrix} \dfrac{\partial \delta\phi}{\partial x} & \dfrac{\partial \delta\phi}{\partial y} & \dfrac{\partial \delta\phi}{\partial z} \end{bmatrix} \begin{Bmatrix} q_x \\ q_y \\ q_z \end{Bmatrix} \mathrm{d}\Omega = 0 \end{aligned} \tag{5.3}$$

where $q_n = \mathbf{n}^{\mathrm{T}}\mathbf{q}$ is the flux normal to the boundary with $\mathbf{n}$ the outward pointing boundary normal. In the above we have also split the boundary term as

$$\int_\Gamma \delta\phi\, q_n \,\mathrm{d}\Gamma = \int_{\Gamma_\phi} \delta\phi\, q_n \,\mathrm{d}\Gamma + \int_{\Gamma_q} \delta\phi\, \bar{q}_n \,\mathrm{d}\Gamma \tag{5.4}$$

and assumed $\phi = \bar{\phi}$ is satisfied such that we may set $\delta\phi = 0$ on $\Gamma_\phi$. We also recall from Section 2.3.2 that $\bar{q}_n$ can include boundary radiation effects as

$$\bar{q}_n = \bar{q} + H(\phi - \phi_0)$$

which we will use in some of the subsequent developments.

### 5.1.1 Irreducible form

To obtain an irreducible form we introduce a linear relationship to relate the flux to the gradient of $\phi$. The form was given in (2.83) as

$$\mathbf{q} = -\mathbf{k}\,\nabla\phi \tag{5.5}$$

where $\mathbf{k}$ is a symmetric matrix of coefficients and

$$\nabla\phi = \begin{Bmatrix} \dfrac{\partial \phi}{\partial x} \\ \dfrac{\partial \phi}{\partial y} \\ \dfrac{\partial \phi}{\partial z} \end{Bmatrix} \tag{5.6}$$

denotes the *gradient* of $\phi$. We also recall that the relationship is known as Fourier's, Fick's, or Darcy's law depending on which physical problem we consider.

Introducing (5.5) into (5.3) for a general three-dimensional problem yields the desired irreducible form

$$\begin{aligned} \mathrm{G}(\delta\phi, \phi) = &\int_\Omega \delta\phi \left[ c\frac{\partial \phi}{\partial t} - Q \right] \mathrm{d}\Omega + \int_\Omega (\nabla\delta\phi)^{\mathrm{T}}\mathbf{k}\,\nabla\phi \,\mathrm{d}\Omega \\ &+ \int_{\Gamma_q} \delta\phi \left[ \bar{q} + H(\phi - \phi_0) \right] \mathrm{d}\Gamma = 0 \end{aligned} \tag{5.7}$$

### 5.1.2 Finite element discretization

The finite element solution process follows the standard solution methodology introduced in Chapter 3. For the quasi-harmonic equation approximates the trial function using $C_0$ shape functions we use

$$\phi \approx \hat{\phi} = \sum_a N_a \tilde{\phi}_a = \mathbf{N}\tilde{\boldsymbol{\phi}} \tag{5.8}$$

in the weak formulation (5.7) together a similar expression for $\delta\phi$.

We define the approximation to the gradient of $\phi$ as

$$\begin{aligned} \nabla\hat{\phi} &= \sum_a (\nabla N_a)\tilde{\phi}_a \\ &= \sum_a \begin{bmatrix} \dfrac{\partial N_a}{\partial x} & \dfrac{\partial N_a}{\partial y} & \dfrac{\partial N_a}{\partial z} \end{bmatrix}^{\mathrm{T}} \tilde{\phi}_a = \sum_a \mathbf{b}_a \tilde{\phi}_a \end{aligned} \tag{5.9}$$

where $\mathbf{b}_a$ denotes the *gradient matrix*.

#### 5.1.2.1 Two-dimensional plane and axisymmetric problem

The two-dimensional *plane case* is obtained by taking the gradient in the form

$$\nabla = \begin{bmatrix} \dfrac{\partial}{\partial x} & \dfrac{\partial}{\partial y} \end{bmatrix}^{\mathrm{T}} \tag{5.10}$$

and taking the flux as

$$\mathbf{q} = \begin{Bmatrix} q_x \\ q_y \end{Bmatrix} = -\begin{bmatrix} k_{xx} & k_{xy} \\ k_{yx} & k_{yy} \end{bmatrix} \begin{Bmatrix} \dfrac{\partial \phi}{\partial x} \\ \dfrac{\partial \phi}{\partial y} \end{Bmatrix} \tag{5.11}$$

On discretization a slightly simplified form of the matrices will now be found with $\mathbf{b}_a$ in Eq. (5.9) replaced by

$$\mathbf{b}_a = \begin{bmatrix} \dfrac{\partial N_a}{\partial x} & \dfrac{\partial N_a}{\partial y} \end{bmatrix}^{\mathrm{T}} \tag{5.12}$$

and the volume element replaced by

$$\mathrm{d}\Omega = h_z \,\mathrm{d}x\,\mathrm{d}y$$

where $h_z$ is the slab thickness. Alternatively the formulation may be specialized to cylindrical coordinates and used for the solution of *axisymmetric* situations by introducing the gradient

$$\nabla = \begin{bmatrix} \dfrac{\partial}{\partial r} & \dfrac{\partial}{\partial z} \end{bmatrix}^{\mathrm{T}} \tag{5.13}$$

where $r$, $z$ replace $x$, $y$ to describe both the gradient and $\mathbf{b}_a$. With the flux now given by

$$\mathbf{q} = \begin{Bmatrix} q_r \\ q_z \end{Bmatrix} = -\begin{bmatrix} k_{rr} & k_{rz} \\ k_{zr} & k_{zz} \end{bmatrix} \begin{Bmatrix} \dfrac{\partial \phi}{\partial r} \\ \dfrac{\partial \phi}{\partial z} \end{Bmatrix} \tag{5.14}$$

the discretization is now performed with the volume element expressed by

$$\mathrm{d}\Omega = 2\pi r \, \mathrm{d}r \, \mathrm{d}z$$

### 5.1.2.2 *Element matrices*

Substituting (5.8) and (5.9) into (5.7), we obtain a typical element contribution as

$$\widehat{\mathrm{G}}_e(\delta\hat{\phi}, \hat{\phi}) = \delta\tilde{\phi}_a \left[ C^e_{ab}\dot{\tilde{\phi}}_b + H^e_{ab}\tilde{\phi}_b - s^e_a \right] \tag{5.15}$$

where

$$\begin{aligned} C^e_{ab} &= \int_{\Omega_e} N_a c N_b \, \mathrm{d}\Omega \\ H^e_{ab} &= \int_{\Omega_e} \mathbf{b}^{\mathrm{T}}_a \mathbf{k} \mathbf{b}_b \, \mathrm{d}\Omega + \int_{\Gamma_{qe}} N_a H N_b \, \mathrm{d}\Gamma \\ s^e_a &= \int_{\Omega_e} N_a Q \, \mathrm{d}\Omega - \int_{\Gamma_{qe}} N_a(\bar{q} - H\phi_0)\mathrm{d}\Gamma \end{aligned} \tag{5.16}$$

define the matrix contributions to each element.

Evaluating the integrals and assembling all elements leads to the set of standard semi-discrete equations of the form

$$\mathbf{C}\dot{\tilde{\boldsymbol{\phi}}} + \mathbf{H}\tilde{\boldsymbol{\phi}} = \mathbf{s} \tag{5.17}$$

to which prescribed values of $\bar{\phi}$ have to be imposed on boundaries $\Gamma_\phi$. We again note that an additional "stiffness" is contributed on any boundaries for which a radiation constant $H$ is specified. Solution of the semi-discrete equations is considered in the next section. However, for steady-state problems the first term may be ignored and the problem solved to give

$$\tilde{\boldsymbol{\phi}} = \mathbf{H}^{-1}\mathbf{s}$$

After solution the same standard operations are followed to evaluate the fluxes using

$$\mathbf{q} \equiv -\mathbf{k}\,\nabla\phi = -\mathbf{k}\sum_a \mathbf{b}_a\tilde{\phi}_a \tag{5.18}$$

The fluxes may be easily computed within the elements; however, it is often desirable to obtain their values at nodes. This is best accomplished by the procedure to be described in Section 7.8. However, here we summarize a simple averaging method.

A least squares method may be used to project the element flux to nodal values using a functional

$$\Pi_q = \frac{1}{2}\sum_e \int_{\Omega_e} \left(q^\star - \hat{q}(\mathbf{x})\right)^2 \mathrm{d}\Omega = \text{minimum}$$

where $q$ is any component of the flux, $\hat{q}$ is the element flux, and $q^\star$ is given in each element by

$$q^\star = \sum_a N_a \tilde{q} = \mathbf{N}\tilde{\mathbf{q}}$$

Minimizing the functional gives

$$\mathbf{M}\tilde{\mathbf{q}} = \sum_e \int_{\Omega_e} \mathbf{N}^{\mathrm{T}} \hat{q}(\mathbf{x}) \mathrm{d}\Omega = \mathbf{f}$$

where

$$\mathbf{M} = \sum_e \int_{\Omega_e} \mathbf{N}^{\mathrm{T}}\mathbf{N}\,\mathrm{d}\Omega$$

The solution becomes trivial if we *diagonalize* $\mathbf{M}$ (see Appendix H) since then we obtain

$$\tilde{\mathbf{q}} = \mathbf{M}^{-1}\mathbf{f}$$

by simple divisions. An alternative to the above is the use of a local least squares on each individual element followed by averaging at each node [7].

### 5.1.3 Shape functions for triangle, rectangle, and tetrahedron

To compute the element matrices given in (5.16) it is necessary to devise appropriate shape functions. Here for two-dimensional problems we consider the simplest form for elements of triangular and rectangular form. For three-dimensional problems the simple tetrahedral element form is also developed. In the next chapter we generalize these to create families of elements for use in all problems for which $C_0$ approximations are needed.

#### 5.1.3.1 *Triangle with three nodes*

The finite element domain is defined by dividing $\Omega$ into a mesh of two-dimensional triangular elements as shown in Fig. 5.1a. A simple set of $C_0$ functions can be constructed from linear polynomials over three-node triangles as shown in Fig. 5.1b. This element was first used to solve a variational equilibrium problem by Courant [8]. Later Turner et al. used the element to solve plane elasticity problems [9].

The approximation in each triangle may be written as a linear function of the Cartesian coordinates

$$\hat{\phi}^e = \alpha_1 + \alpha_2 x + \alpha_3 y$$

The parameters $\alpha_1$ to $\alpha_3$ may be evaluated in terms of the displacements at each of the three vertices of the triangle. The vertices define the nodes of the triangle.

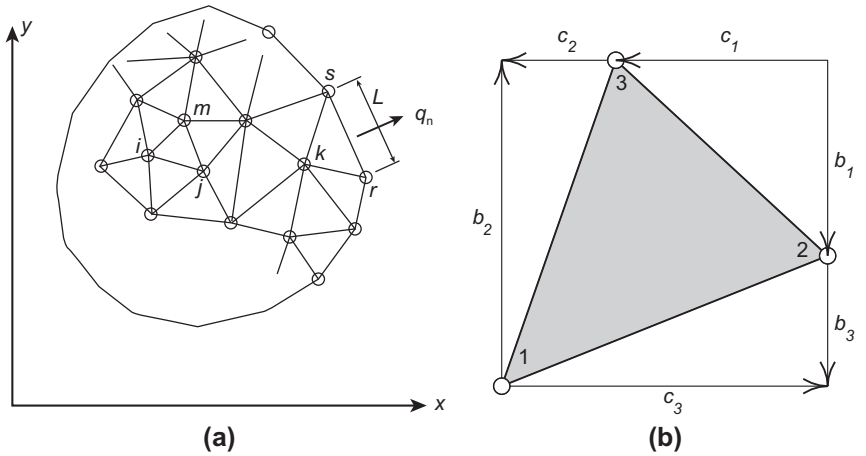



**FIGURE 5.1**
Division of a two-dimensional region into triangular elements: (a) triangular mesh and (b) geometry of triangle.

Accordingly, we write the set of equations

$$\begin{Bmatrix} \tilde{\phi}_1^e \\ \tilde{\phi}_2^e \\ \tilde{\phi}_3^e \end{Bmatrix} = \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{Bmatrix}$$

where $x_a$ and $y_a$ are coordinates at the three vertices of the triangle. The inverse to the coefficient matrix is given by

$$\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix}^{-1} = \frac{1}{2\Delta} \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}$$

where

$$\begin{aligned} a_1 &= x_2y_3 - x_3y_2, & b_1 &= y_2 - y_3, & c_1 &= x_3 - x_2 \\ a_2 &= x_3y_1 - x_1y_3, & b_2 &= y_3 - y_1, & c_2 &= x_1 - x_3 \\ a_3 &= x_1y_2 - x_2y_1, & b_3 &= y_1 - y_2, & c_3 &= x_2 - x_1 \end{aligned}$$

and $\Delta = (x_1b_1 + x_2b_2 + x_3b_3)/2$ is the area of the triangle. The meaning of the $b_a$ and $c_a$ parameters is shown in Fig. 5.1b.

The above solution for the parameters $\alpha_a$ permits the element interpolations to be rewritten in terms of nodal parameters as

$$\hat{\phi}^e = \sum_{a=1}^{3} \frac{1}{2\Delta} \left(a_a + b_a x + c_a y\right) \tilde{\phi}_a^e$$

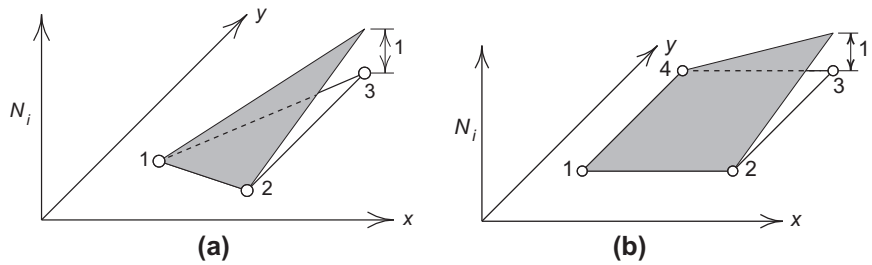



**FIGURE 5.2**
Shape function $N_3$ for one element: (a) three-node triangle and (b) four-node rectangle.

Thus, the three shape functions for the triangle are given by

$$N_a(x, y) = \frac{1}{2\Delta}\left(a_a + b_a x + c_a y\right), \quad a = 1, 2, 3 \tag{5.19}$$

The shape function for $a = 3$ is shown in Fig. 5.2a. With this definition it is clear that we can write the set of approximations in each individual element as

$$\hat{\phi}^e = \sum_{a=1}^{3} N_a(x, y)\tilde{\phi}_a^e$$

Since these shape functions vary linearly along any side of a triangle, with identical nodal values imposed, the same value of the function will clearly exist along an interface between adjacent elements. We note, however, that the derivatives may not be continuous between elements; consequently, the above form only provides $C_0$ continuity.

### 5.1.3.2 *Rectangle with four nodes*

As a second example of two-dimensional shape functions we consider rectangles of the form shown in Fig. 5.3. The rectangular element considered has side lengths of $2a$ and $2b$ in the $x$- and $y$-directions, respectively. For the derivation of the shape functions it is convenient to use a local Cartesian system $x'$, $y'$ defined by

$$x' = x - x_0 \quad \text{and} \quad y' = y - y_0$$

where

$$x_0 = \frac{1}{4}\sum_{a=1}^{4} y_a \quad \text{and} \quad y_0 = \frac{1}{4}\sum_{a=1}^{4} x_a$$

in which $x_0$, $y_0$ are located at the center of the rectangle and $x_a$, $y_a$ are coordinates of the nodes. We now need four functions for each displacement component in order to uniquely define the shape functions. In addition these functions must have linear

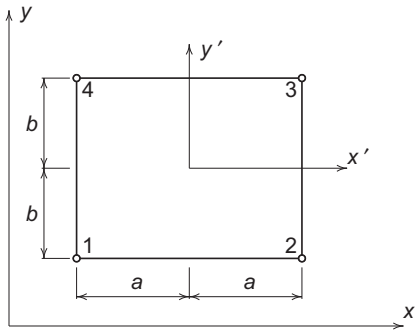



**FIGURE 5.3**
Rectangular element geometry and local node numbers.

behavior along each edge of the element to ensure interelement $C_0$ continuity. A suitable choice is given by

$$\hat{\phi}^e = \alpha_1 + x'\alpha_2 + y'\alpha_3 + x'y'\alpha_4 \tag{5.20}$$

The coefficients $\alpha_a$ may be obtained by expressing (5.20) at each vertex node giving

$$\begin{Bmatrix} \tilde{\phi}_1^e \\ \tilde{\phi}_2^e \\ \tilde{\phi}_3^e \\ \tilde{\phi}_4^e \end{Bmatrix} = \begin{bmatrix} 1 & -a & -b & ab \\ 1 & a & -b & -ab \\ 1 & a & b & ab \\ 1 & -a & b & -ab \end{bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{Bmatrix}$$

We can again solve for $\alpha_a$ in terms of the nodal displacements to obtain finally

$$\begin{aligned} \hat{\phi}^e = {} & \frac{1}{4}\left(1 - \frac{x'}{a}\right)\left(1 - \frac{y'}{b}\right)\tilde{\phi}_1^e + \frac{1}{4}\left(1 + \frac{x'}{a}\right)\left(1 - \frac{y'}{b}\right)\tilde{\phi}_2^e \\ & + \frac{1}{4}\left(1 + \frac{x'}{a}\right)\left(1 + \frac{y'}{b}\right)\tilde{\phi}_3^e + \frac{1}{4}\left(1 - \frac{x'}{a}\right)\left(1 + \frac{y'}{b}\right)\tilde{\phi}_4^e \end{aligned} \tag{5.21}$$

From (5.21) we obtain the four shape functions

$$\begin{aligned} N_1 &= \frac{1}{4}\left(1 - \frac{x'}{a}\right)\left(1 - \frac{y'}{b}\right), & N_2 &= \frac{1}{4}\left(1 + \frac{x'}{a}\right)\left(1 - \frac{y'}{b}\right) \\ N_3 &= \frac{1}{4}\left(1 + \frac{x'}{a}\right)\left(1 + \frac{y'}{b}\right), & N_4 &= \frac{1}{4}\left(1 - \frac{x'}{a}\right)\left(1 + \frac{y'}{b}\right) \end{aligned} \tag{5.22}$$

The shape function for $N_3$ is shown in Fig. 5.2b.

#### 5.1.3.3 *Tetrahedron with four nodes*

For three-dimensional problems a simple element is a tetrahedron with four nodes as shown in Fig. 5.4. A $C_0$ compatible displacement field is again given

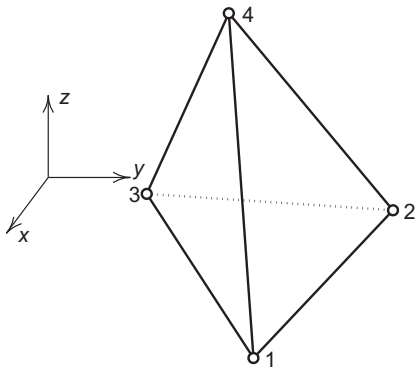



**FIGURE 5.4**
Tetrahedron element.

by a complete linear polynomial expansion as

$$\hat{\phi}^e = [\,1 \quad x \quad y \quad z\,] \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{Bmatrix} \tag{5.23}$$

The parameters $\alpha_a$ can be evaluated by evaluating (5.23) at each of the vertex nodes of the tetrahedron. Accordingly we have

$$\begin{Bmatrix} \tilde{\phi}_1^e \\ \tilde{\phi}_2^e \\ \tilde{\phi}_3^e \\ \tilde{\phi}_4^e \end{Bmatrix} = \begin{bmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{Bmatrix}$$

The inverse is expressed as

$$\begin{bmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{bmatrix}^{-1} = \frac{1}{6V} \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ d_1 & d_2 & d_3 & d_4 \end{bmatrix}$$

where

$$6V = \det \begin{bmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{bmatrix} \tag{5.24a}$$

and

$$a_1 = \det\begin{bmatrix} x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \\ x_4 & y_4 & z_4 \end{bmatrix}, \quad b_1 = \det\begin{bmatrix} y_2 & z_2 & 1 \\ y_3 & z_3 & 1 \\ y_4 & z_4 & 1 \end{bmatrix}$$
$$c_1 = \det\begin{bmatrix} z_2 & 1 & x_2 \\ z_3 & 1 & x_3 \\ z_4 & 1 & x_4 \end{bmatrix}, \quad d_1 = \det\begin{bmatrix} 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \\ 1 & x_4 & y_4 \end{bmatrix} \tag{5.24b}$$

with the other constants defined by cyclic interchange of the subscripts in the order 1, 2, 3, 4. This gives the shape functions

$$N_a(x, y, z) = \frac{1}{6V}\left(a_a + b_a x + c_a y + d_a z\right), \quad a = 1, 2, 3, 4 \tag{5.25}$$

Integrals to compute element matrices may be carried out using results in Appendix E.

In the next chapter we shall generalize the above element forms to permit a systematic development with polynomials of any degree, as well as to permit curved edges. The above forms, however, provide a basis from which we can consider solutions to a number of different problem forms for the quasi-harmonic equation. We begin by describing the element matrices for the simple triangular element.

**Example 5.1.** Element arrays for plane three-node triangular element

With shape functions written in the form

$$N_a = \frac{a_a + b_a x + c_a y}{2\Delta}$$

in which $\Delta$ and $a_a$, $b_a$, $c_a$ are defined in Section 5.1.3, the computation of the "mass" matrix, $\mathbf{C}^e$, is given by

$$C^e_{ab} = \int_\Delta N_a c N_b h_z \, \mathrm{d}x \, \mathrm{d}y$$

which for $c$ constant over the element gives

$$\mathbf{C}^e = \frac{c h_z \Delta}{12}\begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix}$$

This result may be computed using the exact expression given in Appendix E.

The derivatives of the shape functions are expressed as

$$\frac{\partial N_a}{\partial x} = \frac{b_a}{2\Delta}, \qquad \frac{\partial N_a}{\partial y} = \frac{c_a}{2\Delta}$$

giving the gradient matrix

$$\mathbf{b}_a = \frac{1}{2\Delta}\begin{bmatrix} b_a & c_a \end{bmatrix}^{\mathrm{T}}$$

Since the gradient matrix is constant the element "stiffness" matrix (ignoring the $H$ boundary term) is given by (noting $k_{xy} = k_{yx}$)

$$\mathbf{H}^e = \left\{ \frac{k_{xx}}{4\Delta} \begin{bmatrix} b_1b_1 & b_1b_2 & b_1b_3 \\ b_2b_1 & b_2b_2 & b_2b_3 \\ b_3b_1 & b_3b_2 & b_3b_3 \end{bmatrix} + \frac{k_{yy}}{4\Delta} \begin{bmatrix} c_1c_1 & c_1c_2 & c_1c_3 \\ c_2c_1 & c_2c_2 & c_2c_3 \\ c_3c_1 & c_3c_2 & c_3c_3 \end{bmatrix} + \frac{k_{xy}}{4\Delta} \begin{bmatrix} (b_1c_1 + c_1b_1) & (b_1c_2 + c_1b_2) & (b_1c_3 + c_1b_3) \\ (b_2c_1 + c_2b_1) & (b_2c_2 + c_2b_2) & (b_2c_3 + c_2b_3) \\ (b_3c_1 + c_3b_1) & (b_3c_2 + c_3b_2) & (b_3c_3 + c_3b_3) \end{bmatrix} \right\} h_z$$

The load matrices follow a similar simple pattern and thus, for instance, due to constant $Q$ we have

$$s_a^e = \int_\Delta N_a Q h_z \, \mathrm{d}x \, \mathrm{d}y = \frac{1}{3} Q h_z \Delta$$

This is a very simple (almost "obvious") result.

**Example 5.2.** "Stiffness" matrix for axisymmetric three-node triangular element

The computation of the arrays for an axisymmetric problem involves an integral of the radius over the area of the triangle. This is given by [viz. Appendix E]

$$\int_\Delta r \, \mathrm{d}r \, \mathrm{d}z = \bar{r}\Delta$$

where $\bar{r} = (r_1 + r_2 + r_3)/3$. This then gives the results

$$\mathbf{H}^e = \left\{ \frac{k_{rr}}{4\Delta} \begin{bmatrix} b_1b_1 & b_1b_2 & b_1b_3 \\ b_2b_1 & b_2b_2 & b_2b_3 \\ b_3b_1 & b_3b_2 & b_3b_3 \end{bmatrix} + \frac{k_{zz}}{4\Delta} \begin{bmatrix} c_1c_1 & c_1c_2 & c_1c_3 \\ c_2c_1 & c_2c_2 & c_2c_3 \\ c_3c_1 & c_3c_2 & c_3c_3 \end{bmatrix} + \frac{k_{rz}}{4\Delta} \begin{bmatrix} (b_1c_1 + c_1b_1) & (b_1c_2 + c_1b_2) & (b_1c_3 + c_1b_3) \\ (b_2c_1 + c_2b_1) & (b_2c_2 + c_2b_2) & (b_2c_3 + c_2b_3) \\ (b_3c_1 + c_3b_1) & (b_3c_2 + c_3b_2) & (b_3c_3 + c_3b_3) \end{bmatrix} \right\} 2\pi\bar{r}$$

**Example 5.3.** Load matrix for axisymmetric three-node triangular element

The nodal forces from a constant source term $Q$ are computed from

$$s_a^e = \int_\Delta N_a Q 2\pi r_b N_b \, \mathrm{d}r \, \mathrm{d}z$$

and thus now has quadratic terms in the coordinates. If we substitute

$$r = \bar{r} + x$$

and use the results from Appendix E we obtain for node 1

$$s_1^e = \frac{1}{6}(2r_1 + r_2 + r_3)\pi Q\Delta$$

with results for $s_2^e$ and $s_3^e$ obtained by cyclic permutation.

## 5.2 Partial discretization: Transient problems

The transient problem may be solved in a number of ways. In Chapter 3 we considered a simple discrete time method for the one-dimensional problem in terms of the rate. We again consider the procedure described in Section 3.9.3 in which the time dependence is given by (3.96). However, here we will formulate our solution in terms of the discrete values of $\phi$ itself. If we let

$$\widehat{\boldsymbol{\phi}}_{n+1} = \tilde{\boldsymbol{\phi}}_n + (1-\theta)\Delta t \dot{\tilde{\boldsymbol{\phi}}}_n$$

denote all the terms from the solution at time $t_n$, from (3.96) the approximation to the rate at $t_{n+1}$ is given by

$$\dot{\tilde{\boldsymbol{\phi}}}_{n+1} = \frac{1}{\theta\,\Delta t}\left(\tilde{\boldsymbol{\phi}}_{n+1} - \widehat{\boldsymbol{\phi}}_{n+1}\right)$$

where $\Delta t = t_n - t_{n-1}$ and $\theta > 0$. An approximate solution to the semi-discrete equations at each time $t_{n+1}$ is obtained by solving the set of equations

$$\left[\frac{1}{\theta\,\Delta t}\mathbf{C} + \mathbf{H}\right]\tilde{\boldsymbol{\phi}}_{n+1} = \mathbf{s}_{n+1} + \frac{1}{\theta\,\Delta t}\mathbf{C}\widehat{\boldsymbol{\phi}}_{n+1} \tag{5.26}$$

If the initial condition is approximated as

$$\phi(\mathbf{x}, 0) \approx \mathbf{N}(\mathbf{x})\tilde{\boldsymbol{\phi}}(0) \qquad \text{with } \tilde{\boldsymbol{\phi}}(0) = \tilde{\boldsymbol{\phi}}_0$$

a solution for $\tilde{\boldsymbol{\phi}}_1$ is immediately available from (5.26) by solving a set of *algebraic equations*. For each subsequent time step the solution process is identical to the time-independent problem except for the modified force vector and a need to use a coefficient matrix which has a term inversely proportional to the size of the time increment.

## 5.3 Numerical examples: An assessment of accuracy

Assembling explicitly worked out "stiffnesses" of triangular elements for the "regular" mesh pattern shown in Fig. 5.5a the discretized equations are *identical* with those that are derived by well-known finite difference methods [49]. The same result holds for the mesh pattern shown in Fig. 5.5b [10]. For cases where all boundary conditions are given as prescribed values

$$\phi = \bar{\phi} \qquad \text{on } \Gamma_\phi$$

the solutions obtained by the two methods obviously will be identical, and so also will be the orders of approximation.

However, if the mesh shown in Fig. 5.5c, which is also based on a square arrangement of nodes but with an "irregular" element pattern, is used a difference between the two approaches for the "load" vector $\mathbf{s}^e$ will be evident. The assembled equations will have the same "stiffness" matrix as in Fig. 5.5a but will show "loads" which

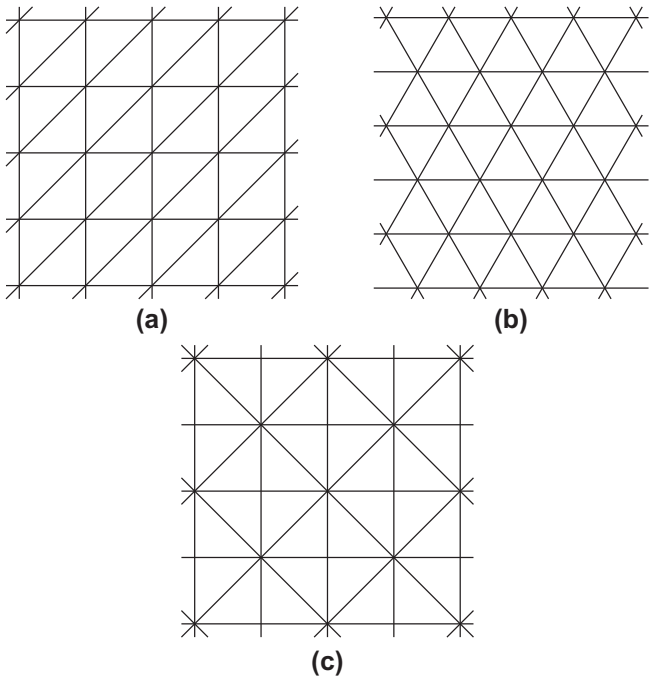



**FIGURE 5.5**
"Regular" and "irregular" subdivision patterns.

differ by small amounts from node to node, the sum of the loads is still the same as that due to the finite difference expressions. The solutions therefore differ only locally and will represent the same averages.

Further advantages of the finite element process are:

1. It can deal simply with nonhomogeneous and anisotropic situations (particularly when the direction of anisotropy is variable).
2. The elements can be graded in shape and size to follow arbitrary boundaries and to allow for regions of rapid variation of the function sought, thus controlling the errors in a most efficient way (viz. Chapters 15 and 16).
3. Specified gradient or "radiation" boundary conditions are introduced naturally and with a better accuracy than in standard finite difference procedures.
4. Higher order elements presented in the next chapter can be readily used to improve accuracy without complicating boundary conditions—a difficulty always arising with finite difference approximations of a higher order.
5. Finally, but of considerable importance in the computer age, standard programs may be used for assembly and solution.

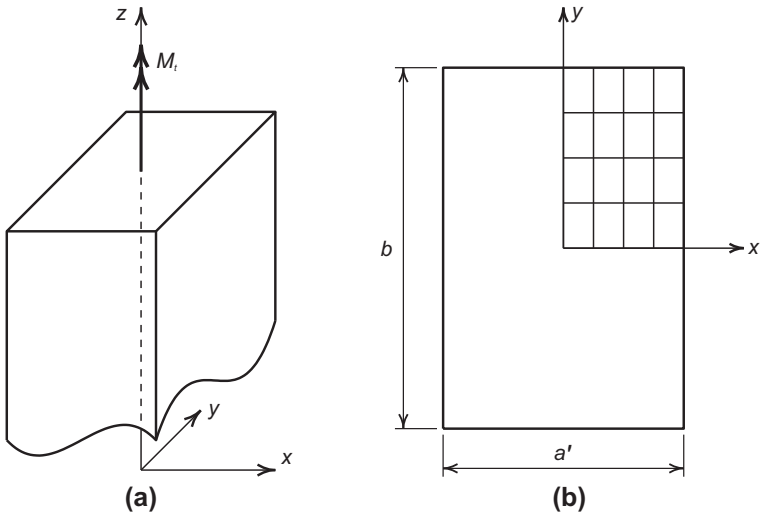



**FIGURE 5.6**
Torsion of rectangular prismatic bar.

### 5.3.1 Torsion of prismatic bars

The torsion of prismatic elastic bars may be solved using a quasi-harmonic equation formulation. Here either a *warping function* or a *stress function* approach may be used. In Fig. 5.6a we show a rectangular bar loaded by an end torque $M_t$. The analysis is performed on the cross-section as shown in Fig. 5.6b.

The use of a warping function is governed by the formulation in which displacements are given as

$$u = -yz\theta, \qquad v = xz\theta, \qquad \text{and} \qquad w = \psi(x, y)\theta$$

where $x$, $y$ are coordinates in the cross-section, $z$ is a coordinate of the bar axis, $\theta$ is the rate of twist, and $\psi$ is the warping function. The nonzero strain components resulting from these displacements are given by

$$\gamma_{xz} = \theta\left(\frac{\partial\psi}{\partial x} - y\right) \qquad \text{and} \qquad \gamma_{yz} = \theta\left(\frac{\partial\psi}{\partial y} + x\right) \tag{5.27}$$

giving, for an isotropic elastic material, the nonzero stresses

$$\tau_{xz} = G\gamma_{xz} \qquad \text{and} \qquad \tau_{yz} = G\gamma_{yz} \tag{5.28}$$

Inserting the expression for stresses into the equilibrium equation (2.18) with zero body and inertia forces gives the governing differential equation

$$\theta\left[\frac{\partial}{\partial x}\left(G\frac{\partial\psi}{\partial x}\right) + \frac{\partial}{\partial y}\left(G\frac{\partial\psi}{\partial y}\right)\right] = 0 \tag{5.29}$$

and for stress-free boundary conditions

$$\tau_{nz} = n_x \tau_{xz} + n_y \tau_{yz} = 0 \tag{5.30}$$

in which $n_x$ and $n_y$ are the direction cosines for the outward normal to the boundary of the rectangular section. Thus, for $\theta G = k_{xx} = k_{yy}, k_{xy} = 0$ we have the standard quasi-harmonic equation. Introducing appropriate changes in parameter definitions into (5.7), the problem is solved from the weak form expressed as

$$\mathrm{G}(\delta\psi, \psi) = \theta \int_\Omega \left[ \frac{\partial \delta\psi}{\partial x} G \left( \frac{\partial \psi}{\partial x} - y \right) + \frac{\partial \delta\psi}{\partial y} G \left( \frac{\partial \psi}{\partial y} + x \right) \right] \mathrm{d}\Omega = 0$$

At least one value of the warping function must be specified to have a unique solution. In addition a unit value of $\theta$ may be used in $G$ during computation of $\psi$. In this case the element $s_a$ is given by

$$s_a = \int_\Omega \left( \frac{\partial N_a}{\partial x} y - \frac{\partial N_a}{\partial y} x \right) \mathrm{d}\Omega$$

The total torque acting on a cross-section is given by

$$\begin{aligned} M_t &= \int_A \left[ -\tau_{xz} y + \tau_{yz} x \right] \mathrm{d}A \\ &= \int_A G \left[ x^2 + y^2 - y \frac{\partial \psi}{\partial x} + x \frac{\partial \psi}{\partial y} \right] \mathrm{d}A \, \theta = \overline{GJ}_\psi \theta \end{aligned}$$

where $\overline{GJ}_\psi$ is the effective torsional stiffness.

As an alternative a stress function formulation is deduced using the representation for stresses

$$\tau_{xz} = -\frac{\partial \phi}{\partial y} \quad \text{and} \quad \tau_{yz} = \frac{\partial \phi}{\partial x} \tag{5.31}$$

Combining (5.27) and (5.28) with (5.31) and eliminating the warping function $\psi$ gives the differential equation

$$\frac{\partial}{\partial x} \left( \frac{1}{G} \frac{\partial \phi}{\partial x} \right) + \frac{\partial}{\partial y} \left( \frac{1}{G} \frac{\partial \phi}{\partial y} \right) = 2\theta$$

with

$$\phi(s) = \text{Constant} \qquad \text{on } \Gamma_q$$

representing a stress-free boundary condition.

The total torque acting on a cross-shaded section is now given by

$$M_t = \int_A \left[ x \frac{\partial \phi}{\partial x} + y \frac{\partial \phi}{\partial y} \right] \mathrm{d}A = \overline{GJ}_\phi \theta$$

where $\overline{GJ}_\phi$ is the effective torsional stiffness.

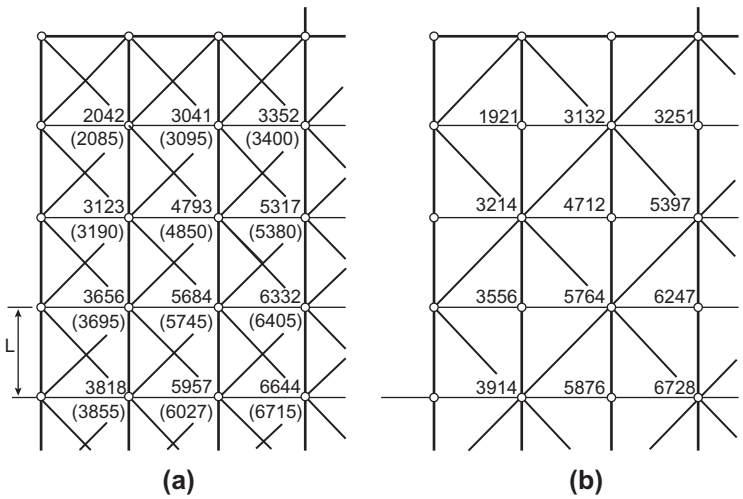



**FIGURE 5.7**
Torsion of a rectangular shaft. The numbers in parentheses show a more accurate solution due to Southwell using a $12 \times 16$ mesh (values of $\phi/G\theta L^2$).

The two solutions provide a bound on the torsional stiffness with the warping function solution giving an upper bound, $\overline{GJ}_\psi$, and the stress function a lower bound, $\overline{GJ}_\phi$.

#### 5.3.1.1 *Torsion of rectangular shaft*

In Fig. 5.7 a test comparing the results obtained on an "irregular" mesh of three-node triangular elements with a relaxation solution of the lowest order finite difference approximation is shown. Both give results of similar accuracy, as indeed would be anticipated. In general superior accuracy is available with the finite element discretization. Furthermore, it is possible to get bounds on the torsional stiffness, as indicated above. To illustrate this latter aspect we consider a square bar which is solved using four-node rectangular elements and a range of $n \times n$ meshes in which $n$ is the number of spaces between nodes on each side. The results for the computed torsional stiffness values are plotted in Fig. 5.8.

#### 5.3.1.2 *Torsion of hollow bimetallic shaft*

The pure torsion of a nonhomogeneous rectangular shaft with a circular hole is illustrated in Fig. 5.9. In the finite element solution presented, the hollow section is represented by a material for which $G$ has a value of the order of $10^{-3}$ compared with the other materials.[1] The results compare well with the contours derived from an accurate finite difference solution [11].

[1]This was done to avoid difficulties due to the "multiple connection" of the region and to permit the use of a standard program.

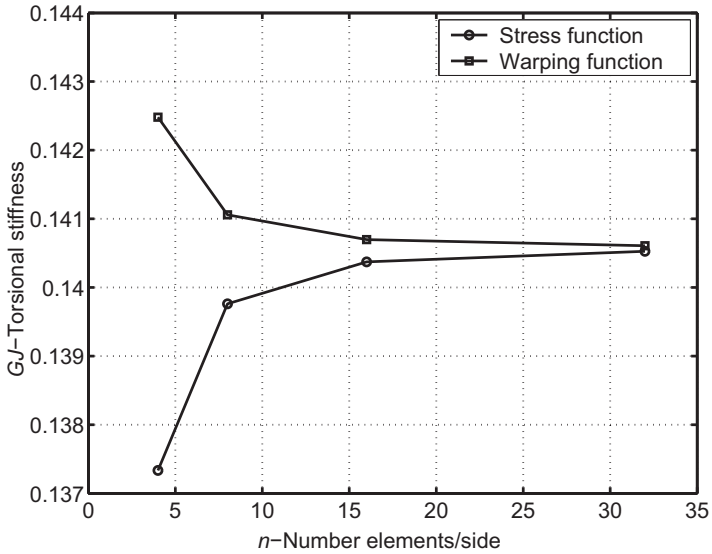



**FIGURE 5.8**

Bound on torsional stiffness for square bar.

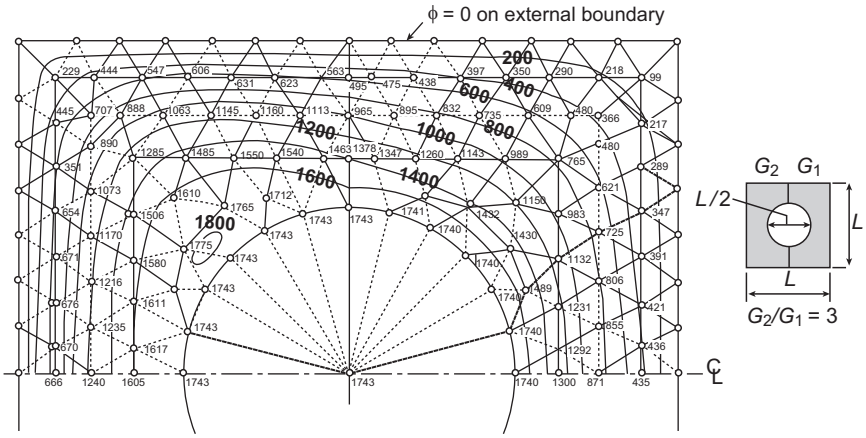



**FIGURE 5.9**

Torsion of a hollow bimetallic shaft. $\phi/G\theta L^2 \times 10^4$.

## 5.3.2 Transient heat conduction

### *5.3.2.1 Transient heat conduction of a rectangular bar*

Consider a long bar with a square cross-section of size $L \times L$ in which the transient heat conduction equation (2.84) applies and assume that the rate of heat generation

varies with time as

$$Q = Q_0 \, \mathrm{e}^{-\alpha t} \tag{5.32}$$

(this might approximate a problem of heat development due to hydration of concrete). We assume that at $t = 0, \phi = 0$ throughout. Further, we shall take $\phi = 0$ on all boundaries for all times.

A Fourier series approximation for the solution is given by

$$\begin{aligned} \phi &= \sum_{m=1}^{M} \sum_{n=1}^{N} N_{mn}(x, y) \tilde{\phi}_{mn}(t) \\ N_{mn} &= \cos \frac{m\pi x}{L} \cos \frac{n\pi y}{L}, \quad m, n = 1, 3, 5, \ldots \end{aligned} \tag{5.33}$$

with $x$ and $y$ measured from the center (Fig. 5.10). The even components of the Fourier series are omitted due to the required symmetry of solution. When substituted into the weak form only diagonal terms exist in $\mathbf{H}$ and $\mathbf{C}$, and we have

$$\begin{aligned} H_{mn} &= \int_{-L/2}^{L/2} \int_{-L/2}^{L/2} \left[ k \left( \frac{\partial N_{mn}}{\partial x} \right)^2 + k \left( \frac{\partial N_{mn}}{\partial y} \right)^2 \right] \mathrm{d}x \, \mathrm{d}y = \frac{\pi^2 k}{4} (m^2 + n^2) \\ C_{mn} &= \int_{-L/2}^{L/2} \int_{-L/2}^{L/2} c N_{mn}^2 \, \mathrm{d}x \, \mathrm{d}y = \frac{L^2 c}{4} \\ s_{mn} &= \int_{-L/2}^{L/2} \int_{-L/2}^{L/2} N_{mn} Q_0 \, \mathrm{e}^{-\alpha t} \, \mathrm{d}x \, \mathrm{d}y \\ &= \frac{4 Q_0 L^2}{mn\pi^2} (-1)^{(m+3)/2} (-1)^{(n+3)/2} \, \mathrm{e}^{-\alpha t} \end{aligned} \tag{5.34}$$

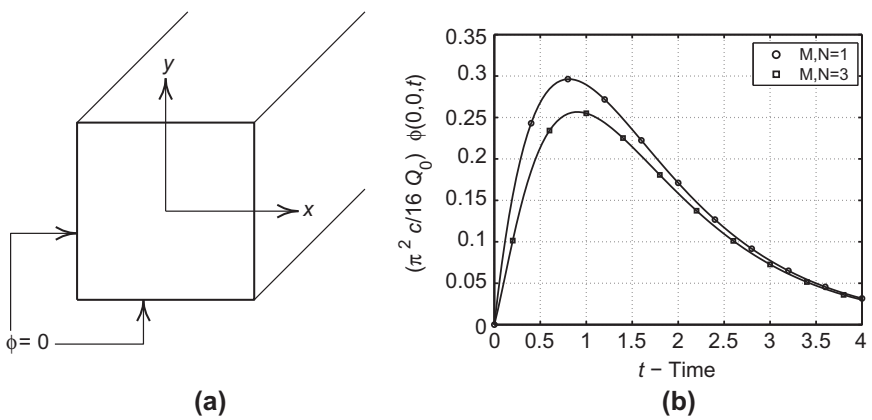



**FIGURE 5.10**
Two-dimensional transient heat development in a square prism—plot of temperature at center: (a) boundary condition and (b) solution at center.

This leads to an ordinary differential equation with parameters $\tilde{\phi}_{mn}$:

$$H_{mn}\tilde{\phi}_{mn} + C_{mn}\frac{\mathrm{d}\tilde{\phi}_{mn}}{\mathrm{d}t} + s_{mn} = 0 \tag{5.35}$$

with initial condition $\tilde{\phi}_{mn} = 0$ when $t = 0$. The exact solution of this is easy to obtain, as is shown in Fig. 5.10 for the parameters

$$L = c = Q_0 = \alpha = 1 \qquad \text{and} \qquad k = \frac{0.75}{\pi^2}$$

and choices for $M$ and $N$.

The above solution also may be used to assess the accuracy of a finite element result. For the finite element solution we use four-node square elements. The transient solution is performed using the procedure given in Section 5.2. Using symmetry conditions, a mesh of $20 \times 20$ four-node elements is used to approximate one quadrant of the domain. A constant increment in time, $\Delta t = 0.01$, is used to perform the solution. Results for the temperature at the center of the prism are given in Fig. 5.11 and compared to the series solutions.

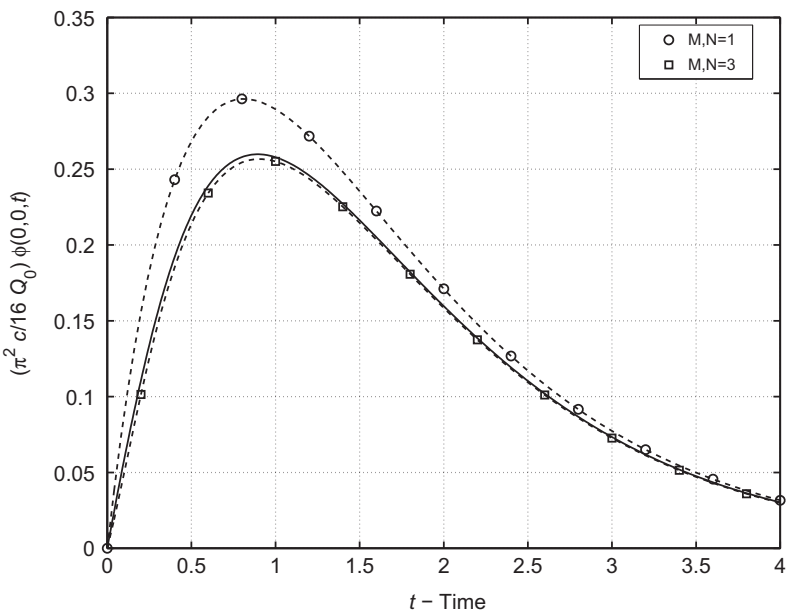



**FIGURE 5.11**
Transient heat development in a square prism—plot of temperature at center.

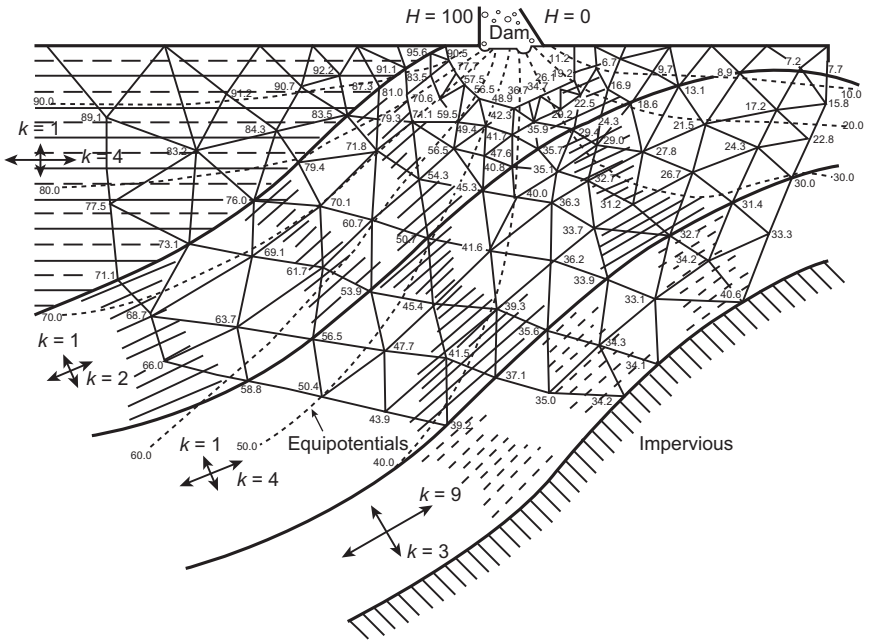



**FIGURE 5.12**
Flow under a dam through a highly nonhomogeneous and contorted foundation.

### 5.3.3 Anisotropic seepage

The next problem is concerned with the flow through highly nonhomogeneous, anisotropic, and contorted strata. The basic governing equation is

$$\frac{\partial}{\partial x'}\left(k_{x'x'}\frac{\partial H}{\partial x'}\right)+\frac{\partial}{\partial y'}\left(k_{y'y'}\frac{\partial H}{\partial y'}\right)=0 \tag{5.36}$$

in which $H$ is the hydraulic head and $k_{x'x'}$ and $k_{y'y'}$ represent the permeability coefficients in the direction of the (inclined) principal axes. However, a special feature has to be incorporated to allow for changes of $x'$ and $y'$ principal directions from element to element.

No difficulties are encountered in computation, and the problem together with its solution is given in Fig. 5.12 [3].

### 5.3.4 Electrostatic and magnetostatic problems

In this area of activity frequent need arises to determine appropriate field strengths and the governing equations are commonly of the standard quasi-harmonic type discussed here. Thus the formulations are directly transferable. One of the first applications made as early as 1967 [4] was to fully three-dimensional electrostatic field distributions governed by simple Laplace equations (Fig. 5.13).

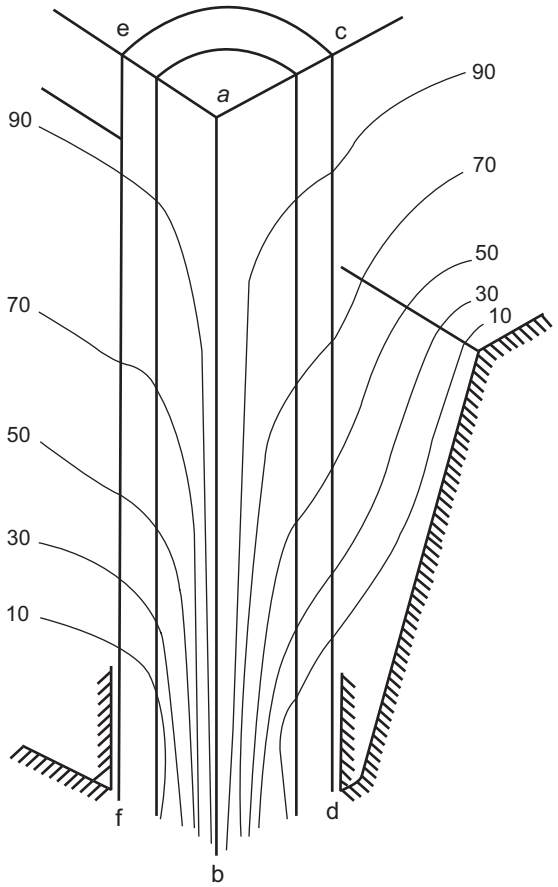



**FIGURE 5.13**
A three-dimensional distribution of electrostatic potential around a porcelain insulator in an earthed trough.

In Fig. 5.14 a similar use of triangular elements was made in the context of magnetic two-dimensional fields by Winslow [6]. These early works stimulated considerable activity in this area and much additional work has been published [12–15].

### 5.3.5 Lubrication problems

Once again a standard Poisson type of equation is encountered in the two-dimensional domain of a bearing pad. In the simplest case of constant lubricant density and viscosity the equation to be solved is the Reynolds equation

$$\frac{\partial}{\partial x}\left(h^3\frac{\partial p}{\partial x}\right)+\frac{\partial}{\partial y}\left(h^3\frac{\partial p}{\partial y}\right)=6\mu V\frac{\partial h}{\partial x} \tag{5.37}$$

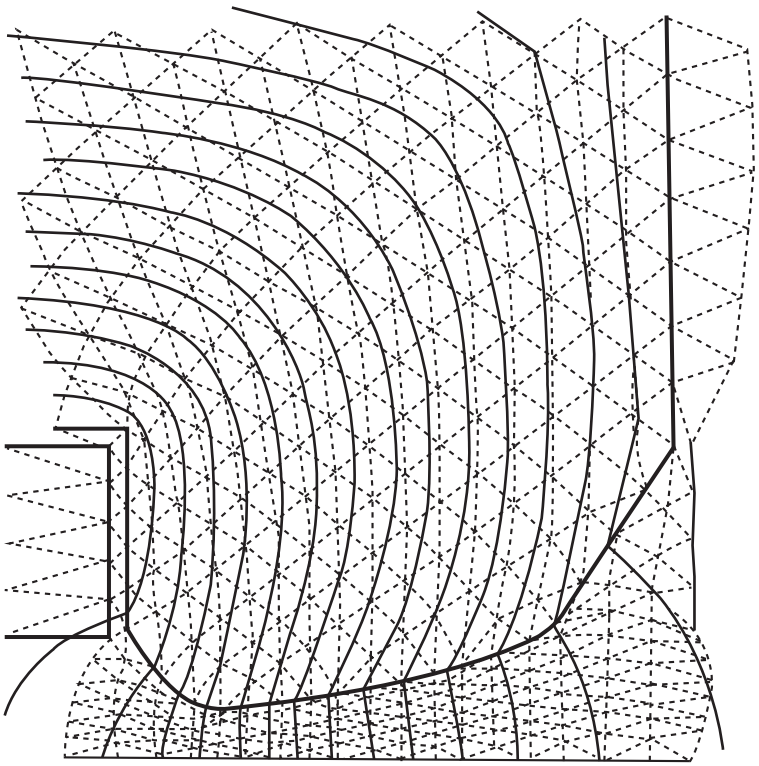


**FIGURE 5.14**
Field near a magnet.

where $h$ is the film thickness, $p$ is the pressure developed, $\mu$ is the viscosity, and $V$ is the velocity of the pad in the $x$-direction.

Figure 5.15 shows the pressure distribution in a typical finite width stepped pad [16]. The boundary condition is simply that of zero pressure and it is of interest to note that the step causes an equivalent of a "line load" on integration by parts of the right-hand side of Eq. (5.37).

More general cases of lubrication problems, including vertical pad movements (squeeze films) and compressibility, can obviously be dealt with, and much work has been done here [17–25].

### 5.3.6 Irrotational and free surface flows

The basic Laplace equation which governs the flow of viscous fluid in seepage problems is also applicable in the problem of irrotational fluid flow outside the boundary layer created by viscous effects. The seepage example given above is adequate to illustrate the general applicability in this context. Further examples for this class of problems are cited by Martin [26] and others [25,27–32].

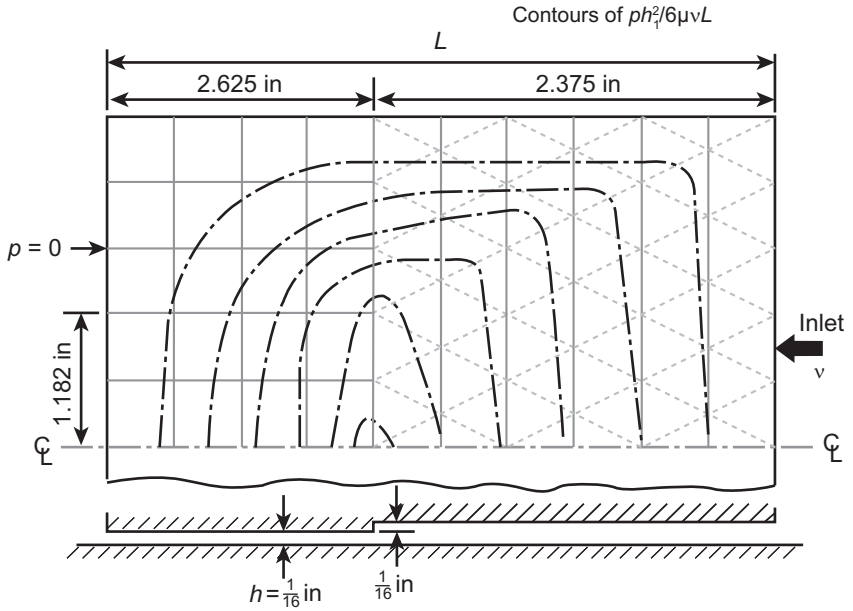



**FIGURE 5.15**
Pressure distribution for a stepped pad bearing.

If no viscous effects exist, then it can be shown that for a fluid starting at rest the motion must be irrotational, i.e.,

$$\omega_z \equiv \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x} = 0 \tag{5.38}$$

where $u$ and $v$ are appropriate velocity components.

This implies the existence of a velocity potential, giving

$$u = -\frac{\partial \phi}{\partial x}, \qquad v = -\frac{\partial \phi}{\partial y} \tag{5.39a}$$

or

$$\mathbf{u} = -\nabla \phi \tag{5.39b}$$

If, further, the flow is incompressible, the continuity equation [which is similar to Eq. (2.80)] has to be satisfied, i.e.,

$$\nabla^{\mathrm{T}} \mathbf{u} = 0 \tag{5.40}$$

and therefore

$$\nabla^{\mathrm{T}} (\nabla \phi) = \nabla^2 \phi = 0 \tag{5.41}$$

Alternatively, for two-dimensional flow a stream function may be introduced defining the velocities as

$$u = -\frac{\partial \psi}{\partial y}, \qquad v = \frac{\partial \psi}{\partial x} \tag{5.42}$$

and this identically satisfies the continuity equation. The irrotational condition must now ensure that

$$\nabla^{\mathrm{T}}(\nabla\psi) = \nabla^2\psi = 0 \tag{5.43}$$

and thus problems of ideal fluid flow can be posed in either form. As the standard formulation is again applicable, there is little more that needs to be added, and for examples the reader can consult the literature cited. We also discuss this problem in more detail in Ref. [33].

The similarity with problems of seepage flow, which has already been discussed, is obvious [34,35].

A particular class of fluid flow deserves mention. This is the case when a free surface limits the extent of the flow and this surface is not known *a priori*.

The class of problem is typified by two examples—that of a freely overflowing jet (Fig. 5.16a) and that of flow through an earth dam (Fig. 5.16b). In both, the free surface represents a streamline and in both the position of the free surface is unknown *a priori* but has to be determined so that an *additional condition* on this surface is satisfied. For instance, in the second problem, if formulated in terms of the potential for the hydraulic head $H$, Eq. (5.36) governs the problem.

The free surface, being a streamline, imposes the condition that

$$\frac{\partial H}{\partial n} = 0 \tag{5.44}$$

be satisfied there. In addition, however, the pressure must be zero on the surface as this is exposed to atmosphere. As

$$H = \frac{p}{\gamma} + y \tag{5.45}$$

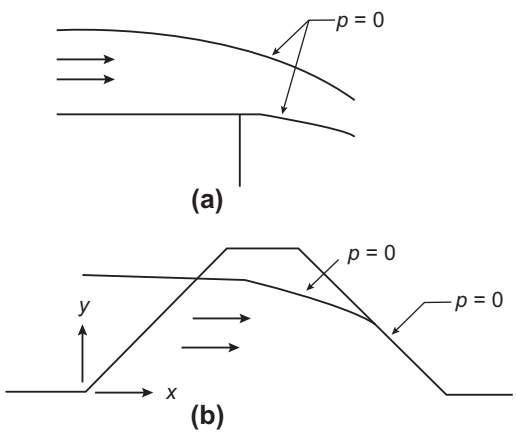



**FIGURE 5.16**
Typical free surface problems with a streamline also satisfying an additional condition of pressure = 0: (a) jet overflow and (b) seepage through an earth dam.

where $\gamma$ is the fluid specific weight, $p$ is the fluid pressure, and $y$ is the elevation above some (horizontal) datum, we must have on the surface

$$H = y \tag{5.46}$$

The solution may be approached iteratively. Starting with a prescribed free surface streamline the standard problem is solved. A check is carried out to see if Eq. (5.46) is satisfied and, if not, an adjustment of the surface is carried out to make the new $y$ equal to the $H$ just found. A few iterations of this kind show that convergence is reasonably rapid. Taylor and Brown [36] show such a process. Alternative methods including special variational principles for dealing with this problem have been devised over the years and interested readers can consult Refs. [37–45].

## 5.4 Problems

**5.1** The anisotropic properties for $\mathbf{k}$ are $k_{x'} = 0.4$, $k_{y'} = 2.1$, and $k_{z'} = 1.0$. The axes are oriented as shown in Fig. 5.17. For $\theta = 30°$ compute the terms in the matrix $\mathbf{k}$ (e.g., $k_{xx}$, $k_{xy}$, etc.) with respect to the axes $x$, $y$, $z$.

**5.2** A two-dimensional heat equation is located in the $x$-$y$ plane. The problem is allowed to convect heat to the surrounding region according to

$$Q(x, y) = -\beta[\phi(x, y) - \phi_0]$$

where $\beta$ is a convection parameter and $\phi_0$ the temperature of the surrounding medium.

Construct a weak form for the problem.

For a finite element approximation to $\phi$ and $\delta\phi$ deduce the form of the new matrices which result from the modified weak form.

**5.3** For the quasi-harmonic equation consider a square four-node element with unit side lengths in the $x$- and $y$-directions. Using *FEAPpv* (or any other available

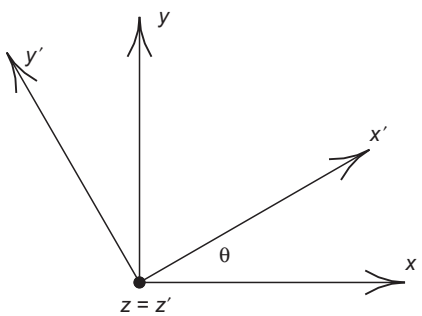



**FIGURE 5.17**
Orientation of axes for Problem 5.1.

program) determine the rank of the element matrix $\mathbf{H}$ for the case where $\mathbf{k} = \mathbf{I}$ (i.e., isotropic with $k = 1$) and $H = 0$ using $1 \times 1$ Gaussian quadrature. Repeat the calculation using $2 \times 2$ and $3 \times 3$ quadrature where each direction uses the values given in Table 3.3:

**(a)** What is the lowest order quadrature that gives a matrix $\mathbf{H}$ with full rank?
**(b)** What is the lowest order quadrature that evaluates the matrix $\mathbf{H}$ exactly?

Hint: The rank of $\mathbf{H}$ may be determined from the eigenproblem given by

$$\mathbf{H}\mathbf{v}_i = \lambda_i \mathbf{v}_i \qquad \text{with } \mathbf{v}_i^{\mathrm{T}} \mathbf{v}_j = \delta_{ij}$$

where $\delta_{ij}$ is the Kronecker delta. The rank of $\mathbf{H}$ is the number of nonzero eigenvalues $\lambda_i$ (a zero is any value below the round-off limit).

**5.4** Solve Problem 5.3 for a three-node triangular element mesh.
Using the eigenvector for the zero eigenvalue of the fully integrated element array $\mathbf{H}$ determine and sketch the shape of eigenvectors for the nonzero eigenvalues. (Note: The element has one zero eigenvalue, $\mathbf{v}_0$.)

**5.5** Consider the torsion of a rectangular bar by the warping function formulation discussed in Section 5.3.1. Let $a$ and $b$ be the side lengths in the $x$- and $y$-directions, respectively. For a homogeneous section with shear modulus $G$ the warping function has the behavior shown in Fig. 5.18 for $a/b$ ratios of 1, 1.25, and 2. Note that the behavior transitions from eight to four regions of $\pm$ variation. Estimate the $a/b$ ratio where this transition just occurs.
To make your estimate use *FEAPpv* (or any other available program) with a fine mesh of four-node rectangular elements. Set the boundary conditions to make the warping function zero along the $x$ and $y$ axes. The transition will occur at the smallest $a/b$ for which all the values on the perimeter of one quadrant of the cross-section have the same sign or are "numerically" zero.

**5.6** A cross-section of a long prismatic section is shown in Fig. 5.19 and subjected to a constant uniform temperatures 370 °C on the left boundary and 66 °C on the right boundary. The top and bottom edges are assumed to be insulated so that $q_n = 0$.

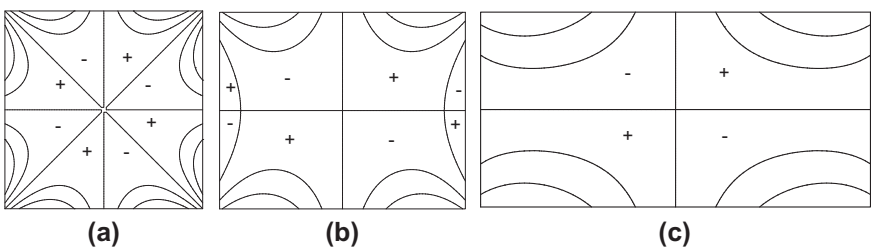



**FIGURE 5.18**
Warping function for torsion of rectangular bar for Problem 5.5: (a) $a/b = 1$; (b) $a/b = 1.25$; (c) $a/b = 2$.

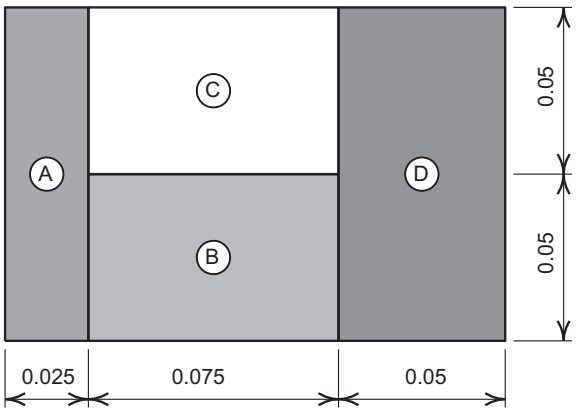



**FIGURE 5.19**
Thermal analysis of composite section for Problem 5.6.

The cross-section is a composite of fir (A), concrete (B), glass wool (C), and yellow pine (D). The thermal conductivity for each of the parts is $k_A = 0.11$, $k_B = 0.78$, $k_C = 0.04$, and $k_D = 0.147$ in consistent units for the geometry of the section shown:

**(a)** Estimate the heat flow through the cross-section assuming $q_y = 0$ and $q_x$ is constant in each part. Let the temperatures at each junction be $T(0) = 370$, $T(0.025) = T_1$, $T(0.10) = T_2$, and $T(0.15) = 66$.
Hint: Assume $T$ is a function of $x$ only.

**(b)** Use *FEAPpv* (or any other available program) to compute a finite element solution using four-node rectangular elements. First perform a solution on a coarse mesh and use this to design a mesh using a finer discretization.

Plot a distribution of heat flow $q_n$ across each of the internal boundaries.

**5.7** Company X&Y plans to produce a rectangular block which needs to be processed by a thermal quench in a medium which is 100 °C above room temperature. The block shown in Fig. 5.20a has $a = 10$ and $b = 20$ (i.e., the block is $10 \times 10 \times 20$). It has been determined that the thermal properties of the block may be specified by an isotropic Fourier model in which $k = 1$ and $c = 1$. The surface convection constant $H$ is 0.05.
The quench must be maintained until the minimum temperature in the block reaches 99 °C above room temperature. Use *FEAPpv* (or any other available program) to perform a transient analysis to estimate the required quench time:

**(a)** First perform a 2-D plane analysis on a $10 \times 10$ cross-section using a uniform mesh of four-node quadrilateral elements. Use symmetry to reduce the size of the domain analyzed. The surface convection will be modeled by two-node line elements along the outer perimeter. The analysis region is shown in Fig. 5.20b with the boundary conditions to be imposed. Locate

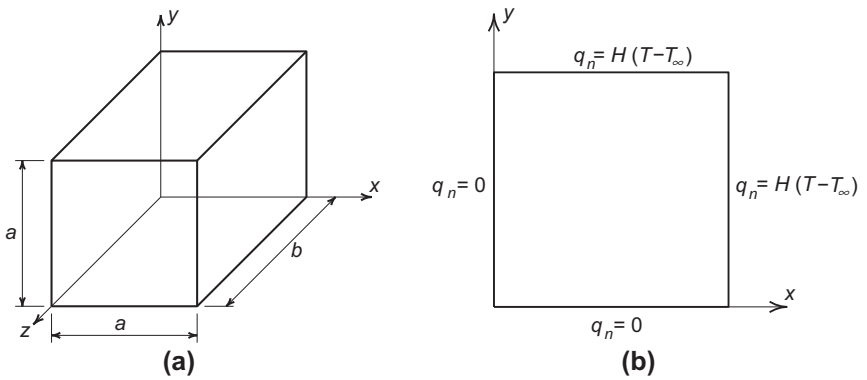



**FIGURE 5.20**
Thermal quench in two and three dimensions for Problem 5.7.

the node where the minimum temperature occurs and plot the behavior vs. time (a good option is to use MATLAB or GNU Octave to perform plots). Estimate the duration of time needed for the minimum temperature to reach the desired value. (Hint: One approach to selecting time increments is to select a very small value, e.g., $\Delta t = 10^{-8}$ and perform 10 steps of the solution. Multiply the time increment by 10 and perform 9 more steps. Repeat the multiplication until the desired final time value is reached.)

**(b)** Using the time duration estimated in (a) perform a 3-D analysis using a uniform mesh of eight-node hexahedral elements. Use symmetry to reduce the size of the region analyzed. Note: The convection condition applies to all outer surfaces.
Estimate the duration of quench time needed for the minimum temperature to reach the desired value.

**(c)** What analyses would you perform if the block was $10 \times 10 \times 5$?

**(d)** Comment on use of a 2-D analysis to estimate the required quench times for other part shapes.

**5.8** The distribution of shear stresses on the cross-section of a cantilever beam shown in Fig. 5.21a may be determined by solving the quasi-harmonic equation [46]

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0$$

with boundary condition

$$\phi = \frac{P}{2I}\left[\int y^2 \mathrm{d}x - \frac{\nu}{3(1+\nu)} y^3\right]$$

where $P$ is the end load, $I$ is the moment of inertia of the cross-section, $\nu$ is the Poisson ratio of an isotropic elastic material, and $\phi$ is a stress function. The

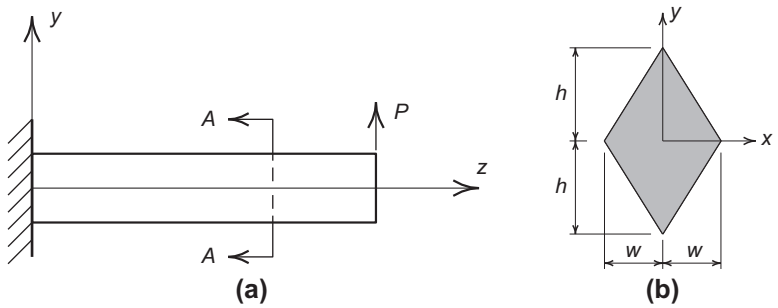



**FIGURE 5.21**
End-loaded cantilever beam for Problem 5.8: (a) cantilever beam and (b) section A-A.

shear stresses are determined from

$$\tau_{xz} = -\frac{\partial \phi}{\partial y} \quad \text{and} \quad \tau_{yz} = \frac{\partial \phi}{\partial x} + \frac{P}{2I}\left[\frac{\nu}{1+\nu}x^2 - y^2\right]$$

See Ref. [46] for details on the formulation.

**(a)** Show that the stress function satisfies the equilibrium equation when the bending stress is computed from

$$\sigma_z = -\frac{P(L-z)y}{I}$$

and $\sigma_x = \sigma_y = \tau_{xy} = 0$. $L$ is the length of the beam.

**(b)** Develop a weak form for the problem in terms of the stress function $\phi$.

**(c)** For a finite element formulation develop the relation to compute the boundary condition for the case when either three-node triangular or four-node rectangular elements are used.

**(d)** Write a program to determine the boundary values for the cross-section shown in Fig. 5.21b. Let $w = 2$ and $h = 3$. Use the quasi-harmonic thermal element in *FEAPpv* (or any other available program) to solve for the stress function $\phi$. Plot the distribution for $\phi$ on the cross-section.

**(e)** Modify the expressions in *FEAPpv* (or any other program for which source code is available) to compute the stress distribution on the cross-section. Solve and plot their distribution. Compare your results to those computed from the classical strength of materials approach.

Hint: Normalize your solution by the factor $P/2I$ to simplify expressions.

**5.9** A long sheet pile is placed in soil as shown in Fig. 5.22. The anisotropic properties of the soil are oriented so that $x = x'$ and $y = y'$. The governing differential equation is given in Section 5.3.3. The soil has the properties $k_x = 2$ and $k_y = 3$. Use *FEAPpv* (or any other available program) to determine the distribution of

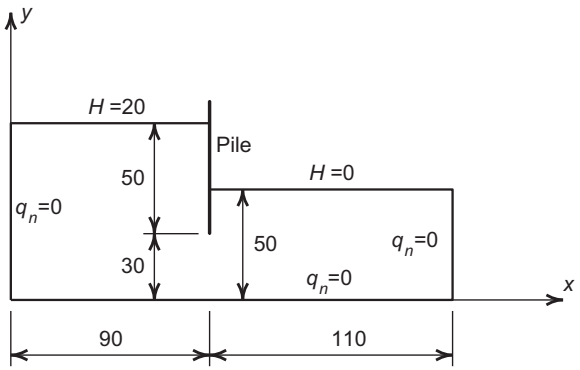



**FIGURE 5.22**
Seepage under a sheet pile for Problem 5.9.

head and the flow in the region shown. Solve the problem using a mesh of four-node, eight-node, and nine-node quadrilateral elements. Model the problem so that there are about four times as many four-node elements as used for the eight- and nine-node models (and thus approximately an equal number of nodes for each model). Compare total flow obtained from each analysis.

**5.10** An axisymmetric sheet pile is placed in soil as shown in Fig. 5.23. The anisotropic properties of the soil are oriented so that $r = r'$ and $z = z'$. The governing equation for plane flow is given in Section 5.3.3. Deduce the Euler differential equation for the axisymmetric problem from the weak form and definitions given in Section 5.1.2.1 suitably modified for the seepage problem.

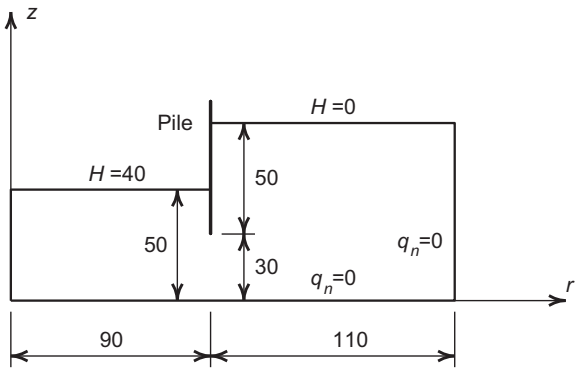



**FIGURE 5.23**
Seepage under an axisymmetric sheet pile for Problem 5.10.

Assuming isotropic properties with $k = 3$, use *FEAPpv* (or any other available program) to determine the distribution of head and the flow in the region shown. Solve the problem using a mesh of four-node, eight-node, and nine-node quadrilateral elements. Model the problem so that there are about four times as many four-node elements as used for the eight- and nine-node models (and thus approximately an equal number of nodes for each model). Compare total flow obtained from each analysis.

**5.11** A membrane occupies a region in the $x$-$y$ plane and is stretched by a uniform tension $T$. When subjected to a transient load $q(x, y, t)$ acting normal to the surface the governing differential equation is given by

$$-T\left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right] + m\frac{\partial^2 u}{\partial t^2} = q(x, y, t)$$

**(a)** Construct a weak form for the differential equation for the case when boundary conditions are given by $u(s, t) = 0$ for $s$ on $\Gamma$.

**(b)** Show that the solution by a finite element method may be constructed using $C_0$ functions.

**(c)** Approximate the $u$ and $\delta u$ by $C_0$ shape functions $N_a(x, y)$ and determine the semi-discrete form of the equations.

**(d)** For the case of steady harmonic motion, $u$ may be replaced by

$$u(x, y, t) = w(x, y) \exp i\omega t$$

where $i = \sqrt{-1}$ and $\omega$ is the frequency of excitation.
Using this approximation, deduce the governing equation for $w$. Construct a weak form for this equation. Using $C_0$ approximations for $w$ determine the form of the discretized problem.

**5.12** Program development project:[2] Write a MATLAB [47] or GNU Octave [48] program[3] to solve plane and axisymmetric quasi-harmonic problems.
Your program system should have the following features:

**(a)** Input module which describes:

**(i)** Nodal coordinate values, $\mathbf{x}_a$

**(ii)** Nodes connected to each element and material properties of the element

**(iii)** Node and degree-of-freedom (dof) for each applied nodal flux force (for problems in this chapter only one degree of freedom is used, however later we will use more)

[2] If programming is included as a part of your study, it is recommended that this problem be solved. Several extensions will be suggested later to create a solution system capable of performing additional steps of finite element analysis.

[3] Another programming language may be used, however, MATLAB and GNU Octave offer many advantages to write simple programs and are also useful to easily complete later exercises.

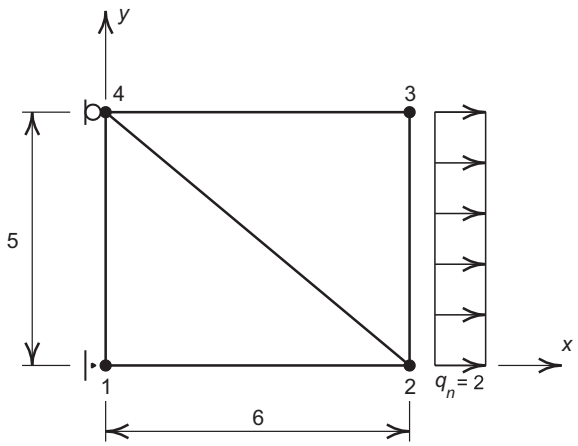



**FIGURE 5.24**
Patch test for triangles for Problem 5.12.

**(iv)** Node and dof for fixed (essential) boundary condition—also value if nonzero.

**(b)** Module to compute the **H** matrix for a three-node triangular element
**(c)** Module to assemble element arrays into global arrays and specified nodal forces and displacements
**(d)** Module to solve $\mathbf{H}\tilde{\boldsymbol{\phi}} = \mathbf{s}$
**(e)** Module to output nodal and element values.

Use your program to solve the *patch test* problem shown in Fig. 5.24. Use the properties $k = 200$ and $q_n = 1$. You can verify the correctness of your answer by computing an exact solution to the problem. The correctness of computed arrays may be obtained using results from *FEAPpv* (or any available quasi-harmonic program).

**5.13** Program development project: Add a graphics capability to the program developed in Problem 5.12 to plot contours of the computed finite element $\phi$. (Hint: MATLAB has `contour` and `surf` options to easily perform this operation.) Solve a two-dimensional plane problem of your choice and plot contours for $\phi$.

**5.14** Program development project: Extend the program developed in Problem 5.12 to solve transient problems.
Include an input module to specify the initial temperatures.
Also add a capability to consider time-dependent source terms for $Q$.
Test your program by solving the square cross-section problem described in Section 5.3.2.

## References


[1] O.C. Zienkiewicz, Y.K. Cheung, Finite elements in the solution of field problems, The Engineer, September 1965, pp. 507–510.

[2] W. Visser, A finite element method for the determination of non-stationary temperature distribution and thermal deformation, in: Proceedings of the First Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-66-80, Wright Patterson Air Force Base, Ohio, October 1966.

[3] O.C. Zienkiewicz, P. Mayer, Y.K. Cheung, Solution of anisotropic seepage problems by finite elements, J. Eng. Mech., ASCE 92 (1) (1966) 111–120.

[4] O.C. Zienkiewicz, P.L. Arlett, A.K. Bahrani, Solution of three-dimensional field problems by the finite element method, The Engineer, October 1967.

[5] L.R. Herrmann, Elastic torsion analysis of irregular shapes, J. Eng. Mech., ASCE 91 (6) (1965) 11–19.

[6] A.M. Winslow, Numerical solution of the quasi-linear Poisson equation in a non-uniform triangle 'mesh', J. Comput. Phys. 1 (1966) 149–172.

[7] S. Govindjee, J. Strain, T.J. Mitchell, R.L. Taylor, Convergence of an efficient local least-squares fitting method for bases with compact support, Comput. Methods Appl. Mech. Eng. 213–216 (2012) 84–92, doi: http://dx.doi.org/10.1016/j.cma.2011.11.017.

[8] R. Courant, Variational methods for the solution of problems of equilibrium and vibration, Bull. Am. Math. Soc. 49 (1943) 1–61.

[9] M.J. Turner, R.W. Clough, H.C. Martin, L.J. Topp, Stiffness and deflection analysis of complex structures, J. Aeronaut. Sci. 23 (1956) 805–823.

[10] D.N. de G. Allen, Relaxation Methods, McGraw-Hill, London, 1955.

[11] J.F. Ely, O.C. Zienkiewicz, Torsion of compound bars—a relaxation solution, Int. J. Mech. Sci. 1 (1960) 356–365.

[12] P. Silvester, M.V.K. Chari, Non-linear magnetic field analysis of DC machines, Trans. IEEE (7) (1970) 5–89.

[13] P. Silvester, M.S. Hsieh, Finite element solution of two dimensional exterior field problems, Proc. IEEE 118 (1971).

[14] B.H. McDonald, A. Wexler, Finite element solution of unbounded field problems, Proc. IEEE MTT-20 (12) (1972).

[15] E. Munro, Computer design of electron lenses by the finite element method, in: Image Processing and Computer Aided Design in Electron Optics, Academic Press, New York, 1973, p. 284.

[16] D.V. Tanesa, I.C. Rao, Student Project Report on Lubrication, Royal Naval College, 1966.

[17] M.M. Reddi, Finite element solution of the incompressible lubrication problem, Trans. Am. Soc. Mech. Eng. 91 (Ser. F) (1969) 524.

[18] M.M. Reddi, T.Y. Chu, Finite element solution of the steady state compressible lubrication problem, Trans. Am. Soc. Mech. Eng. 92 (Ser. F) (1970).

[19] J.H. Argyris, D.W. Scharpf, The incompressible lubrication problem, J. Roy. Aeronaut. Soc. 73 (1969) 1044–1046.
[20] J.F. Booker, K.H. Huebner, Application of finite element methods to lubrication: an engineering approach, J. Lubr. Technol., Trans. ASME 14 (Ser. F) (1972) 313.
[21] K.H. Huebner, Application of finite element methods to thermohydrodynamic lubrication, Int. J. Numer. Methods Eng. 8 (1974) 139–168.
[22] S.M. Rohde, K.P. Oh, Higher order finite element methods for the solution of compressible porous bearing problems, Int. J. Numer. Methods Eng. 9 (1975) 903–912.
[23] A.K. Tieu, Oil film temperature distributions in an infinitely wide glider bearing: an application of the finite element method, J. Mech. Eng. Sci. 15 (1973) 311.
[24] K.H. Huebner, Finite element analysis of fluid film lubrication—a survey, in: R.H. Gallagher, J.T. Oden, C. Taylor, O.C. Zienkiewicz (Eds.), Finite Elements in Fluids, vol. II, John Wiley & Sons, New York, 1975, pp. 225–254.
[25] A. Curnier, R.L. Taylor, A thermomechanical formulation and solution of lubricated contacts between deformable solids, J. Lubr. Technol., ASME 104 (1982) 109–117.
[26] H.C. Martin, Finite element analysis of fluid flows, in: Proceedings of the Second Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-68-150, Wright Patterson Air force Base, Ohio, October 1968.
[27] G. de Vries, D.H. Norrie, Application of the finite element technique to potential flow problems, Technical Report, Reports 7 and 8, Dept. Mech. Eng., University of Calgary, Alberta, Canada, 1969.
[28] J.H. Argyris, G. Mareczek, D.W. Scharpf, Two and three dimensional flow using finite elements, J. Roy. Aeronaut. Soc. 73 (1969) 961–964.
[29] L.J. Doctors, An application of finite element technique to boundary value problems of potential flow, Int. J. Numer. Methods Eng. 2 (1970) 243–252.
[30] G. de Vries, D.H. Norrie, The application of the finite element technique to potential flow problems, J. Appl. Mech., ASME 38 (1971) 798–802.
[31] S.T.K. Chan, B.E. Larock, L.R. Herrmann, Free surface ideal fluid flows by finite elements, J. Hydraul. Div., ASCE 99 (6) (1973).
[32] B.E. Larock, Jets from two dimensional symmetric nozzles of arbitrary shape, J. Fluid Mech. 37 (1969) 479–483.
[33] O.C. Zienkiewicz, R.L. Taylor, P. Nithiarasu, The Finite Element Method for Fluid Dynamics, Elsevier, Oxford, seventh ed., 2013.
[34] C.S. Desai, Finite element methods for flow in porous media, in: J.T. Oden, O.C. Zienkiewicz, R.H. Gallagher, C. Taylor (Eds.), Finite Elements in Fluids, vol. 1, John Wiley & Sons, New York, 1976, pp. 157–182.
[35] I. Javandel, P.A. Witherspoon, Applications of the finite element method to transient flow in porous media, Trans. Soc. Petrol. Eng. 243 (1968) 241–251.
[36] R.L. Taylor, C.B. Brown, Darcy flow solutions with a free surface, J. Hydraul. Div., ASCE 93 (2) (1967) 25–33.

[37] J.C. Luke, A variational principle for a fluid with a free surface, J. Fluid Mech. 27 (1957) 395–397.
[38] K. Washizu, Variational Methods in Elasticity and Plasticity, Pergamon Press, New York, third ed., 1982.
[39] J.C. Bruch, A survey of free-boundary value problems in the theory of fluid flow through porous media, Adv. Water Resour. 3 (1980) 65–80.
[40] C. Baiocchi, V. Comincioli, V. Maione, Unconfined flow through porous media, Meccanice, Ital. Ass. Theor. Appl. Mech. 10 (1975) 51–60.
[41] J.M. Sloss, J.C. Bruch, Free surface seepage problem, J. Eng. Mech., ASCE 108 (5) (1978) 1099–1111.
[42] N. Kikuchi, Seepage flow problems by variational inequalities, Int. J. Numer. Anal. Methods Geomech. 1 (1977) 283–290.
[43] C.S. Desai, Finite element residual schemes for unconfined flow, Int. J. Numer. Methods Eng. 10 (1976) 1415–1418.
[44] C.S. Desai, G.C. Li, A residual flow procedure and application for free surface, and porous media, Adv. Water Resour. 6 (1983) 27–40.
[45] K.J. Bathe, M. Koshgoftar, Finite elements from surface seepage analysis without mesh iteration, Int. J. Numer. Anal. Methods Geomech. 3 (1979) 13–22.
[46] S.P. Timoshenko, J.N. Goodier, Theory of Elasticity, McGraw-Hill, New York, second ed., 1951.
[47] MATLAB, 2012. <www.mathworks.com>.
[48] GNU Octave, 2012. <www.gnu.org/software/octave>.
[49] O.C. Zienkiewicz, R.L Taylor, J.Z. Zhu, The Finite Element Method: Its Basis & Fundamentals, Elsevier, Oxford, sixth ed., 2005.

This page is intentionally left blank

CHAPTER

# Shape Functions, Derivatives, and Integration 6

## 6.1 Introduction

In the previous chapter we found that the quasi-harmonic equation created a weak form that contained derivatives of the dependent variable up to first order. Indeed we will find that there is a large class of problems which have equations which have no more than second derivatives of the dependent variables and thus their weak form will contain no more than first derivatives of the dependent variable and arbitrary function. All such forms will require the use of $C_0$ shape functions for the approximate solution by a finite element method. Thus, in this chapter we present the development of shape functions for some general families of $C_0$ interpolations. Both two- and three-dimensional forms are considered. With the general families available we are then able to consider their use to solve problems in elasticity, plates, and shells. To this end, we note that these functions must satisfy the following conditions:

1. Be such that continuity is achieved at each point on the interface between adjacent elements.
2. Include complete linear polynomials such that constant derivatives may be obtained in each element.

These two conditions are quite easy to achieve for two-dimensional elements that have triangular or quadrilateral shape and for elements with three-dimensional tetrahedral, wedge, or hexahedral forms.

The shape functions presented below are described by local nodes with each local node associated to a specific global node during the assembly process as described in previous chapters.

## 6.2 Two-dimensional shape functions

### 6.2.1 Shape functions for triangles

#### *6.2.1.1 Triangle with three nodes*

As an example let us consider again the two-dimensional plane problem for which a simple set of $C_0$ functions for a parameter $\phi$ may be constructed from linear polynomials over three-node triangles as shown in Fig. 6.1a. The three shape functions for

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00006-X

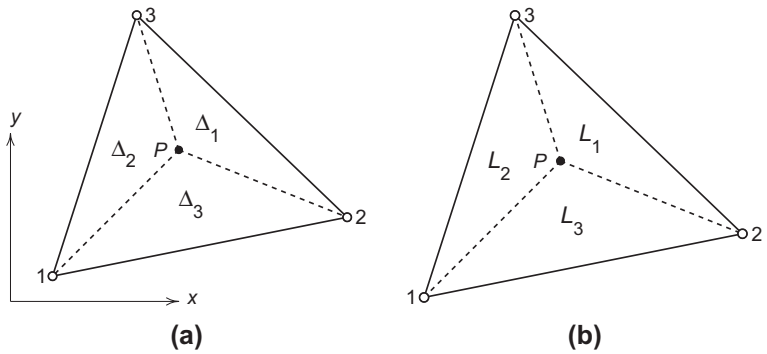



**FIGURE 6.1**
Triangular element with area and baricentric parent coordinates: (a) subareas of triangle and (b) parent coordinates.

the triangle with nodal coordinates $x_a$, $y_a$, developed in Section 5.1.2, are given by

$$N_a(x, y) = \frac{1}{2\Delta}\left(a_a + b_a\, x + c_a\, y\right), \quad a = 1, 2, 3$$

where

$$\begin{aligned} a_1 &= x_2y_3 - x_3y_2, \quad b_1 = y_2 - y_3, \quad c_1 = x_3 - x_2 \\ a_2 &= x_3y_1 - x_1y_3, \quad b_2 = y_3 - y_1, \quad c_2 = x_1 - x_3 \\ a_3 &= x_1y_2 - x_2y_1, \quad b_3 = y_1 - y_2, \quad c_3 = x_2 - x_1 \end{aligned}$$

and $\Delta = (x_1b_1 + x_2b_2 + x_3b_3)/2$ is the area of the triangle.

Geometrically, the numerator of the shape function is twice the area for the triangle composed of point $(x,\ y)$ and two of the nodal coordinates $x_a,\ y_a$. Thus, as shown in Fig. 6.1a, we can define

$$2\,\Delta_a(x, y) = a_a + b_a\, x + c_a\, y \tag{6.1}$$

and the shape functions become

$$N_a = \frac{\Delta_a(x, y)}{\Delta} \tag{6.2}$$

This ratio also defines a convenient parent coordinate system (Fig. 6.1b) for triangles which is known as *baricentric* or *area* coordinates and was first introduced by the German mathematician August Möbius in 1827 [1]. We shall denote these as

$$L_a = \frac{\Delta_a}{\Delta}, \quad a = 1, 2, 3 \tag{6.3}$$

Since there are three parent coordinates it is necessary to always note that they are constrained by the total area such that

$$L_1 + L_2 + L_3 = 1 \tag{6.4}$$

Using area coordinates the shape functions for the three-node triangle are given by

$$N_a = L_a, \quad a = 1, 2, 3 \tag{6.5}$$

### 6.2.1.2 *Higher order triangular elements*

The process originally used for the three-node triangle may be performed for triangular elements with more than three nodes. The first three members of the triangular element family are shown in Fig. 6.2. It is useful to note that a complete set of polynomials in two dimensions may be represented on a Pascal triangle as shown in Fig. 6.3. Using a Pascal triangle we note that each complete order of the polynomials exactly matches

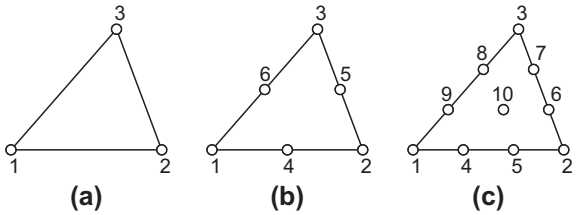



**FIGURE 6.2**
Triangular element family: (a) linear, (b) quadratic, and (c) cubic.

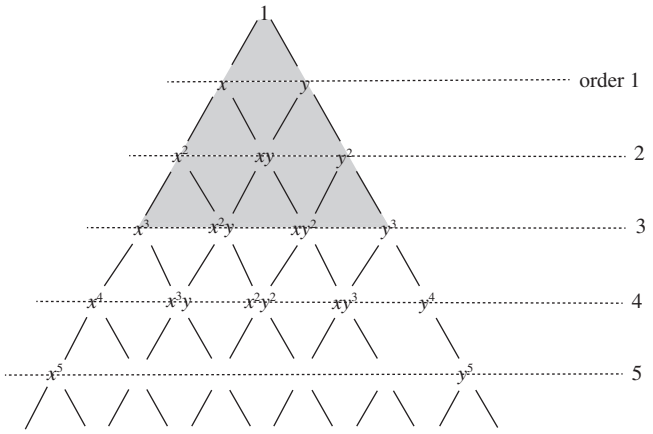



**FIGURE 6.3**
The Pascal triangle (cubic expansion shaded—10 terms).

the number of nodes of the triangle with the same order. Thus, the shape functions for a six-node triangle may be obtained using quadratic order polynomials as

$$\hat{\phi}^e = \alpha_1 + x\,\alpha_2 + y\,\alpha_3 + x^2\,\alpha_4 + xy\,\alpha_5 + y^2\,\alpha_6 \tag{6.6}$$

Pursuing this approach will require the inverse of a $6\times6$ matrix to obtain the expression for the parameters $\alpha_a$ in terms of the nodal parameters $\tilde{\phi}_a^e$. Furthermore, each triangle so constructed must have straight edges in order to satisfy the continuity condition between elements. Thus, we shall find it useful to develop the shape functions directly using area coordinates and, thus, eliminate the need to invert matrices.

To derive shape functions for higher order elements a simple recurrence relation can be derived [2]. However, it is very simple to write an arbitrary triangle of order $M$ in a direct manner.

Denoting a typical node $a$ by three numbers $I$, $J$, and $K$ corresponding to the position of coordinates $L_{1a}$, $L_{2a}$, and $L_{3a}$, we can write the shape function in terms of three Lagrangian interpolations as [see Eq. (3.39)]

$$N_a = l_I^I(L_1)l_J^J(L_2)l_K^K(L_3) \quad \text{with } I + J + K = M \tag{6.7}$$

In the above $l_I^I, l_J^J, l_K^K$ are given by expression (3.39), with $L_1, L_2, L_3$ taking the place of $\xi$.

It is easy to verify that the above expression gives

$$N_a = 1 \qquad \text{at } L_1 = L_{1I}, \quad L_2 = L_{2J}, \quad L_3 = L_{3K}$$

and zero at all other nodes. The highest term occurring in the expansion is $L_1^I L_2^J L_3^K$ and as $I + J + K \equiv M$ for all points the polynomial is also of order $M$.

Expression (6.7) is valid for quite arbitrary distributions of nodes of the pattern given in Fig. 6.4 and simplifies if the spacing of the nodal lines is equal (i.e., $1/m$). The formula was first obtained by Argyris et al. [3] and formalized in a different manner by others [4,5].

The reader can verify the shape functions for the second- and third-order elements as given below and indeed derive ones of any higher order easily.

#### 6.2.1.3 *Quadratic triangle (Fig. 6.2b)*

Corner nodes:

$$N_a = (2L_a - 1)L_a, \quad a = 1, 2, 3$$

Mid-side nodes:

$$N_4 = 4L_1L_2, \quad N_5 = 4L_2L_3, \quad N_6 = 4L_3L_1$$

#### 6.2.1.4 *Cubic triangle (Fig. 6.2c)*

Corner nodes:

$$N_a = \frac{1}{2}(3L_a - 1)(3L_a - 2)L_a, \quad a = 1, 2, 3$$

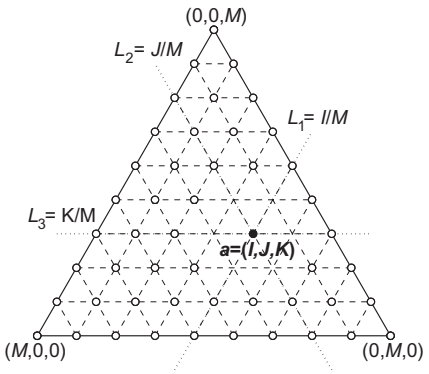



**FIGURE 6.4**
A general triangular element.

Mid-side nodes:

$$N_4 = \frac{9}{2}L_1L_2(3L_1 - 1), \quad N_5 = \frac{9}{2}L_1L_2(3L_2 - 1), \quad \text{etc.}$$

The internal node:

$$N_{10} = 27L_1L_2L_3$$

The last shape again is a "bubble" function giving zero contribution along boundaries—this will be found to be useful in other contexts (see the mixed forms in Chapter 10).

The quadratic triangle was first derived by de Veubeke [6] and used later in the context of plane stress analysis by Argyris [7].

## 6.2.2 Shape functions for quadrilaterals

### 6.2.2.1 *Quadrilateral with four nodes*

In Section 5.1.2 the shape functions for the rectangular element shown in Fig. 6.5a were obtained as

$$N_1 = \frac{1}{4}\left(1 - \frac{x'}{a}\right)\left(1 - \frac{y'}{b}\right), \quad N_2 = \frac{1}{4}\left(1 + \frac{x'}{a}\right)\left(1 - \frac{y'}{b}\right)$$
$$N_3 = \frac{1}{4}\left(1 + \frac{x'}{a}\right)\left(1 + \frac{y'}{b}\right), \quad N_4 = \frac{1}{4}\left(1 - \frac{x'}{a}\right)\left(1 + \frac{y'}{b}\right)$$

If we let

$$\xi = \frac{x'}{a} \quad \text{and} \quad \eta = \frac{y'}{b}$$

the shape functions on a parent element (viz. Fig. 6.5b) may be written as

$$\begin{aligned} N_1 &= \frac{1}{2}(1-\xi)\cdot\frac{1}{2}(1-\eta), \quad N_2 = \frac{1}{2}(1+\xi)\cdot\frac{1}{2}(1-\eta) \\ N_3 &= \frac{1}{2}(1+\xi)\cdot\frac{1}{2}(1+\eta), \quad N_4 = \frac{1}{2}(1-\xi)\cdot\frac{1}{2}(1+\eta) \end{aligned} \tag{6.8}$$

which we recognize as products of one-dimensional Lagrange interpolations using the parent coordinates $-1 \le \xi,\ \eta \le 1$.

#### 6.2.2.2 *Lagrangian family of quadrilaterals*

Based on the above a systematic and easy method to generate shape functions for any order of rectangle can be achieved by a simple product of Lagrange polynomials in the two parent coordinates [8–10]. Thus, in two dimensions, if we label the node by its column and row number, $I$, $J$, using the definition of Lagrange polynomials given in (3.39) we have

$$N_a \equiv N_{IJ} = l_I^n(\xi) l_J^m(\eta) \tag{6.9}$$

where $n$ and $m$ stand for the number of subdivisions in each direction. Table 6.1 shows how the $I,\ J$ values are mapped to the node numbers $a$ shown in Fig. 6.5. Figure 6.6 shows a few members of this unlimited family where $m = n$. For $m = n = 1$ we obtain the simple result

$$N_a = \frac{1}{4}(1+\xi_a\xi)(1+\eta_a\eta)$$

in which $\xi_a$, $\eta_a$ are the normalized coordinates at node $a$.

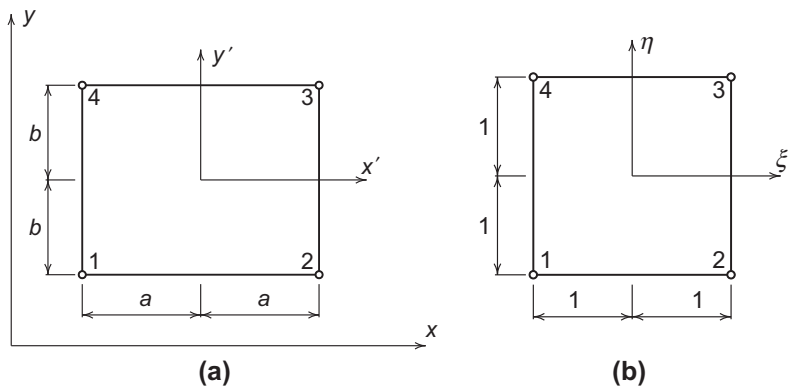



**FIGURE 6.5**
Rectangular element geometry and local node numbers: (a) global coordinates and (b) parent coordinates.

**Table 6.1** Numbering for Four-Node Rectangle

| Label | Node Numbers | | | |
|---|---|---|---|---|
| a | 1 | 2 | 3 | 4 |
| I | 1 | 2 | 2 | 1 |
| J | 1 | 1 | 2 | 2 |

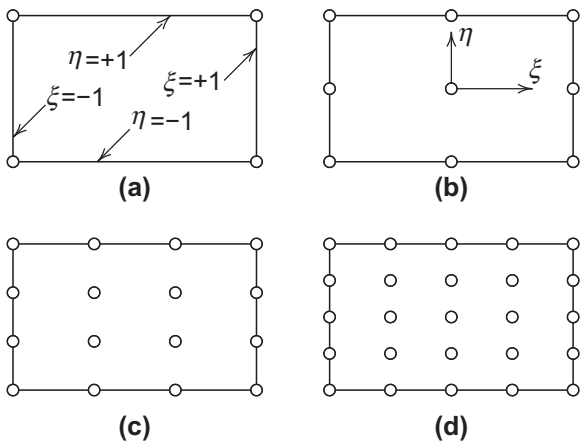



**FIGURE 6.6**
Four elements of the Lagrangian family: (a) linear, (b) quadratic, (c) cubic, and (d) quartic.

#### 6.2.2.3 *"Quadratic" element (Fig. 6.6b)*

Products of one-dimensional quadratic Lagrange interpolations yield the following:

Corner nodes:

$$N_a = \frac{1}{4}\xi\eta(\xi + \xi_a)(\eta + \eta_a)$$

Mid-side nodes:

$$\xi_a = 0, \quad N_a = \frac{1}{2}\eta(1 - \xi^2)(\eta + \eta_a)$$

$$\eta_a = 0, \quad N_a = \frac{1}{2}\xi(\xi + \xi_a)(1 - \eta^2)$$

Center node:

$$N_a = (1 - \xi^2)(1 - \eta^2)$$

Indeed, if we examine the polynomial terms present in a situation where $n = m$ we observe in Fig. 6.7, based on the Pascal triangle, that a large number of polynomial

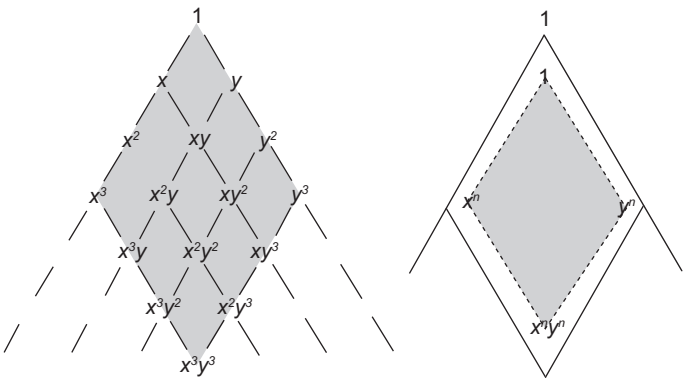



**FIGURE 6.7**
Terms generated by a Lagrangian expansion of order $3 \times 3$ [or $m \times n$ using Eq. (6.9)]. The shaded region shows complete polynomials of order 3 and $n$.

terms is present above those needed for a complete expansion [4]. However, when mapping of shape functions to more general shapes is considered (viz. Section 6.5) some advantages occur for this family.

#### 6.2.2.4 *Serendipity family of quadrilaterals*

For some classes of problems it is more efficient to make the functions dependent on nodal values placed mostly on the element boundary. Consider, for instance, the first four elements of Fig. 6.8. In each a progressively increasing and equal number of

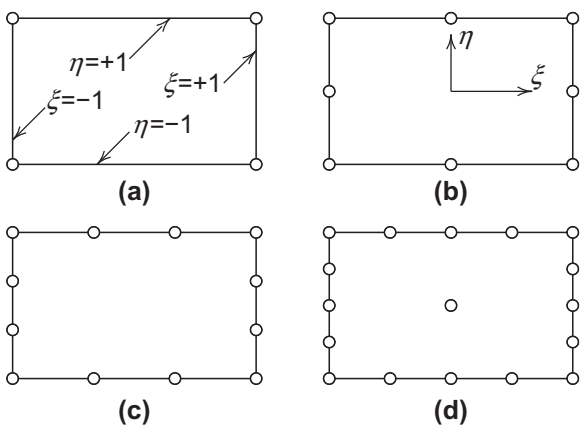



**FIGURE 6.8**
Four elements of the serendipity family: (a) linear, (b) quadratic, (c) cubic, and (d) quartic.

nodes is placed on the element boundary. The variation of the function on the edges to ensure continuity is linear, parabolic, and cubic in increasing element order.

To achieve the shape function for the first element it is obvious that a product of linear Lagrangian polynomials may again be used. Indeed this element is identical to the Lagrangian one with $n = 1$ and again all the shape functions may be written as one expression:

$$N_a = \frac{1}{4}(1 + \xi_a\xi)(1 + \eta_a\eta)$$

The reader can verify that the following functions satisfy all the necessary criteria for quadratic and cubic members of the family.

#### 6.2.2.5 "Quadratic" element (Fig. 6.8b)

Corner nodes:

$$N_a = \frac{1}{4}(1 + \xi_a\xi)(1 + \eta_a\eta)(\xi_a\xi + \eta_a\eta - 1) \tag{6.10a}$$

Mid-side nodes:

$$\begin{aligned} \xi_a = 0, \quad N_a &= \frac{1}{2}(1 - \xi^2)(1 + \eta_a\eta) \\ \eta_a = 0, \quad N_a &= \frac{1}{2}(1 + \xi_a\xi)(1 - \eta^2) \end{aligned} \tag{6.10b}$$

#### 6.2.2.6 "Cubic" element (Fig. 6.8c)

Corner nodes:

$$N_a = \frac{1}{32}(1 + \xi_a\xi)(1 + \eta_a\eta)[9(\xi^2 + \eta^2) - 10] \tag{6.11a}$$

Mid-side nodes:

$$\begin{aligned} \xi_a &= \pm 1 \quad \text{and} \quad \eta_a = \pm\frac{1}{3} \\ N_a &= \frac{9}{32}(1 + \xi_a\xi)(1 - \eta^2)(1 + 9\eta_a\eta) \end{aligned} \tag{6.11b}$$

and

$$\begin{aligned} \xi_a &= \pm\frac{1}{3} \quad \text{and} \quad \eta_a = \pm 1 \\ N_a &= \frac{9}{32}(1 - \xi^2)(1 + 9\xi_a\xi)(1 + \eta_a\eta) \end{aligned} \tag{6.11c}$$

which all satisfy the requirement

$$N_a(\xi_b, \eta_b) = \delta_{ab} = \begin{cases} 1, & a = b \\ 0, & a \neq b \end{cases} \tag{6.11d}$$

The above functions were originally derived by inspection, and progression to yet higher members is difficult and requires some ingenuity [8,9]. It was therefore appropriate to name this family "serendipity" after the famous princes of Serendip noted for their chance discoveries (Horace Walpole, 1754).

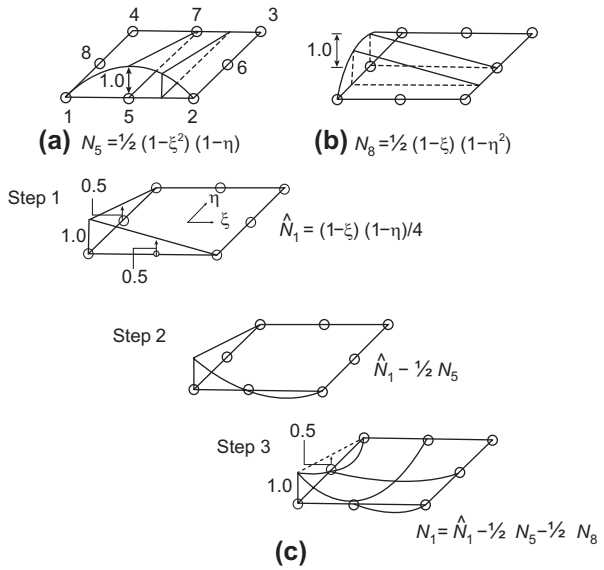



**FIGURE 6.9**
Systematic generation of "serendipity" shape functions.

However, a quite systematic way of generating the "serendipity" shape functions can be devised, which becomes apparent from Fig. 6.9 where the generation of a quadratic shape function is presented [4,11].

As a starting point we observe that for *mid-side* nodes a Lagrangian interpolation of a quadratic $\times$ linear type suffices to determine $N_a$ at nodes 5 to 8. $N_5$ to $N_8$ are shown in Fig. 6.9a and b. For a *corner* node, such as Fig. 6.9c, we start with a bilinear Lagrangian family $\hat{N}_1$ and note immediately that while $\hat{N}_1 = 1$ at node 1, it is not zero at nodes 5 or 8 (step 1). Successive subtraction of $1/2\, N_5$ (step 2) and $1/2\, N_8$ (step 3) ensures that a zero value is obtained at these nodes. The reader can verify that the final expressions obtained coincide with those of Eqs. (6.10a) and (6.10b).

Indeed, it should now be obvious that for all higher order elements the *mid-side* and *corner shape* functions can be generated by an identical process. For the former a simple multiplication of $m$th order and first-order Lagrangian interpolations suffices. For the latter a combination of bilinear corner functions, together with appropriate fractions of mid-side shape functions to ensure zero at appropriate nodes, is necessary.

It is also quite easy to generate shape functions for elements with different numbers of nodes along each side by a similar systematic algorithm. This may be very desirable if a transition between elements of different order is to be achieved, enabling a different order of accuracy in separate sections of a large problem to be studied. Figure 6.10 illustrates the necessary shape functions for a cubic/linear transition. Use of such special elements was first introduced in Ref. [11], but the simpler formulation used here is that of Ref. [4].

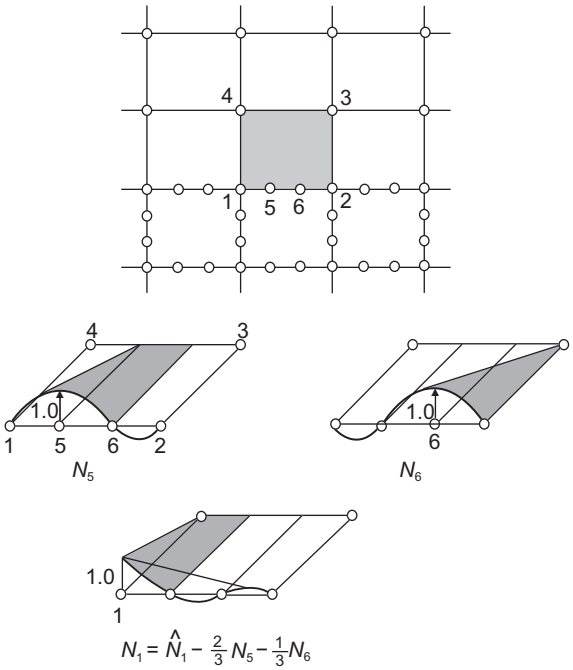



**FIGURE 6.10**
Shape functions for a transition "serendipity" element, cubic/linear.

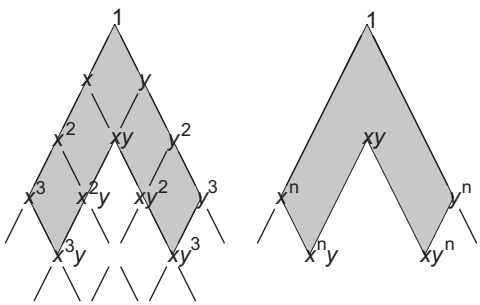



**FIGURE 6.11**
Terms generated by edge-shaped functions in serendipity-type elements ($3 \times 3$ and $n \times n$).

With the mode of generating shape functions for this class of elements available it is immediately obvious that fewer degrees of freedom are now necessary for a given complete polynomial expansion. Figure 6.11 shows this for a cubic element where only two surplus terms arise (as compared with six surplus terms in a Lagrangian of the same degree). However, when mapping to general quadrilateral shape is introduced in Section 6.5 some of these advantages are lost, rendering the Lagrangian form of interpolation more advantageous.

It is immediately evident, however, that the functions generated by nodes placed only along the edges will not generate complete polynomials beyond cubic order. For higher order ones it is necessary to supplement the expansion by internal nodes or by the use of "node-less" variables which contain appropriate polynomial terms. For example in the next quartic member [12] of this family a central node is added [viz. Fig. 6.8d] so that all terms of a complete fourth-order expansion will be available. This central node adds a shape function $(1-\xi^2)(1-\eta^2)$ which is zero on all outer boundaries and coincides with the internal function used in the quadratic Lagrangian element. Once interior nodes are added it is necessary to modify the corner and mid-side shape functions to preserve the Kronecker delta property (6.11d). However, one may also consider the interior nodes to be hierarchical (viz. Section 3.6).

The reader can verify that at element edges all the above two-dimensional shape functions are expressed entirely by nodal parameters on the edge. Thus, matching values at nodes of contiguous edges ensures the required $C_0$ continuity.

## 6.3 Three-dimensional shape functions

For interelement continuity the simple rules given previously for edges have to be modified. What is necessary to achieve such continuity is that *along a whole face of an element the nodal values define a unique variation of the unknown function*. In all the elements described below each face will be a two-dimensional interpolation of triangular or quadrilateral shape. Moreover, the face will be described by nodal parameters on the face. Thus, if the nodes on faces of contiguous elements match we are assured that $C_0$ compatibility is ensured over the whole face.

### 6.3.1 Tetrahedral elements

The tetrahedral family shown in Fig. 6.12 not surprisingly exhibits properties similar to those of the triangle family as deduced in the previous chapter. Firstly, once again complete polynomials in three coordinates are achieved at each stage. Secondly, as faces are divided in a manner identical with that of the previous triangles, the same order of polynomial in two coordinates in the plane of the face is achieved and element compatibility ensured. Finally, no surplus terms in the polynomial occur.

#### 6.3.1.1 Tetrahedron with four nodes

For three-dimensional problems a simple tetrahedron-shaped element was developed in Section 5.1.3 with shape functions given in (5.25) by

$$N_a(x,y,z) = \frac{1}{6V}\left(a_a + b_a\, x + c_a\, y + d_a\, z\right), \quad a = 1,2,3,4$$

The above form permits the generalization of area coordinates to *volume coordinates* expressed by $L_1, L_2, L_3, L_4$, where

$$L_a = \frac{V_a}{V}, \quad a = 1,2,3,4$$

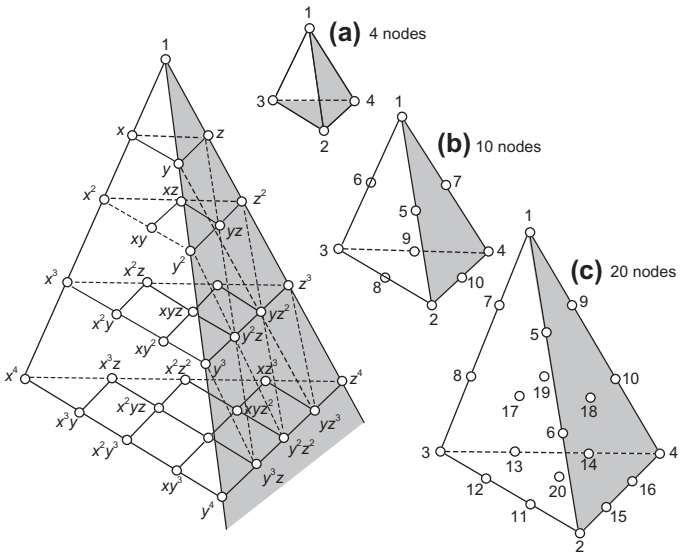



**FIGURE 6.12**
The tetrahedral family: (a) linear, (b) quadratic, and (c) cubic.

In the above $V$ is the volume of the tetrahedron with vertices 1234 and $V_1$, $V_2$, $V_3$, $V_4$ are volumes of the tetrahedron $432P, 413P, 421P, 123P$, respectively, that are expressed by

$$6V_a = a_a + b_a\, x + c_a\, y + d_a\, z$$

The subvolume $V_1$ is shown shaded in Fig. 6.13b. The volume coordinates must satisfy the constraint

$$\sum_{i=1}^{4} L_i = L_1 + L_2 + L_3 + L_4 = 1 \tag{6.12}$$

The shape functions for a four-node tetrahedron in terms of volume coordinates are given by

$$N_a = L_a, \quad a = 1, 2, 3, 4 \tag{6.13}$$

This form will be used later in a parametric mapping approach as introduced for the one-dimensional problem in Section 3.5.

To show $C_0$ continuity between contiguous tetrahedra each face must be described in terms its degrees of freedom. This is trivial to show for the four-node tetrahedron since on each face one of the volume coordinates is zero. For example on the face 123 in Fig. 6.13a the value of $L_4$ is zero and an interpolation is given by

$$\hat{\phi}^e = N_1(L_1)\,\tilde{\phi}_1^e + N_2(L_1)\,\tilde{\phi}_2^e + N_3(L_1)\,\tilde{\phi}_3^e$$

which is the interpolation in area coordinates for the face. Since this is a linear function it is obvious that continuity is preserved with any adjacent element.

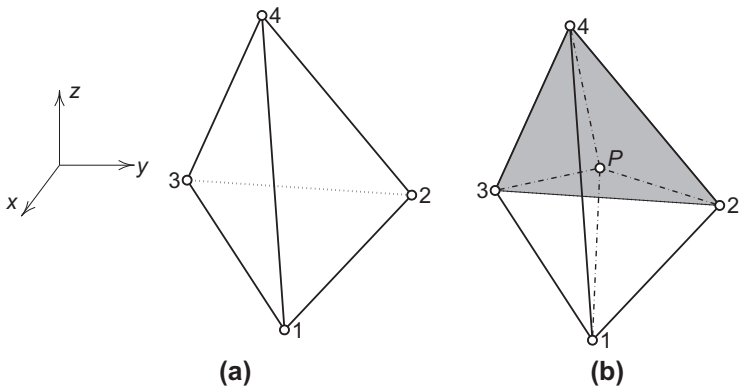



**FIGURE 6.13**
Tetrahedron element. Volume and baricentric coordinates: (a) four-node tetrahedron and (b) $V_1$ for $432P$.

#### 6.3.1.2 *Higher order tetrahedrons*

Formulae for shape functions of higher order tetrahedra are derived in precisely the same manner as for the triangles by establishing appropriate Lagrange-type formulae similar to Eq. (6.7). The result is

$$N_a = l_I^I(L_1) l_J^J(L_2) l_K^K(L_3) l_L^L(L_4) \quad \text{with } I + J + K + L = M \tag{6.14}$$

where $M$ is the order of the tetrahedron.

The reader may verify the following shape functions for the quadratic and cubic order cases.

#### 6.3.1.3 *"Quadratic" tetrahedron (Fig. 6.12b)*

Corner nodes:

$$N_a = L_a(2L_a - 1), \quad a = 1, 2, 3, 4$$

Mid-edge nodes:

$$N_5 = 4L_2L_1, \quad N_6 = 4L_3L_1, \quad N_7 = 4L_4L_1$$
$$N_8 = 4L_2L_3, \quad N_9 = 4L_3L_4, \quad N_{10} = 4L_4L_2$$

#### 6.3.1.4 *"Cubic" tetrahedron (Fig. 6.12c)*

Corner nodes:

$$N_a = \frac{1}{2} L_a(3L_a - 1)(3L_a - 2), \quad a = 1, 2, 3, 4$$

Mid-edge nodes:

$$N_5 = \frac{9}{2} L_1L_2(3L_1 - 1), \quad N_6 = \frac{9}{2} L_1L_2(3L_2 - 1), \quad \text{etc.}$$

Mid-face nodes:

$$N_{17} = 27L_1L_2L_3, \quad \text{etc.}$$

## 6.3.2 Hexagon elements: Brick family

### 6.3.2.1 Hexagon with eight nodes

For a three-dimensional problem we consider the rectangular brick shown in Fig. 6.14. The development of shape functions follows the procedure used for the four-node quadrilateral in Section 6.2.2.1. Introducing the coordinates $x'$, $y'$, $z'$ we may write a polynomial expression for $\phi$ as

$$\hat{\phi}^e = \begin{bmatrix} 1 & x & y & z & xy & yz & zx & xyz \end{bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \\ \alpha_7 \\ \alpha_8 \end{Bmatrix} \tag{6.15}$$

Introducing the nondimensional (parent coordinates)

$$\xi = \frac{x'}{a}, \quad \eta = \frac{y'}{b}, \quad \text{and} \quad \zeta = \frac{z'}{c} \tag{6.16}$$

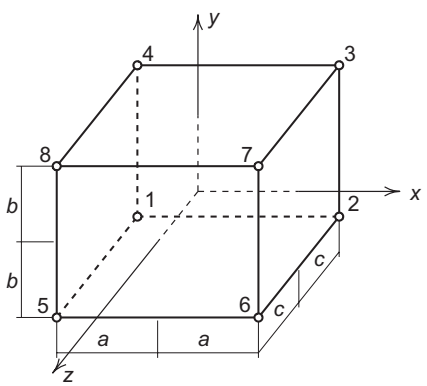



**FIGURE 6.14**
Brick element geometry and local node numbers.

the coefficients $\alpha_a$ may be obtained in the identical manner used above for the two-dimensional rectangle. The reader can verify that the result is

$$\begin{aligned} N_1 &= \frac{1}{8}(1-\xi)(1-\eta)(1-\zeta), \quad N_5 = \frac{1}{8}(1-\xi)(1-\eta)(1+\zeta) \\ N_2 &= \frac{1}{8}(1+\xi)(1-\eta)(1-\zeta), \quad N_6 = \frac{1}{8}(1+\xi)(1-\eta)(1+\zeta) \\ N_3 &= \frac{1}{8}(1+\xi)(1+\eta)(1-\zeta), \quad N_7 = \frac{1}{8}(1+\xi)(1+\eta)(1+\zeta) \\ N_4 &= \frac{1}{8}(1-\xi)(1+\eta)(1-\zeta), \quad N_8 = \frac{1}{8}(1-\xi)(1+\eta)(1+\zeta) \end{aligned} \tag{6.17}$$

which we may recognize again as products of linear Lagrange interpolations in the parent coordinates $\xi, \eta, \zeta$.

#### 6.3.2.2 Brick elements: Lagrangian family

In a precisely analogous way to that given in Section 6.2.2.2 equivalent Lagrangian family elements of the three-dimensional brick type can be described. Shape functions for such elements will be generated by a direct product of three Lagrange polynomials. Extending the notation of Eq. (6.9) we now have the interpolations in the parent element given by

$$N_a \equiv N_{IJK} = l_I^n(\xi) l_J^m(\eta) l_K^p(\zeta) \tag{6.18}$$

for $n, m$, and $p$ subdivisions along each side and with

$$-1 \le \xi,\ \eta,\ \zeta \le 1$$

This element again is suggested by Zienkiewicz et al. [9] and elaborated upon by Argyris et al. [10]. All the remarks about internal nodes are applicable here. The first three of the three-dimensional Lagrangian family are shown in Fig. 6.15a.

#### 6.3.2.3 "Linear" element (eight nodes) (Fig. 6.15a)

The linear element has shape functions defined by the product of linear Lagrangian interpolations and is given by

$$N_a(\xi, \eta, \zeta) = \frac{1}{8}(1+\xi_a\xi)(1+\eta_a\eta)(1+\zeta_a\zeta)$$

If we order the nodal numbers with the $\xi_a, \eta_a, \zeta_a$ as defined in Table 6.2, on face $\zeta = 1$ the shape functions for nodes 1 to 4 are zero and, thus, the interpolation on the face is given by

$$\hat{\phi}^e = \sum_{a=5}^{8} N_a(\xi, \eta, 1)\, \tilde{\phi}_a^e \equiv \sum_{a=5}^{8} N_a(\xi, \eta)\, \tilde{\phi}_a^e$$

The three-dimensional shape functions $N(\xi, \eta, 1)$ are precisely the two-dimensional shape functions $N_a(\xi, \eta)$. Thus, we are assured that matching the face with a face of the same shape and global node numbers on an adjacent element will assure $C_0$ continuity.

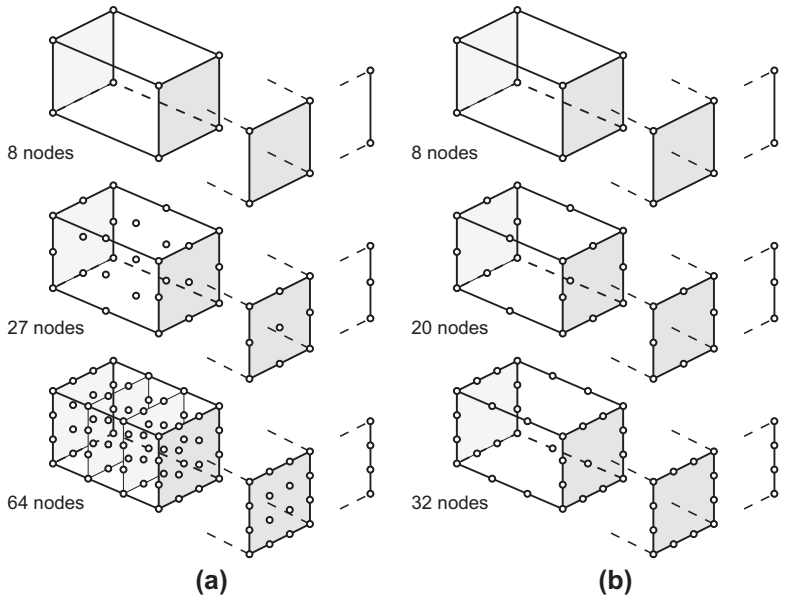



**FIGURE 6.15**
Linear, quadratic, and cubic right prisms with corresponding sheet and line elements (extra shading on 64-node element to show node location more clearly): (a) Lagrangian family and (b) serendipity family.

**Table 6.2** Numbering for Eight-Node Brick

| Label | Node Numbers | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **a** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| $\xi_a$ | 1 | 2 | 2 | 1 | 1 | 2 | 2 | 1 |
| $\eta_a$ | 1 | 1 | 2 | 2 | 1 | 1 | 2 | 2 |
| $\zeta_a$ | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |

#### 6.3.2.4 *Brick elements: "Serendipity" family*

The serendipity family of elements shown in Fig. 6.15b is precisely equivalent to that of Fig. 6.8 for the two-dimensional case [8,11,13]. Using now three coordinates and otherwise following the terminology used for quadrilaterals, we have the following shape functions.

#### 6.3.2.5 *"Linear" element (eight nodes) (Fig. 6.15b)*

$$N_a = \frac{1}{8}(1+\xi_a\xi)(1+\eta_a\eta)(1+\zeta_a\zeta)$$

which is identical with the linear Lagrangian element.

#### 6.3.2.6 "Quadratic" element (20 nodes) (Fig. 6.15b)

Corner nodes:

$$N_a = \frac{1}{8}(1+\xi_a\xi)(1+\eta_a\eta)(1+\zeta_a\zeta)(\xi_a\xi+\eta_a\eta+\zeta_a\zeta-2)$$

Typical mid-side node:

$$\xi_a = 0, \quad \eta_a = \pm 1, \quad \zeta_a = \pm 1$$
$$N_a = \frac{1}{4}(1-\xi^2)(1+\eta_a\eta)(1+\zeta_a\zeta)$$

#### 6.3.2.7 "Cubic" elements (32 nodes) (Fig. 6.15b)

Corner node:

$$N_a = \frac{1}{64}(1+\xi_a\xi)(1+\eta_a\eta)(1+\zeta_a\zeta)[9(\xi^2+\eta^2+\zeta^2)-19]$$

Typical mid-side node:

$$\xi_a = \pm\frac{1}{3}, \quad \eta_a = \pm 1, \quad \zeta_a = \pm 1$$
$$N_a = \frac{9}{64}(1-\xi^2)(1+9\xi_a\xi)(1+\eta_a\eta)(1+\zeta_a\zeta)$$

When $\zeta_a\zeta = \zeta^2 = 1$ the above expressions reduce to those of Eqs. (6.10a) to (6.11b). Indeed such elements of three-dimensional type can be joined in a compatible manner to two-dimensional sheet or one-dimensional line elements of the appropriate type as shown in Fig. 6.15.

Once again the procedure for generating the shape functions follows that described in Figs. 6.9 and 6.10. Elements with varying degrees of freedom along the edges also can be derived following the same steps.

The equivalent of a Pascal triangle is now a tetrahedron and again we find a small number of surplus degrees of freedom—a situation of even greater magnitude than in two-dimensional analysis.

## 6.4 Other simple three-dimensional elements

The possibilities of simple shapes in three dimensions are greater, for obvious reasons, than in two dimensions. A quite useful series of elements can, for instance, be based on triangular prisms (wedges) (Fig. 6.16). Here again variants of the product Lagrange approach or of the "serendipity" type can be distinguished. The first element of both families, shown in Fig. 6.16a, is identical and the shape functions are

$$N_a = \frac{1}{2}L_a(1+\zeta_a\zeta), \quad a = 1, 2, \ldots, 6$$

where $\zeta$ is used as the parent direction perpendicular to triangular faces. For the "quadratic" element illustrated in Fig. 6.16b the shape functions are as follows.

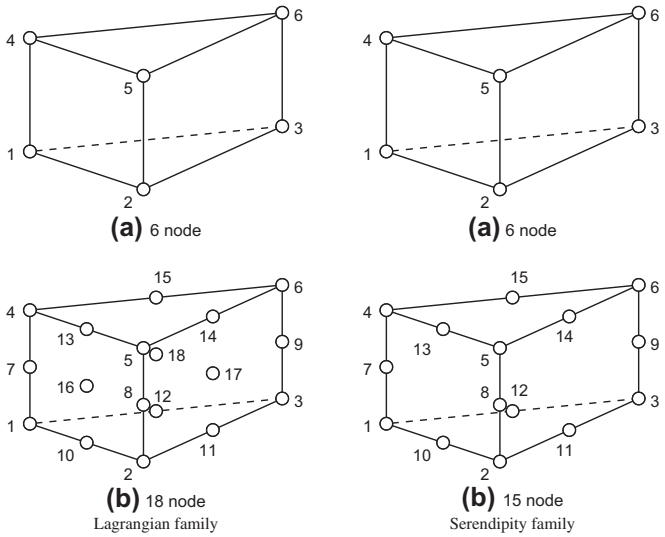



**FIGURE 6.16**
Triangular prism elements for the Lagrangian and serendipity family: (a) linear and (b) quadratic.

### 6.4.1 "Serendipity" quadratic

Corner nodes:

$$N_a = \frac{1}{2}L_a(2L_a - 1)(1 + \zeta_a\zeta) - \frac{1}{2}L_a(1 - \zeta^2), \quad a = 1, 2, \ldots, 6$$

Mid-edge of rectangle:

$$N_{a+6} = L_a(1 - \zeta^2), \quad a = 1, 2, 3$$

Mid-edge of triangles:

$$N_{10} = 2L_1L_2(1 - \zeta), \quad N_{11} = 2L_2L_3(1 - \zeta), \quad N_{12} = 2L_3L_1(1 - \zeta)$$
$$N_{13} = 2L_1L_2(1 + \zeta), \quad N_{14} = 2L_2L_3(1 + \zeta), \quad N_{15} = 2L_3L_1(1 + \zeta)$$

Similar elements may be developed for higher order elements as well as for the Lagrangian family. Such elements are not purely esoteric but have a practical application as "fillers" in conjunction with 20-node serendipity elements.

A similar wedge combining Lagrangian faces with triangular ones may be developed and used in conjunction with 27-node Lagrangian brick elements.

## 6.5 Mapping: Parametric forms

Once the shape functions are available in terms of parent coordinates we may immediately use the parametric mapping concept introduced in Section 3.5. Indeed, we do not always need to use the same interpolation for coordinates and dependent variables; however an isoparametric mapping is the most convenient since then nodal coordinates and nodal dependent variables are placed at the same physical locations. The use of isoparametric mappings was first used for a four-node quadrilateral element by Taig in 1957 [14]. Figure 6.17a shows the four-node rectangular element in parent coordinates $\xi$, $\eta$ and Fig. 6.17b shows the element after a parametric mapping given by

$$\mathbf{x}(\xi, \eta) = \sum_a N'_a(\xi, \eta)\, \mathbf{x}_a \tag{6.19}$$

where $N'_a = (1 + \xi_a \xi)(1 + \eta_a \eta)/4$. The map of the element must be such that all the vertex angles remain less than 180°, otherwise there will be multiple $\xi$, $\eta$ that have the same global coordinates $x$, $y$ at some points of the element or lie outside the boundary of the quadrilateral. An example of such a mapping is shown for a four-node quadrilateral in Fig. 6.18 where the parts above node 4 are areas with a negative area. Thus, the isoparametric mapping must always be such that single-valued mappings exist. This places restrictions on where the element nodes may be placed. Some curvilinear mappings for solid and surfaces are shown in Fig. 6.19. The use of mapping allows for any general shape to be approximated by the element types described above.

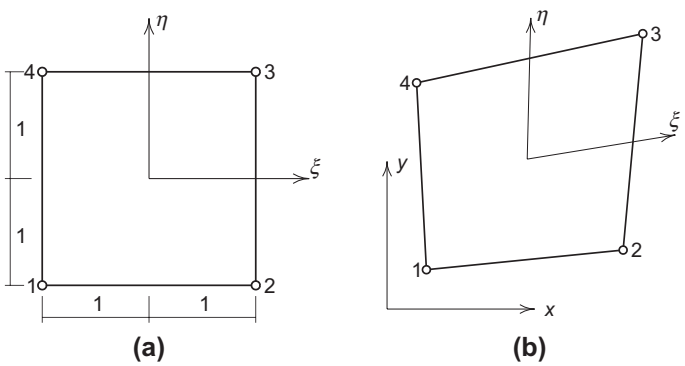



**FIGURE 6.17**
Isoparametric mapping of four-node quadrilateral element: (a) paren coordinates and (b) Global coordinates.

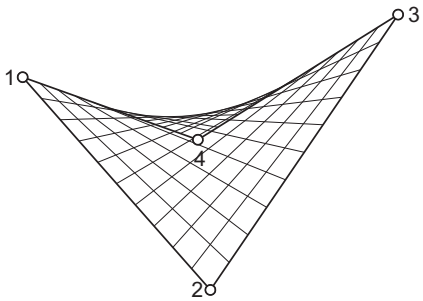



**FIGURE 6.18**
Inadmissible mapping of four-node quadrilateral element.

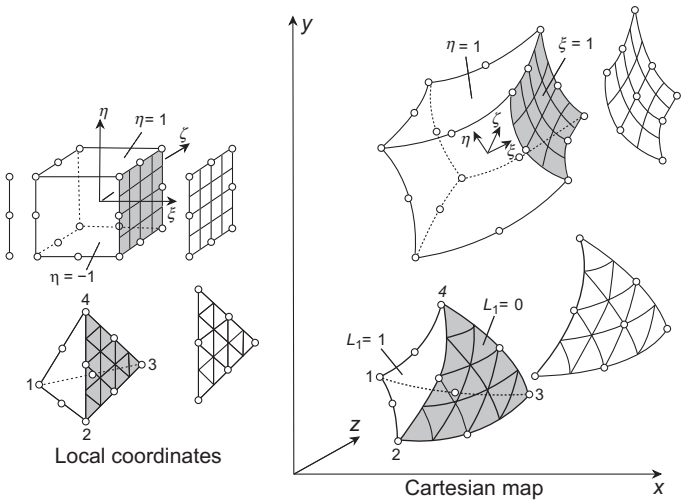



**FIGURE 6.19**
Isoparametric mapping of solids and surfaces.

The types of parametric mapping that can be considered in an analysis are defined by the relationship between the coordinate interpolation and the dependent variable interpolation. If we denote the mapping by

$$\begin{aligned} \mathbf{x}(\xi_j) &= \sum_a N'_a(\xi_j)\,\mathbf{x}_a \\ \hat{\phi}(\xi_j) &= \sum_a N_a(\xi_j)\,\tilde{\phi}_a \end{aligned} \tag{6.20}$$

where $\xi_j$ are any of the parent coordinates defined above, then we have the following three cases:

*Sub-parametric interpolation:* The order of the interpolation for $\mathbf{x}$ is lower than that for $\phi$.
*Isoparametric interpolation:* The order of the interpolation for $\mathbf{x}$ is the same as that for $\phi$.
*Super-parametric interpolation:* The order of the interpolation for $\mathbf{x}$ is higher than that for $\phi$.

In developing solutions to $C_0$ problems one may use either "sub-parametric" or "isoparametric" interpolations since either ensures that the polynomials $1, x, y$ and for three dimensions $z$ are always available, thus ensuring that constant derivatives can be computed. On the other hand use of "super-parametric" interpolation should generally be avoided.

## 6.6 Order of convergence for mapped elements

If the shape functions are chosen in curvilinear coordinate space so as to observe the usual rules of convergence (continuity and presence of complete first-order polynomials in these coordinates), then convergence will occur. In the case of isoparametric (or sub-parametric) elements a complete linear field is always reproduced (i.e., $1,\ x,\ y,\ z$) by the curvilinear coordinate expansion, and thus a constant derivative with respect to global coordinates will always exist.

The proof of this is simple. Consider a standard isoparametric expansion

$$\phi = \sum_{a=1}^{n} N_a \tilde{\phi}_a \equiv \mathbf{N}\,\tilde{\boldsymbol{\phi}}, \quad \mathbf{N} = \mathbf{N}(\xi, \eta, \zeta) \tag{6.21}$$

with coordinates of nodes defining the transformation as

$$x = \sum N_a x_a, \quad y = \sum N_a y_a, \quad z = \sum N_a z_a \tag{6.22}$$

The question is under what circumstances is it possible for expression (6.21) to define a linear expansion in Cartesian coordinates:

$$\begin{aligned} \phi &= \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 z \\ &\equiv \alpha_1 + \alpha_2 \sum N_a x_a + \alpha_3 \sum N_a y_a + \alpha_4 \sum N_a z_a \end{aligned} \tag{6.23}$$

If we take

$$\tilde{\phi}_a = \alpha_1 + \alpha_2 x_a + \alpha_3 y_a + \alpha_4 z_a$$

and compare expression (6.21) with (6.23) we note that identity is obtained between these providing

$$\sum N_a = 1$$

As this is the usual requirement of standard element shape functions [see Eq. (3.51)] we can conclude that *the constant derivative condition will be satisfied for all isoparametric elements.*

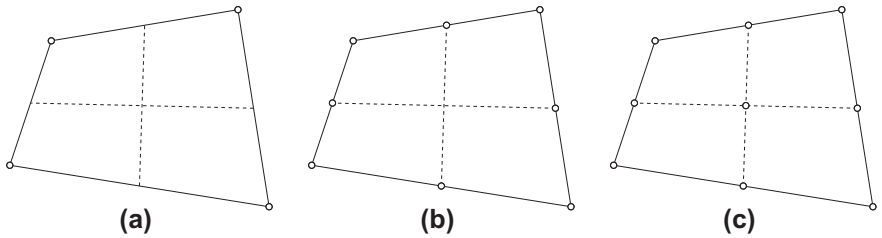



**FIGURE 6.20**
Bilinear mapping of sub-parametric quadratic eight- and nine-node elements: (a) four-node mapping, (b) eight-node element, (c) nine-node element.

As sub-parametric elements can always be expressed as specific cases of an isoparametric transformation this result is obviously valid here also.

It is of interest to pursue the argument and to see under what circumstances higher polynomial expansions in Cartesian coordinates can be achieved under various transformations. The simple linear case in which we "guessed" the solution has now to be replaced by considering in detail the polynomial terms occurring in expressions such as (6.21) and (6.23) and establishing conditions for equating appropriate coefficients.

Consider a specific two-dimensional problem: the circumstances under which the bilinear mapped quadrilateral of Fig. 6.20 can fully represent any quadratic Cartesian expansion. We now have

$$x = \sum_{a=1}^{4} N'_a x_a, \quad y = \sum_{a=1}^{4} N'_a y_a \tag{6.24}$$

and we wish to be able to reproduce

$$\phi = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 x^2 + \alpha_5 xy + \alpha_6 y^2 \tag{6.25}$$

Noting that the bilinear form of $N'_a$ contains terms such as 1, $\xi$, $\eta$, and $\xi\eta$, the above can be written as

$$\phi = \beta_1 + \beta_2\xi + \beta_3\eta + \beta_4\xi^2 + \beta_5\xi\eta + \beta_6\eta^2 + \beta_7\xi\eta^2 + \beta_8\xi^2\eta + \beta_9\xi^2\eta^2 \tag{6.26}$$

where $\beta_1$ to $\beta_9$ depend on the values of $\alpha_1$ to $\alpha_6$.

We shall now try to match the terms arising from the quadratic expansions of the serendipity kind shown in Fig. 6.20b where the interpolation is

$$\phi = \sum_{a=1}^{8} N_a \tilde{\phi}_a \tag{6.27}$$

with the appropriate shape functions of the kind defined in Section 6.2.2.4. We can also write (6.27) directly using polynomial coefficients $b_a$, $a = 1, \ldots, 8$, in place of

the nodal variables $\tilde{\phi}_a$ (noting the terms occurring in the Pascal triangle) as

$$\phi = b_1 + b_2\xi + b_3\eta + b_4\xi^2 + b_5\xi\eta + b_6\eta^2 + b_7\xi\eta^2 + b_8\xi^2\eta \tag{6.28}$$

It is immediately evident that for arbitrary values of $\beta_1$ to $\beta_9$ it is impossible to match the coefficients $b_1$ to $b_8$ due to the absence of the term $\xi^2\eta^2$ in Eq. (6.28). This term is also missing from the cubic serendipity kind, however if higher order quartic expansions were used such matching would evidently be possible and we could conclude that for linearly distorted elements the serendipity family of order 4 or greater will always represent quadratic polynomials in $x$, $y$.

For the nine-node, Lagrangian, (Fig. 6.20c) an expansion similar to (6.27) gives

$$\phi = \sum_{a=1}^{9} N_a\tilde{\phi}_a \tag{6.29}$$

which when expressed directly in polynomial coefficients $b_a$, $a = 1, \ldots, 9$, yields

$$\phi = b_1 + b_2\xi + b_3\eta + b_4\xi^2 + \cdots + b_8\xi^2\eta + b_9\xi^2\eta^2 \tag{6.30}$$

The matching of the coefficients of Eqs. (6.30) and (6.26) can be made directly.

We can conclude therefore that nine-node elements better represent Cartesian polynomials (when distorted linearly) and therefore are generally preferable in modeling smooth solutions. This matter was first presented by Wachspress [15] but the simple proof presented above is due to Crochet [16].

An example of this is given in Fig. 6.21 where we consider the results of a finite element calculation with eight- and nine-node elements, respectively, used to reproduce a simple quasi-harmonic solution in which we know that the exact answers are quadratic. The data for the quasi-harmonic solution is $k = 10$, $Q = 20$, $\bar{q}_n$ is zero on the top and bottom, $\bar{q}_n = -200$ on the right boundary, and $\bar{\phi} = 0$ on the left boundary. The exact solution is given by

$$\phi(x, y) = 20\,x - x^2$$

With no distortion both elements give exact results but when distorted only the nine-node element does so, with the eight-node element giving errors as shown in Fig. 6.21d. These errors also appear in the computed flux. While the errors are small for this problem, in elasticity applications the errors can be much larger. Similar arguments will lead to the conclusion that in three dimensions again only the Lagrangian 27-node element is capable of reproducing fully a quadratic function in Cartesian coordinates when trilinearly distorted (i.e., using the mapping for $N'_a$ for the eight-node hexahedron).

Lee and Bathe [17] investigate the problem for cubic and quartic serendipity and Lagrangian quadrilateral elements and show that under bilinear distortions the *full order* cartesian polynomial terms remain in Lagrangian elements but not in serendipity ones. They also consider edge distortion and show that this polynomial order is always lost. Additional discussion of such problems is also given by Wachspress [15].

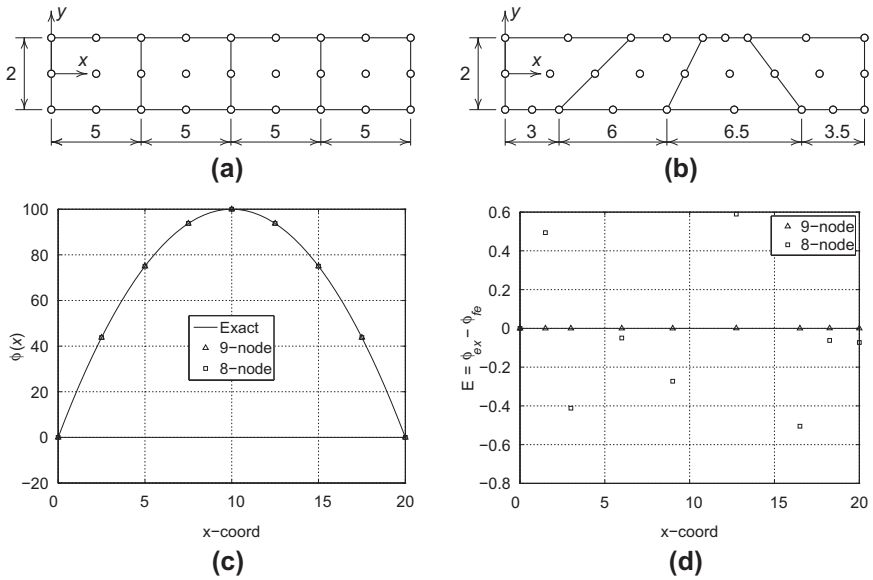



**FIGURE 6.21**

Quadratic eight-node serendipity and nine-node Lagrangian elements in regular and distorted form. Distribution of $\phi$ for quasi-harmonic solution with quadratic variation in $x$: (a) regular mesh, (b) distorted mesh, (c) $\phi$ solution, and (d) error $\phi$.

## 6.7 Computation of global derivatives

In order to define the gradient matrices appearing in weak forms we need to compute derivatives of the shape functions with respect to the global coordinates $x,\ y,\ z$ (or for the axisymmetric problem with respect to $r,\ z$). Consider, for instance, a two-dimensional case with the set of local coordinates $\boldsymbol{\xi} = [\xi,\ \eta]$ and a corresponding set of global coordinates $\mathbf{x} = [x,\ y]$. By the usual chain rules of partial differentiation we can write the $x$ and $y$ derivatives of any function $f(\xi, \eta)$ as

$$\begin{Bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{Bmatrix} = \begin{bmatrix} \dfrac{\partial \xi}{\partial x} & \dfrac{\partial \eta}{\partial x} \\ \dfrac{\partial \xi}{\partial y} & \dfrac{\partial \eta}{\partial y} \end{bmatrix} \begin{Bmatrix} \dfrac{\partial f}{\partial \xi} \\ \dfrac{\partial f}{\partial \eta} \end{Bmatrix} \tag{6.31}$$

If we consider two functions $f_1 = x$ and $f_2 = y$ then evaluation of (6.31) may be written in matrix form as

$$\begin{bmatrix} \dfrac{\partial f_1}{\partial x} & \dfrac{\partial f_2}{\partial x} \\ \dfrac{\partial f_1}{\partial x} & \dfrac{\partial f_2}{\partial x} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \dfrac{\partial \xi}{\partial x} & \dfrac{\partial \eta}{\partial x} \\ \dfrac{\partial \xi}{\partial y} & \dfrac{\partial \eta}{\partial y} \end{bmatrix} \begin{bmatrix} \dfrac{\partial x}{\partial \xi} & \dfrac{\partial y}{\partial \xi} \\ \dfrac{\partial x}{\partial \eta} & \dfrac{\partial y}{\partial \eta} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial \xi}{\partial x} & \dfrac{\partial \eta}{\partial x} \\ \dfrac{\partial \xi}{\partial y} & \dfrac{\partial \eta}{\partial y} \end{bmatrix} \mathbf{J} \tag{6.32}$$

The array $\mathbf{J}$ is known as the *Jacobian matrix* for the transformation and by the relation defining the parent coordinates [Eq. (6.19)], the matrix $\mathbf{J}$ can be found explicitly in terms of the local coordinates.

To find now the global derivatives for $f = N_a$ we invert $\mathbf{J}$ and write (6.31) as

$$\begin{Bmatrix} \dfrac{\partial N_a}{\partial x} \\ \dfrac{\partial N_a}{\partial y} \end{Bmatrix} = \mathbf{J}^{-1} \begin{Bmatrix} \dfrac{\partial N_a}{\partial \xi} \\ \dfrac{\partial N_a}{\partial \eta} \end{Bmatrix} \tag{6.33}$$

In terms of the mapping defining the coordinate transformation we have

$$\mathbf{J} = \begin{bmatrix} \sum_a \dfrac{\partial N_a}{\partial \xi} x_a & \sum_a \dfrac{\partial N_a}{\partial \xi} y_a \\ \sum_a \dfrac{\partial N_a}{\partial \eta} x_a & \sum_a \dfrac{\partial N_a}{\partial \eta} y_a \end{bmatrix} \tag{6.34}$$

The inverse of the Jacobian matrix is easily obtained from (6.32) and is given by

$$\mathbf{J}^{-1} = \frac{1}{j} \begin{bmatrix} \dfrac{\partial y}{\partial \eta} & -\dfrac{\partial y}{\partial \xi} \\ -\dfrac{\partial x}{\partial \eta} & \dfrac{\partial x}{\partial \xi} \end{bmatrix} \tag{6.35a}$$

where

$$j = \det \mathbf{J} = \frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta}\frac{\partial y}{\partial \xi} \tag{6.35b}$$

is the determinant of the Jacobian matrix.

For triangular elements it is necessary to modify the above since all the $L_j$ are not independent but are constrained by their sum being unity. Thus, to compute the derivatives we now include the constraint as one of the functions to be differentiated and express derivatives as

$$\frac{\partial f}{\partial x} = \sum_{i=1}^{3} \frac{\partial f}{\partial L_i}\frac{\partial L_i}{\partial x} \quad \text{and} \quad \frac{\partial f}{\partial y} = \sum_{i=1}^{3} \frac{\partial f}{\partial L_i}\frac{\partial L_i}{\partial y} \tag{6.36}$$

The derivatives of the area coordinates now may be obtained by differentiating the three equations

$$\mathbf{f} = \begin{Bmatrix} 1 \\ x \\ y \end{Bmatrix} = \begin{Bmatrix} \sum_{i=1}^{3} L_i \\ \sum_a N_a x_a \\ \sum_a N_a y_a \end{Bmatrix}$$

Using the chain rule, the derivatives of $\mathbf{f}$ with respect to $x$ and $y$ now may be written as

$$\begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ X_1 & X_2 & X_3 \\ Y_1 & Y_2 & Y_3 \end{bmatrix} \begin{bmatrix} \dfrac{\partial L_1}{\partial x} & \dfrac{\partial L_1}{\partial y} \\ \dfrac{\partial L_2}{\partial x} & \dfrac{\partial L_2}{\partial y} \\ \dfrac{\partial L_3}{\partial x} & \dfrac{\partial L_3}{\partial y} \end{bmatrix}$$

where we denote the parent derivatives by

$$X_i = \sum_a \frac{\partial N_a}{\partial L_i} x_a \quad \text{and} \quad Y_i = \sum_a \frac{\partial N_a}{\partial L_i} y_a$$

The solution yields

$$\begin{bmatrix} \dfrac{\partial L_1}{\partial x} & \dfrac{\partial L_1}{\partial y} \\ \dfrac{\partial L_2}{\partial x} & \dfrac{\partial L_2}{\partial y} \\ \dfrac{\partial L_3}{\partial x} & \dfrac{\partial L_3}{\partial y} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ X_1 & X_2 & X_3 \\ Y_1 & Y_2 & Y_3 \end{bmatrix}^{-1} \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} = \frac{1}{2\,\Delta} \begin{bmatrix} B_1 & C_1 \\ B_2 & C_2 \\ B_3 & C_3 \end{bmatrix} \tag{6.37}$$

where

$$\begin{aligned} B_1 &= Y_2 - Y_3\,, \quad C_1 = X_3 - X_2 \\ B_2 &= Y_3 - Y_1\,, \quad C_2 = X_1 - X_3 \\ B_3 &= Y_1 - Y_2\,, \quad C_3 = X_2 - X_1 \end{aligned} \tag{6.38}$$

and $\Delta = (X_1\,B_1 + X_2\,B_2 + X_3\,B_3)/2$ denotes the Jacobian transformation for a triangle when using quadrature to evaluate integrals. The reader should note the similarity to the results obtained when originally developing shape functions for the three-node triangle; however, the above result also holds for higher order interpolations.

### 6.7.1 Placement of element coordinates

As noted above the mapping of elements from parent to global coordinates must always be single valued. That is each point $\xi_i$[1] of the parent coordinates must map to one and only one global coordinate $\mathbf{x}$. The parametric map is controlled by the placement of nodal coordinates $\mathbf{x}_a$ for each element. The constraint to maintain a one-to-one mapping is given by the Jacobian determinant computed by the chain rule of differentiation. Denoting the Jacobian as $j(\xi_i)$ the requirement is

$$j(\xi_i) > 0 \quad \text{for all } \xi_i \tag{6.39}$$

[1] Where $\xi_i$ denotes any of the parent coordinates considered above.

For the lowest order elements one can establish criteria to ensure that the constraint is satisfied everywhere in the element. However, as the order of elements increases it is more difficult to ensure such criteria. To enforce the constraint the following must be checked:

1. Placement of edge nodes to make $j$ positive.
2. Placement of face nodes to make $j$ positive.
3. Placement of internal node to make $j$ positive.
4. All vertex angles are less than 180°.

Checking that the value of $j(\boldsymbol{\xi})$ is positive at each node of an element is sufficient to ensure the above conditions are satisfied and, thus, the specified parametric mapping produces a valid element.

## 6.8 Numerical integration

Once the shape functions and derivatives are known, the final step requires the evaluation of integrals to obtain the element matrices.

### 6.8.1 Quadrilateral elements

While integrals for a rectangular four-node element may be quite easily obtained, the integration becomes much more difficult for axisymmetric and mapped elements using parent coordinates. Thus, in general, it is best to consider the use of quadrature (numerical integration) as a general computational tool.

Using the isoparametric concept it is first necessary to change the domain over which the elements are integrated. This is a standard operation the reader learned in integral calculus. Thus, for a two-dimensional problem using quadrilateral elements the integral over $\Omega_e$ is converted to an integral over the parent domain $-1 \leq \xi, \eta \leq 1$ as

$$\int_{\Omega_e} g(x, y)\, \mathrm{d}x\, \mathrm{d}y = \int_{-1}^{1}\int_{-1}^{1} \hat{g}(\xi, \eta)\, j(\xi, \eta)\, \mathrm{d}\xi\, \mathrm{d}\eta \tag{6.40}$$

where $j(\xi, \eta)$ is the determinant of the Jacobian matrix described above. Using the isoparametric form the transformation of the integrand is accomplished as

$$g(x, y) = g(x(\xi, \eta), y(\xi, \eta)) = \hat{g}(\xi, \eta) \tag{6.41}$$

With the above choice for the domain of the parent element is now convenient to evaluate integrals using Gaussian quadrature as described in Section 3.5.3. Accordingly, we can integrate in two directions using

$$\int_{\Omega_e} g(x, y)\, \mathrm{d}x\, \mathrm{d}y = \sum_{n=1}^{N}\sum_{m=1}^{M} \hat{g}(\xi_m, \eta_n)\, j(\xi_m, \eta_n)\, w_m\, w_n + R_{NM} \tag{6.42}$$

where $m$ denotes quadrature points in the $\xi$ direction, and $n$ quadrature points in the $\eta$ direction. $R_{NM}$ is a remainder term representing the difference between the exact

integral and the approximation by the sum at the quadrature points. When using quadrature one needs to be assured that sufficient points are used such that the remainder term is of the same order or less than the other errors in the solution procedure. This is considered further in Section 6.8.5. The location and weight for quadrature points in each direction are those given in Table 3.3 for one-dimensional integrations.

### 6.8.2 Brick elements

Integrals for brick elements are performed in a similar manner to that used for quadrilateral shapes. For a three-dimensional problem the quadrature over the domain $\Omega_e$ is converted to an integral over the parent domain $-1 \le \xi, \eta, \zeta \le 1$ as

$$\int_{\Omega_e} g(x, y, z)\, \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z = \int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} \hat{g}(\xi, \eta, \zeta)\, j(\xi, \eta, \zeta)\, \mathrm{d}\xi\, \mathrm{d}\eta\, \mathrm{d}\zeta \tag{6.43}$$

where $j(\xi, \eta, \zeta)$ is the determinant of the Jacobian matrix for the three-dimensional brick. Again using the parametric form the transformation of the integrand is accomplished as

$$g(x, y) = g(x(\xi, \eta, \zeta), y(\xi, \eta, \zeta), z(\xi, \eta, \zeta)) = \hat{g}(\xi, \eta, \zeta) \tag{6.44}$$

Now three sums are required giving the numerical integration form

$$\int_{\Omega_e} g(x, y, z)\, \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z = \sum_{l=1}^{L}\sum_{m=1}^{M}\sum_{m=1}^{M} \hat{g}(\xi_l, \eta_m, \zeta_n)\, j(\xi_l, \eta_m, \zeta_n)\, w_l\, w_m\, w_n + R_{LNM} \tag{6.45}$$

where $l$ denotes quadrature points in the $\xi$ direction, $m$ quadrature points in the $\eta$ direction, and $n$ quadrature points in the $\zeta$ direction. $R_{LNM}$ is a remainder term representing the difference between the exact integral and the approximation by the sum at the quadrature points.

### 6.8.3 Triangular elements

For a triangle, in terms of the area coordinates the integrals are of the form

$$I = \int_0^1 \int_0^{1-L_1} f(L_1, L_2, L_3)\, \mathrm{d}L_2\, \mathrm{d}L_1, \quad L_3 = 1 - L_1 - L_2 \tag{6.46}$$

It is useful to note the following exact integration expression for triangular elements with constant Jacobian $j$:

$$\iint_{\Delta} L_1^a L_2^b L_3^c\, \mathrm{d}x\, \mathrm{d}y = \frac{a!\, b!\, c!}{(a + b + c + 2)!} 2\Delta \tag{6.47}$$

where $\Delta$ is the area of the triangle. However, for general shaped triangles it is again more convenient to use quadrature to evaluate integrals.

Once again we could use $n$ Gauss points and arrive at a summation expression of the type used for quadrilaterals. However, the limits of integration now involve the variable itself and it is convenient to use alternative sampling points for the second integration by use of a special Gauss expression for integrals in which the integrand is multiplied by a linear function. These have been devised by Radau [18] and used successfully in the finite element context [19]. It is, however, much more desirable (and aesthetically pleasing) to use special formulae in which no bias is given to any of the natural coordinates $L_a$. Such formulae were first derived by Hammer et al. [20] and Felippa [21] and a series of necessary sampling points and weights is given in Table 6.3 [22–24].

### 6.8.4 Tetrahedral elements

A similar extension for tetrahedra can obviously be made. For tetrahedral elements we note the exact integral expression is given for constant Jacobian elements by

$$\iiint_V L_1^a L_2^b L_3^c L_4^d \,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \frac{a!\,b!\,c!\,d!}{(a+b+c+d+3)!} 6V \tag{6.48}$$

where $V$ is the volume of the tetrahedron given in (5.24a). Similarly, Table 6.4 presents some quadrature formulae based on Ref. [20].

### 6.8.5 Required order of numerical integration

At this point the reader may inquire as to how many points need to be used to evaluate the arrays. There are some basic rules that should not be violated, however, it is obvious that use of quadrature will imply that all integrals in the arrays may not be exactly computed. Since we have already introduced approximation into the solution process by using polynomial shape functions all that is required is that the quadrature errors be less than these discretization errors. In addition, we need to ensure that some basic physical and mathematical properties are preserved. It is of interest, therefore, to determine (a) the minimum integration requirement permitting convergence and (b) the integration requirements necessary to preserve the rate of convergence which would result if exact integration were used.

It will be found later (Chapters 8 and 10) that it is in fact often a disadvantage to use higher orders of integration than those actually needed under (b) as, for very good reasons, a "cancellation of errors" due to discretization and due to inexact integration can occur.

#### 6.8.5.1 Minimum order of integration for convergence

In problems where the variational functional (or equivalent Galerkin integral statements) defines the approximation we have already stated that convergence can occur providing any arbitrary constant value of $m$th derivatives can be reproduced. In the present $C_0$ case $m = 1$ and we thus require that in integrals of $G^e$ a constant value be correctly integrated. *Thus the volume of the element* $\int_\Omega \mathrm{d}\Omega$ *needs to be evaluated correctly for convergence to occur*. In curvilinear coordinates we can thus argue that for a three-dimensional problem $\int j(\xi,\eta,\zeta)\mathrm{d}\xi\,d\eta\,d\zeta$ has to be evaluated exactly [25,26].

**Table 6.3** Numerical Integration Formulae for Triangles

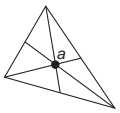
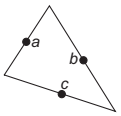
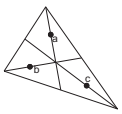
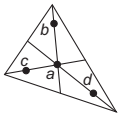
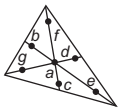
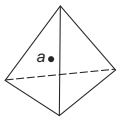
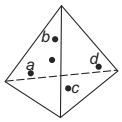
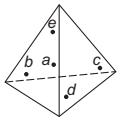
| Order | Figure | Error | Points | Triangular Coordinates | Weights |
|---|---|---|---|---|---|
| Linear |  | $R = O(h^2)$ | $a$ | $\frac{1}{3}, \frac{1}{3}, \frac{1}{3}$ | 1 |
| Quadratic |  | $R = O(h^3)$ | $a$ | $\frac{1}{2}, \frac{1}{2}, 0$ | $\frac{1}{3}$ |
| | | | $b$ | $\frac{1}{2}, 0, \frac{1}{2}$ | $\frac{1}{3}$ |
| | | | $c$ | $0, \frac{1}{2}, \frac{1}{2}$ | $\frac{1}{3}$ |
| Quadratic |  | $R = O(h^3)$ | $a$ | $\frac{2}{3}, \frac{1}{6}, \frac{1}{6}$ | $\frac{1}{3}$ |
| | | | $b$ | $\frac{1}{6}, \frac{2}{3}, \frac{1}{6}$ | $\frac{1}{3}$ |
| | | | $c$ | $\frac{1}{6}, \frac{1}{6}, \frac{2}{3}$ | $\frac{1}{3}$ |
| Cubic |  | $R = O(h^4)$ | $a$ | $\frac{1}{3}, \frac{1}{3}, \frac{1}{3}$ | $-\frac{27}{48}$ |
| | | | $b$ | 0.6, 0.2, 0.2 | $\frac{25}{48}$ |
| | | | $c$ | 0.2, 0.6, 0.2 | $\frac{25}{48}$ |
| | | | $d$ | 0.2, 0.2, 0.6 | $\frac{25}{48}$ |
| Quintic |  | $R = O(h^6)$ | $a$ | $\frac{1}{3}, \frac{1}{3}, \frac{1}{3}$ | 0.2250000000 |
| | | | $b$ | $\alpha_1, \beta_1, \beta_1$ | 0.1323941527 |
| | | | $c$ | $\beta_1, \alpha_1, \beta_1$ | 0.1323941527 |
| | | | $d$ | $\beta_1, \beta_1, \alpha_1$ | 0.1323941527 |
| | | | $e$ | $\alpha_2, \beta_2, \beta_2$ | 0.1259391805 |
| | | | $f$ | $\beta_2, \alpha_2, \beta_2$ | 0.1259391805 |
| | | | $g$ | $\beta_2, \beta_2, \alpha_2$ | 0.1259391805 |
| | | | | with | |
| | | | | $\alpha_1 = 0.059\,715\,871\,7$<br>$\beta_1 = 0.470\,142\,064\,1$<br>$\alpha_2 = 0.797\,426\,985\,3$<br>$\beta_2 = 0.101\,286\,507\,3$ | |

**Table 6.4** Numerical Integration Formulae for Tetrahedra

| Order | Figure | Error | Points | Tetrahedral Coordinates | Weights |
|---|---|---|---|---|---|
| Linear |  | $R = O(h^2)$ | $a$ | $\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}$ | 1 |
| Quadratic |  | $R = O(h^3)$ | $a$ | $\alpha, \beta, \beta, \beta$ | $\frac{1}{4}$ |
| | | | $b$ | $\beta, \alpha, \beta, \beta$ | $\frac{1}{4}$ |
| | | | $c$ | $\beta, \beta, \alpha, \beta$ | $\frac{1}{4}$ |
| | | | $d$ | $\beta, \beta, \beta, \alpha$ | $\frac{1}{4}$ |
| | | | | $\alpha = 0.585\,410\,20$<br>$\beta = 0.138\,196\,60$ | |
| Cubic |  | $R = O(h^4)$ | $a$ | $\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}$ | $-\frac{4}{5}$ |
| | | | $b$ | $\frac{1}{2}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}$ | $\frac{9}{20}$ |
| | | | $c$ | $\frac{1}{6}, \frac{1}{2}, \frac{1}{6}, \frac{1}{6}$ | $\frac{9}{20}$ |
| | | | $d$ | $\frac{1}{6}, \frac{1}{6}, \frac{1}{2}, \frac{1}{6}$ | $\frac{9}{20}$ |
| | | | $e$ | $\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{2}$ | $\frac{9}{20}$ |

### 6.8.5.2 *Order of integration for no loss of convergence rate*

In the one-dimensional problem we have already found that the finite element approximate evaluation of energy (and indeed all the other integrals in a Galerkin-type approximation, see Chapter 3) was exact to the order $2(p-m)$, where $p$ is the degree of the complete polynomial present and $m$ is the order of differential occurring in the appropriate expressions.

Providing the integration is exact to order $2(p-m)$, or shows an error of $O(h^{2(p-m)+1})$, or less, then no loss of convergence order will occur.[2] If in curvilinear

[2]For an energy principle use of quadrature may result in loss of a bound for $\Pi(\tilde{\mathbf{u}})$.

coordinates we take a curvilinear dimension $h$ of an element, the same rule applies. For $C_0$ problems (i.e., $m = 1$) the integration formulae should be as follows:

$$\begin{aligned} p &= 1, \quad \text{linear elements} \quad && O(h) \\ p &= 2, \quad \text{quadratic elements} \quad && O(h^3) \\ p &= 3, \quad \text{cubic elements} \quad && O(h^5) \end{aligned}$$

We shall make use of these results in practice, as will be seen later, but it should be noted that for a linear quadrilateral or triangle a single-point integration is adequate. For parabolic quadrilaterals (or bricks) $2 \times 2$ (or $2 \times 2 \times 2$), Gauss point integration is adequate and for parabolic triangles (or tetrahedra) the three-point (and four-point) formulae of Tables 6.3 and 6.4 are needed.

The basic theorems of this section have been introduced and proved numerically in published work [27–29].

### 6.8.6 Matrix singularity due to numerical integration

The final outcome of a semi-discretized finite element approximation of the quasi-harmonic equation was

$$\mathbf{C}\dot{\tilde{\boldsymbol{\phi}}} + \mathbf{H}\tilde{\boldsymbol{\phi}} = \mathbf{s} \tag{6.49}$$

For the one-dimensional elasticity problem it was of the form

$$\mathbf{M}\ddot{\tilde{\mathbf{u}}} + \mathbf{K}\tilde{\mathbf{u}} = \mathbf{f} \tag{6.50}$$

which we will also find in higher dimensional problems in the next chapter. For steady-state, static problems the first term is omitted. In each, if required, boundary conditions have been inserted. The solution of (6.49) or (6.50) gives an approximate solution for the physical situation. If a solution is unique, as is the case with well-posed physical problems, the equation matrix $\mathbf{C}$ or $\mathbf{M}$ should be positive definite (have full rank). The equation matrix $\mathbf{H}$ or $\mathbf{K}$ should have proper rank and be positive semi-definite. The rank of a square symmetric matrix is determined from an eigen-problem of the form

$$\mathbf{A}\,\boldsymbol{\Phi} = \boldsymbol{\Phi}\,\boldsymbol{\Lambda}$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix of eigenvalues and $\boldsymbol{\Phi}$ is a square matrix whose columns are the eigenvectors (see Appendix A). The rank of the matrix is the number of nonzero eigenvalues. If all eigenvalues are positive a matrix is positive definite; if all are positive or zero the matrix is positive semi-definite. Applying to (6.49), for the quasi-harmonic equation $\mathbf{C}$ should have rank $n - 1$ where $n$ is the size of the matrix without any boundary conditions. This implies the need for one boundary condition of the type $\phi_a = \bar{\phi}_a$ on one node. For elasticity $\mathbf{K}$ can admit rigid body modes only and we shall address this further in the next chapter.

We have *a priori* assumed that these conditions are satisfied with exact integration. With numerical integration, singularities may arise for low integration orders, and this may make such orders impractical. It is easy to show how, in some circumstances, a

singularity of a matrix must arise, but it is more difficult to prove that it will not. We shall, therefore, concentrate on the former case.

With numerical integration we replace the integrals by a weighted sum of independent linear relations between the nodal parameters (e.g., $\tilde{\boldsymbol{\phi}}$ for the quasi-harmonic equation). These linear relations supply the only information from which the matrices $\mathbf{H}$ and $\mathbf{C}$ are constructed. *If the number of unknowns exceeds the number of independent relations supplied at all the integrating points, then the matrix must be singular.*

To illustrate this point consider a two-dimensional quasi-harmonic problem where the evaluation of the element matrices by numerical integration may be written as

$$\mathbf{C}^e = \sum_{l=1}^{L} \mathbf{N}(\boldsymbol{\xi}_l)^{\mathrm{T}} c(\boldsymbol{\xi}_l)\, \mathbf{N}(\boldsymbol{\xi}_l)\, j(\boldsymbol{\xi}_l)\, W_l$$

$$\mathbf{H}^e = \sum_{l=1}^{L} \mathbf{b}(\boldsymbol{\xi}_l)^{\mathrm{T}} \mathbf{k}(\boldsymbol{\xi}_l)\, \mathbf{b}(\boldsymbol{\xi}_l)\, j(\boldsymbol{\xi}_l)\, W_l$$

where $L$ denotes the total number of quadrature points and $\boldsymbol{\xi}_l$ the appropriate points (e.g., $\xi$ or $L_1$, etc.). In the above, each matrix is evaluated at quadrature point $l$ and has the form

$$\mathbf{N}(\boldsymbol{\xi}_l) = \left[ N_1 \;\; N_2 \;\cdots\; N_n \right]_{\boldsymbol{\xi}_l}$$

$$\mathbf{b}(\boldsymbol{\xi}_l) = \begin{bmatrix} \dfrac{\partial N_1}{\partial x} & \dfrac{\partial N_2}{\partial x} & \cdots & \dfrac{\partial N_n}{\partial x} \\ \dfrac{\partial N_1}{\partial y} & \dfrac{\partial N_2}{\partial y} & \cdots & \dfrac{\partial N_n}{\partial y} \end{bmatrix}_{\boldsymbol{\xi}_l}$$

$$\mathbf{k}(\boldsymbol{\xi}_l) = \begin{bmatrix} k_{xx} & k_{xy} \\ k_{yx} & k_{yy} \end{bmatrix}_{\boldsymbol{\xi}_l}$$

where $n$ is the number of nodes on an individual element. The quadrature sum for each element array also may be written in the matrix form

$$\mathbf{C}^e = \left[ \mathbf{N}^{\mathrm{T}}(\boldsymbol{\xi}_1)\; \mathbf{N}^{\mathrm{T}}(\boldsymbol{\xi}_2) \cdots \mathbf{N}^{\mathrm{T}}(\boldsymbol{\xi}_L) \right] \begin{bmatrix} \widehat{c}(\boldsymbol{\xi}_l) & & & \\ & \widehat{c}(\boldsymbol{\xi}_l) & & \\ & & \ddots & \\ & & & \widehat{c}(\boldsymbol{\xi}_l) \end{bmatrix} \begin{bmatrix} \mathbf{N}(\boldsymbol{\xi}_1) \\ \mathbf{N}(\boldsymbol{\xi}_2) \\ \cdots \\ \mathbf{N}(\boldsymbol{\xi}_L) \end{bmatrix}$$

$$\mathbf{H}^e = \left[ \mathbf{b}^{\mathrm{T}}(\boldsymbol{\xi}_1)\; \mathbf{b}^{\mathrm{T}}(\boldsymbol{\xi}_2) \cdots \mathbf{b}^{\mathrm{T}}(\boldsymbol{\xi}_L) \right] \begin{bmatrix} \widehat{\mathbf{k}}(\boldsymbol{\xi}_l) & & & \\ & \widehat{\mathbf{k}}(\boldsymbol{\xi}_l) & & \\ & & \ddots & \\ & & & \widehat{\mathbf{k}}(\boldsymbol{\xi}_l) \end{bmatrix} \begin{bmatrix} \mathbf{b}(\boldsymbol{\xi}_1) \\ \mathbf{b}(\boldsymbol{\xi}_2) \\ \vdots \\ \mathbf{b}(\boldsymbol{\xi}_L) \end{bmatrix}$$

where

$$\widehat{c}(\boldsymbol{\xi}_l) = c(\boldsymbol{\xi}_l)\, j(\boldsymbol{\xi}_l)\, W_l \quad \text{and} \quad \widehat{\mathbf{k}}(\boldsymbol{\xi}_l) = \mathbf{k}(\boldsymbol{\xi}_l)\, j(\boldsymbol{\xi}_l)\, W_l$$

The rank of a nonsquare matrix is equal to or less than its smaller dimension. Also in estimating rank any row or column of all zeros is ignored. In the above the rank of $\mathbf{N}(\boldsymbol{\xi}_l)$ and $\widehat{c}(\boldsymbol{\xi}_l)$ is 1. Similarly, the rank of $\mathbf{b}(\boldsymbol{\xi}_l)$ and $\mathbf{k}(\boldsymbol{\xi}_l)$ is 2. For each quadrature point the rank of the resulting matrix products is the minimum of the rank of the contributing matrices. In accumulating quadrature points the rank may increase by the rank of each point, however, it will eventually reach a maximum due to linear dependence between the individual contributions and become the rank of the columns of the $\mathbf{N}$ and $\mathbf{b}$ matrices. For the quasi-harmonic equation the column rank of the $\mathbf{N}$ matrices is $n$, the number of element nodes. For any set of element shape functions which satisfy $\sum_{a=1}^{n} N_a(\boldsymbol{\xi}) = 1$ the sum of any existing derivatives can always be zero. Thus, the column rank of the $\mathbf{b}$ matrix will be $n - 1$, indicating the available "rigid body mode" in $\mathbf{C}^e$. Thus, at some number of $L$ the rank of the resulting matrix does not change with increasing number of points; only the accuracy of computed terms may change. Indeed, only in cases where all terms in the matrices are of polynomial form can the exact value (to within computer round-off) of an integral be computed (i.e., the remainder is zero in the quadrature formula) using the formulae given above and in Chapter 3.

Based on the above we can now estimate the number of points needed to compute the $\mathbf{C}^e$ and $\mathbf{H}^e$ matrices for the quasi-harmonic equation. For comparison we consider three- and six-node triangular elements and four-, eight-, and nine-node quadrilateral elements. For the elements considered, the computation of $\mathbf{C}^e$ requires at least one quadrature point for each element node. At each integrating point *two* independent "gradient relations" are used to compute $\mathbf{H}^e$ and the total number of independent relations possibly equals $2 \times$ (number of integration points). Since a constant value of $\phi$ does not change a derivative, the required rank for $\mathbf{C}^e$ is 1 less than the number of element nodes. Thus, we estimate that a single quadrature point can be used for the three-node element and three points can be used for the six-node element. For the quadrilateral element the estimate is two points for the four-node element, four points for the eight-node element, and five points for the nine-node element. Of course, suitable points need to exist. At present we have only identified use of one-dimensional Gaussian quadrature for each direction of quadrilaterals and brick elements, thus, for these the minimum estimates would use $2 \times 2$ quadrature ($L = 4$) for the four-and eight-node elements and $3 \times 3$ quadrature ($L = 9$) for the nine-node element. The above choices are shown in Fig. 6.22. *We emphasize that the above are merely estimates*. When computing an element by numerical integration it is always important to check individual element arrays to ensure they have proper rank.

The above results indicate that often different order quadrature is required in order to attain proper rank of the $\mathbf{C}^e$ and $\mathbf{H}^e$ matrices. In the above we have emphasized both rank and accuracy and we note that it is possible in low order elements to obtain sufficient accuracy before proper rank is obtained. We shall refer to the minimum value of $L$ that gives full rank as *normal order quadrature* and a value of $L - 1$ as reduced quadrature. There are instances when reduced quadrature has been used,

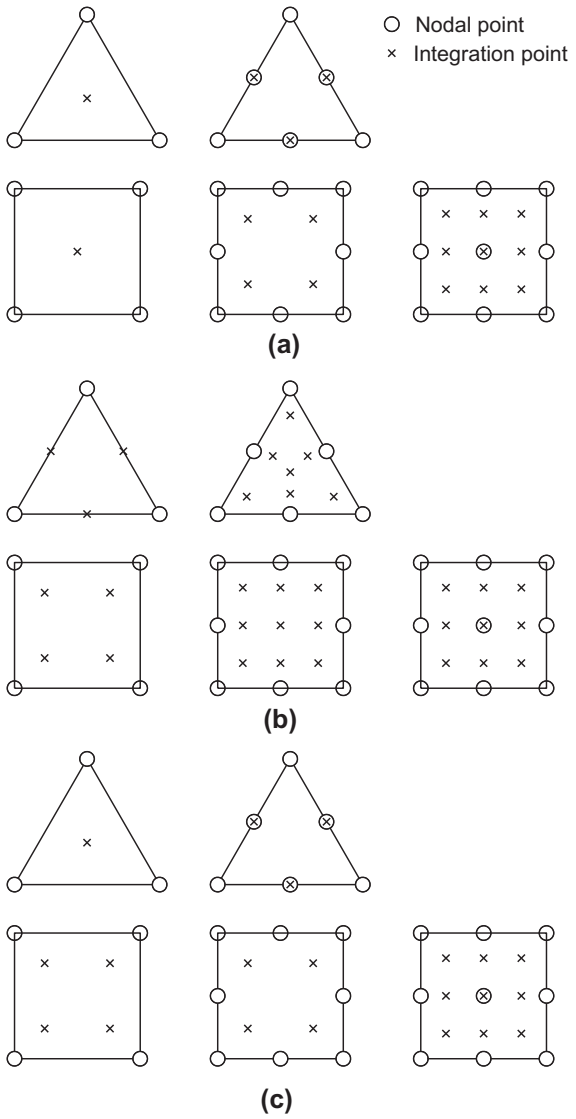



**FIGURE 6.22**

Required quadrature for two-dimensional quasi-harmonic problems: (a) exact Jacobian, (b) $\mathbf{C}^e$ (mass) matrix, and (c) $\mathbf{H}^e$ (stiffness) matrix.

for instance, it is commonly used in "explicit" dynamics codes—but requires certain remedial additions (e.g., hourglass control [30,31]).

In Chapter 8 we shall return to the problem of convergence and will indicate dangers arising from local element singularities.

However, it is of interest to mention that in Chapter 10 we shall in fact *seek* matrix singularities for special purposes (e.g., elastic incompressibility) using similar arguments.

## 6.9 Shape functions by degeneration

In the previous sections we have discussed the construction of shape functions for mapped elements of Lagrangian and serendipity types, as well as those for triangular and tetrahedral types. We have also shown how mixtures of interpolation forms may be used to construct elements of the prism type. One may ask what happens if we distort elements such that nodes for the Lagrangian or serendipity type are coalesced—that is, they are assigned the same node number in the mesh. We call the approach where two or more nodes are common a *degenerate form*. In a degenerate form the shape function for a coalesced set of two or more nodes is obtained by combining together the shape functions of sets of nodes (in a hierarchic form, any mid-side and/or face functions are omitted).

**Example 6.1.** Four-Node quadrilateral degenerated into a triangle

As a simple example we consider the degeneration of a four-node quadrilateral in which nodes 3 and 4 are coalesced to form the third node of a triangular element as shown in Fig. 6.23. For an isoparametric form given in $\xi, \eta$ coordinates, the shape functions for the degenerate triangular element are given by

$$\begin{aligned} N_1 &= \frac{1}{4}(1-\xi)(1-\eta) \\ N_2 &= \frac{1}{4}(1+\xi)(1-\eta) \\ N_3 &= \frac{1}{2}(1+\eta) \end{aligned} \tag{6.51}$$

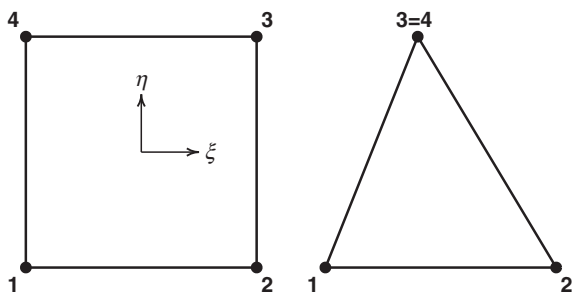



**FIGURE 6.23**
Degeneration of a quadrilateral into a triangle.

where the last function results from adding together the standard shape functions for nodes 3 and 4 of the quadrilateral element. Computing now the global derivatives for the above functions we obtain [using (6.31)]

$$\frac{\partial N_a}{\partial x} = \frac{b_a(1-\eta)}{2\,\Delta(1-\eta)}, \quad \frac{\partial N_a}{\partial y} = \frac{c_a(1-\eta)}{2\,\Delta(1-\eta)} \tag{6.52}$$

where $b_a$ and $c_a$ coincide with results for the standard three-node triangular element shape functions given in Section 6.2.1.1. Except for the point $\eta = 1$ (the point where the nodes are coalesced) the shape function derivatives are constant and identical to those obtained using area coordinates $L_1, L_2, L_3$. Thus, for the degeneration we have the identities

$$\begin{aligned} N_1 &= \frac{1}{4}(1-\xi)(1-\eta) = L_1 \\ N_2 &= \frac{1}{4}(1+\xi)(1-\eta) = L_2 \\ N_3 &= \frac{1}{2}(1+\eta) \qquad\quad = L_3 \end{aligned} \tag{6.53}$$

and, provided we do not consider the point $\eta = 1$, we may compute the derivatives and integrals for three-node triangular elements using the degeneration process.

A similar form to the above example holds when an eight-node brick element is degenerated into a four-node tetrahedron. In addition, however, we can compute shape functions for other degenerate forms as indicated in Fig. 6.24. In all cases, the computation of derivatives gives a $0/0$ form at any point where nodes are coalesced. In addition, however, any faces which degenerate into an edge will also contain a $0/0$ in the derivative along that edge. The behavior on any remaining face of a degenerate element is either the original quadrilateral one or a triangular one in which the shape functions are identical to the results given in (6.53).

### 6.9.1 Higher order degenerate elements

When nodes for higher order quadrilateral and hexahedral elements are coalesced to give a degenerate form, it is necessary to modify the shape functions for some of the noncoalesced nodes in order to produce results which are consistent with those computed using area or volume coordinates, respectively. This aspect was first studied by Newton [32] and Irons [33] for serendipity-type elements. Here we extend the work reported in these references to include the Lagrangian-type elements. Using Lagrangian elements has a distinct advantage since all the degenerate elements preserve the properties of higher order approximation in global coordinates when the element is mapped according to the trilinear form (i.e., a sub-parametric form using the eight-node hexahedron) (see Fig. 6.25).

**Example 6.2.** Quadratic quadrilateral degenerated to a triangular element

As an example we consider the degeneration of a quadratic order quadrilateral to form a quadratic order triangular element. Expressing the shape functions in hierarchical

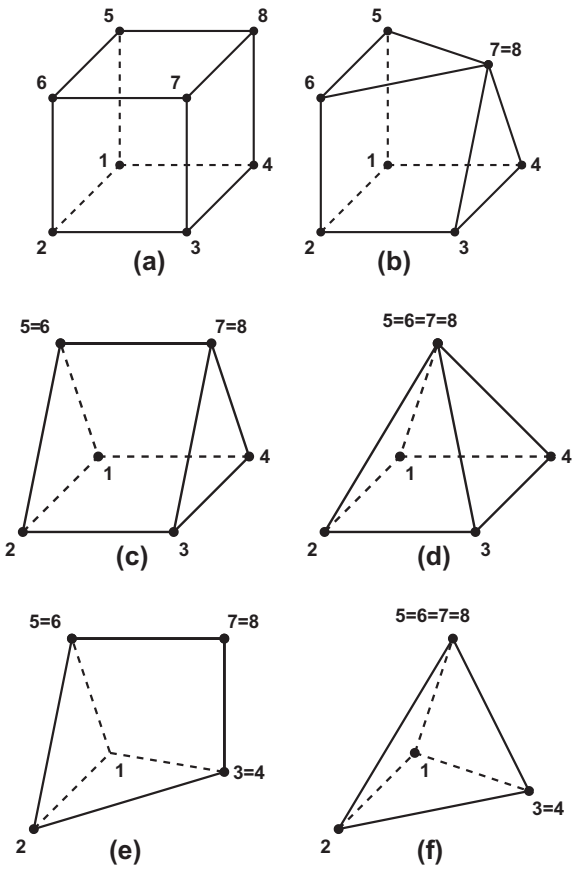



**FIGURE 6.24**
Some degenerate element forms for an eight-node brick element: (a) brick, (b) degenerate, (c) prism, (d) pyramid, (e) chisel, and (f) tetrahedron.

form (viz. Section 3.6) we have for the eight- or nine-node quadrilateral[3]

$$\begin{aligned} N_a^Q &= \frac{1}{4}(1+\xi_a\xi)(1+\eta_a\eta), \quad a = 1,2,3,4 \\ N_a^Q &= \frac{1}{2}(1+\eta_a\eta)(1-\xi^2), \quad a = 5,7 \\ N_a^Q &= \frac{1}{2}(1+\xi_a\xi)(1-\eta^2), \quad a = 6,8 \\ N_a^Q &= (1-\xi^2)(1-\eta^2), \quad a = 9 \end{aligned} \tag{6.54}$$

[3]We use a superscript "$Q$" for shape functions associated with the quadrilateral form and, later, "$T$" to denote those for a triangular form.

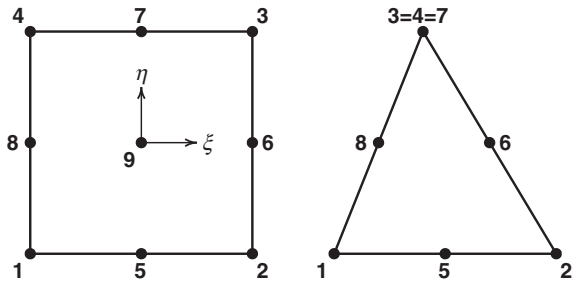



**FIGURE 6.25**
Degeneration of a eight- or nine-node quadrilateral into a six-node triangle.

for which a hierarchical Lagrangian interpolation of any function is given by

$$f = \sum_{a=1}^{4} N_a^Q(\xi, \eta)\, f_a + \sum_{a=5}^{8} N_a^Q(\xi, \eta)\, \Delta f_a + N_9^Q(\xi, \eta)\, \Delta\Delta f_9 \tag{6.55}$$

where $f_a$ are nodal values, $\Delta f_a$ are departures from linear interpolation for mid-side nodes, and $\Delta\Delta f_9$ is the departure from the eight-node serendipity interpolation at the center node. Thus, omitting the ninth function gives the serendipity form. If we now coalesce the nodes 3, 4, and 7 and use the above hierarchic form, the shape functions for the vertex nodes again are given by

$$\begin{aligned} N_1^Q &= \frac{1}{4}(1-\xi)(1-\eta) = L_1 = N_1^{\mathrm{T}} \\ N_2^Q &= \frac{1}{4}(1+\xi)(1-\eta) = L_2 = N_2^{\mathrm{T}} \\ N_3^Q &= \frac{1}{2}(1+\eta) = L_3 = N_3^{\mathrm{T}} \end{aligned} \tag{6.56}$$

(Note that $\Delta f_7 = 0$ in any interpolation and, thus, $N_7^{\mathrm{T}} = 0$.) Also, for the six-node form we omit the interior node 9 and thus, for the degenerate element $N_9^{\mathrm{T}} = 0$. If the resulting degenerate element is to be identical with the six-node triangular element we require

$$\begin{aligned} N_5^{\mathrm{T}} &= 4\, L_1\, L_2 \\ N_6^{\mathrm{T}} &= 4\, L_2\, L_3 \\ N_8^{\mathrm{T}} &= 4\, L_3\, L_1 \end{aligned} \tag{6.57}$$

Substituting the definitions for area coordinates given by (6.56) into (6.57) we find

$$\begin{aligned} N_5^{\mathrm{T}} &= \frac{1}{4}(1-\xi^2)(1-\eta)^2 \\ N_6^{\mathrm{T}} &= \frac{1}{2}(1+\xi)(1-\eta^2) \\ N_8^{\mathrm{T}} &= \frac{1}{2}(1-\xi)(1-\eta^2) \end{aligned} \tag{6.58}$$

and, thus, comparing the forms given by (6.55) and (6.58) we obtain the result

$$N_5^{\mathrm{T}} \neq N_5^{Q}, \quad N_6^{\mathrm{T}} = N_6^{Q}, \quad N_8^{\mathrm{T}} = N_8^{Q} \tag{6.59}$$

Thus, it only remains to correct the shape function for node 5. This is accomplished by noting

$$\begin{aligned} N_5^{\mathrm{T}} &= \frac{1}{4}(1-\xi^2)(1-2\,\eta+\eta^2) \\ &= \frac{1}{4}(1-\xi^2)(2-2\,\eta-1+\eta^2) \\ &= \frac{1}{2}(1-\xi^2)(1-\eta) - \frac{1}{4}(1-\xi^2)(1-\eta^2) \end{aligned}$$

giving the "corrected" degenerate function for node 5 as

$$N_5^{\mathrm{T}} = N_5^{Q} - \frac{1}{4}N_9^{Q} \tag{6.60}$$

The hierarchical forms now can be converted to standard isoparametric form using the process given for serendipity elements in Section 6.2.2.4.

**Example 6.3.** Degenerate forms for a quadratic 27-node hexahedron

The construction for quadratic degenerate three-dimensional forms follows a similar process and, when using the hierarchical form, the mid-side node opposite each coalesced node on a "face" must be modified using a form similar to (6.60). Again, all the shapes shown in Fig. 6.24 are possible and permit the construction of meshes which can use a mix of bricks, tetrahedra, and degenerate transition forms. In addition to the 8 vertex nodes it is necessary to add 12 mid-edge nodes, 6 mid-face nodes, and 1 internal node to form a Lagrangian quadratic order hexahedron. For node numbers as given in Fig. 6.26 and using hierarchical interpolation, the shape functions are given by the following:

**1.** For vertex nodes:

$$N_a = \frac{1}{8}(1+\xi_a\xi)(1+\eta_a\eta)(1+\zeta_a\zeta), \quad a = 1, 2, \ldots, 8 \tag{6.61a}$$

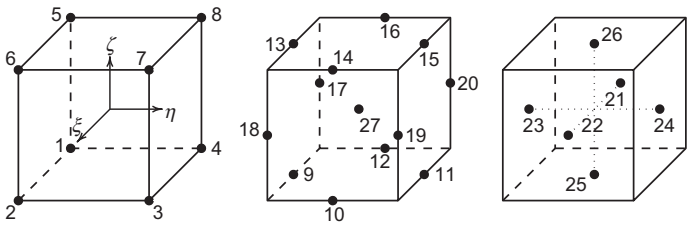



**FIGURE 6.26**
Numbering for 27-node quadratic Lagrangian hexagon (node 27 at origin of $\xi,\eta,\zeta$ coordinates).

**2.** For mid-edge nodes:

$$N_a = \frac{1}{4}\begin{cases}(1-\xi^2)(1+\eta_a\eta)(1+\zeta_a\zeta), & a = 9, 11, 13, 15\\ (1+\xi_a\xi)(1-\eta^2)(1+\zeta_a\zeta), & a = 10, 12, 14, 16\\ (1+\xi_a\xi)(1+\eta_a\eta)(1-\zeta^2), & a = 17, 18, 19, 20\end{cases} \tag{6.61b}$$

**3.** For mid-face nodes:

$$N_a = \frac{1}{2}\begin{cases}(1+\xi_a\xi)(1-\eta^2)(1-\zeta^2), & a = 21, 22\\ (1-\xi^2)(1+\eta_a\eta)(1-\zeta^2), & a = 23, 24\\ (1-\xi^2)(1-\eta^2)(1+\zeta_a\zeta), & a = 25, 26\end{cases} \tag{6.61c}$$

**4.** For the interior node:

$$N_a = (1-\xi^2)(1-\eta^2)(1-\zeta^2), \quad a = 27 \tag{6.61d}$$

Table 6.5 indicates which shape functions are modified when vertex nodes are coalesced. The hierarchical shape functions to be omitted are also indicated. Note that shape functions should only be omitted (set to zero) after all coalesced node pairs are considered. Also if a tetrahedral element is formed then all mid-face nodes are deleted and the interior node may also be omitted, giving the final tetrahedron as a 10-node element. Again, if any of the element forms is mapped using the degenerate sub-parametric form of the eight-node hexahedron for $N_a'$ full quadratic behavior in global coordinates is attained—showing the advantage of starting from Lagrangian form elements.

Consideration of cubic and higher order forms is also possible and is left as an exercise for the interested reader.

## 6.10 Generation of finite element meshes by mapping

It will be observed that it is an easy matter to obtain a coarse subdivision of the analysis domain with a small number of isoparametric elements. If second- or third-degree

elements are used, the fit of these to quite complex boundaries is reasonable, as shown in Fig. 6.27a where three parabolic elements specify a sectorial region. This number of elements would be too small for analysis purposes *but a simple subdivision into finer elements* can be done automatically using increments of the parent coordinates to define new element nodes and boundaries. The process thus allows us, with a small amount of original *input data*, to derive a finite element mesh of any refinement desirable. In Ref. [34] this type of mesh generation is developed for two- and three-dimensional solids and surfaces and is reasonably efficient. However, elements of predetermined size and/or gradation cannot be easily generated.

The main drawback of the mapping and generation suggested is the fact that the originally circular boundaries in Fig. 6.27a are approximated by simple parabolas and a geometric error can be developed there. To overcome this difficulty other forms of mapping can be adopted for this purpose.

An early form of mapping is the use of blending functions, originally developed for the representation of complex motor-car body shapes [35]. A simple use of blending functions is demonstrated in Fig. 6.27b to define the region with exact circular arcs instead of the parabolic ones from serendipity elements. These may be similarly subdivided to create the mesh shown in Fig. 6.27c.

### 6.10.1 Blending functions

Blending functions interpolate the unknown $\phi$ in such a way as to satisfy *exactly* its variations along the edges of a square $\xi$, $\eta$ domain. If the coordinates $x$ and $y$ are used in a parametric expression of the type given in Eq. (6.19), then any complex shape can be mapped by a single element. In Ref. [35] the region of Fig. 6.27 is in fact so mapped and a mesh subdivision obtained directly without any geometric error on the boundary.

**Table 6.5** Degeneration Modifications for 27-Node Hexahedron

| Coalesced Nodes | Modified Nodes | |
|---|---|---|
| 1 and 2 omit 9 | 11 by 25 omit 25 | 13 by 23 omit 23 |
| 2 and 3 omit 10 | 12 by 25 omit 25 | 14 by 22 omit 22 |
| 3 and 4 omit 11 | 9 by 25 omit 25 | 15 by 24 omit 24 |
| 4 and 1 omit 12 | 10 by 25 omit 25 | 16 by 21 omit 21 |
| 5 and 6 omit 13 | 15 by 26 omit 26 | 9 by 23 omit 23 |
| 6 and 7 omit 14 | 16 by 26 omit 26 | 10 by 22 omit 22 |
| 7 and 8 omit 15 | 13 by 26 omit 26 | 11 by 24 omit 24 |
| 8 and 5 omit 16 | 14 by 26 omit 26 | 12 by 21 omit 21 |
| 1 and 5 omit 17 | 18 by 23 omit 23 | 20 by 21 omit 21 |
| 2 and 6 omit 18 | 19 by 22 omit 22 | 17 by 23 omit 23 |
| 3 and 7 omit 19 | 20 by 24 omit 24 | 18 by 22 omit 22 |
| 4 and 8 omit 20 | 17 by 21 omit 21 | 19 by 24 omit 24 |

The blending processes are of considerable importance and have been used to construct some interesting element families [36] (which in fact include the standard serendipity elements as a subclass). To explain the process we shall show how a function with prescribed variations along the boundaries can be interpolated.

Consider a region $-1 \leq \xi, \eta \leq 1$, shown in Fig. 6.28, on the edges of which an arbitrary function $\phi$ is specified [i.e., $\phi(-1, \eta)$, $\phi(1, \eta)$, $\phi(\xi, -1)$, $\phi(\xi, 1)$ are given]. The problem presented is that of interpolating a function $\phi(\xi, \eta)$ so that a smooth surface reproducing precisely the boundary values is obtained. Writing

$$\begin{aligned} N_1(\xi) &= \frac{1}{2}(1-\xi), \quad N_2(\xi) = \frac{1}{2}(1+\xi) \\ N_1(\eta) &= \frac{1}{2}(1-\eta), \quad N_2(\eta) = \frac{1}{2}(1+\eta) \end{aligned} \tag{6.62}$$

for our usual one-dimensional linear interpolating functions, we note that

$$P_\eta \phi \equiv N_1(\eta)\phi(\xi, -1) + N_2(\eta)\phi(\xi, 1) \tag{6.63}$$

interpolates linearly between the specified functions in the $\eta$ direction, as shown in Fig. 6.28b. Similarly,

$$P_\xi \phi \equiv N_1(\xi)\phi(-1, \eta) + N_2(\xi)\phi(1, \eta) \tag{6.64}$$

interpolates linearly in the $\xi$ direction (Fig. 6.28c). Constructing a third function which is a standard bilinear interpolation of the kind we have already encountered

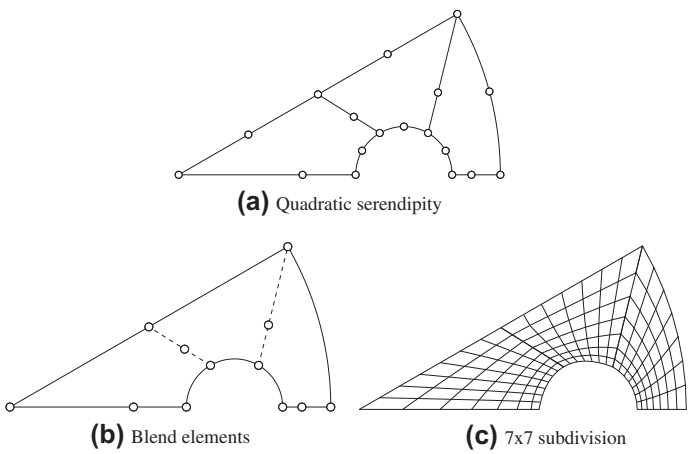



**FIGURE 6.27**
Mesh generation by quadratic isoparametric elements: (a) quadratic serendipity element, (b) specified blending points, and (c) automatic subdivision into linear quadrilaterals.

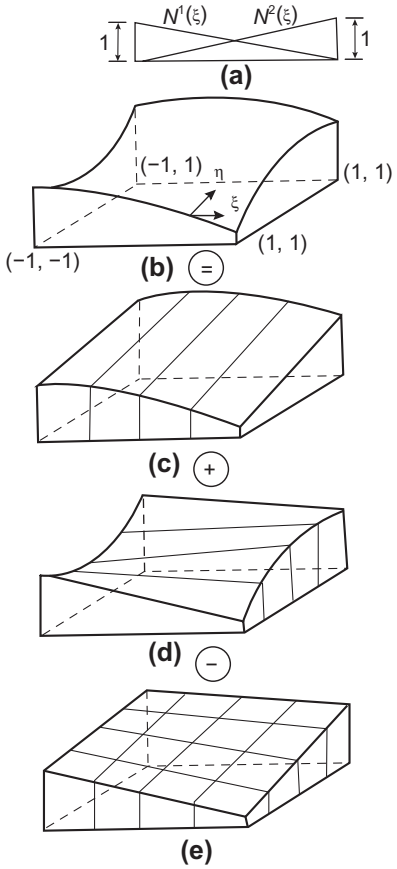



**FIGURE 6.28**
Stages of construction of a blending interpolation (a)–(e).

(Fig. 6.28d), i.e.,

$$P_\xi P_\eta \phi = N_1(\xi)N_1(\eta)\phi(-1,-1) + N_1(\xi)N_2(\eta)\phi(-1,1) + N_2(\xi)N_1(\eta)\phi(1,-1) + N_2(\xi)N_2(\eta)\phi(1,1) \tag{6.65}$$

we note by inspection that

$$\phi(\xi,\eta) = P_\eta\phi + P_\xi\phi - P_\xi P_\eta\phi \tag{6.66}$$

is a smooth surface interpolating exactly the boundary functions.

Extension to functions with higher order blending is almost evident, and immediately the method of mapping the quadrilateral region $-1 \le \xi, \eta \le 1$ to any arbitrary shape is obvious.

Though the above mesh generation method derives from mapping and indeed has been widely applied in two and three dimensions, we shall see in the chapter devoted to adaptivity (Chapter 16) that the optimal solution or specification of *mesh density* or *size* should guide the mesh generation. In Chapter 17 we will discuss in much more detail how meshes with prescribed density can be generated.

### 6.10.2 Bèzier functions

For simple shapes an alternative parameterization of the mapping function that also permits specification of many geometric shapes is the use of Bèzier functions. On the line defined by parent coordinate $-1 \le \xi \le 1$, the $p$th order Bèzier shape functions are given by the Bernstein polynomials [37]

$$B_{a+1}^{(p)}(\xi) = \frac{p!}{2^p\, a!(p-a)!}(1+\xi)^a(1-\xi)^{p-a}, \quad a = 0, 1, \ldots, p \tag{6.67}$$

The end points for a Bèzier interpolation are associated with the points 1 and $p+1$. For the linear case $p = 1$ and the two functions are given by

$$B_1^{(1)}(\xi) = \frac{1}{2}(1-\xi) \quad \text{and} \quad B_2^{(1)}(\xi) = \frac{1}{2}(1+\xi) \tag{6.68a}$$

These are identical to linear Lagrange interpolation. The quadratic order functions are given by

$$\begin{aligned} B_1^{(2)}(\xi) &= \frac{1}{4}(1-\xi)^2 \\ B_2^{(2)}(\xi) &= \frac{1}{2}(1-\xi)(1+\xi) \\ B_3^{(2)}(\xi) &= \frac{1}{4}(1+\xi)^2 \end{aligned} \tag{6.68b}$$

and shown in Fig. 6.29.

Bèzier shape functions satisfy the usual condition

$$\sum_a B_a^{(p)}(\xi) = 1 \tag{6.69}$$

however, when used to describe the coordinates

$$\mathbf{x} = \sum_a B_a^{(p)}(\xi)\, \tilde{\mathbf{x}}_a \tag{6.70}$$

the curve does not pass through all the $\tilde{\mathbf{x}}_a$ points. The $\tilde{\mathbf{x}}_a$ are called the *control points* and define the geometry of any curve. This is illustrated in Fig. 6.30 as the parabola. It is also possible to modify the functions to a rational form as

$$R_a^{(p)}(\xi) = \frac{w_a\, B_a^{(p)}(\xi)}{w(\xi)} \tag{6.71a}$$

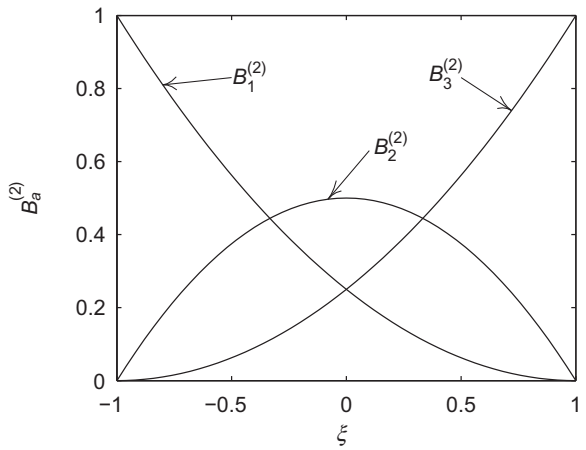



**FIGURE 6.29**
Quadratic Bèzier shape functions.

where

$$w(\xi) = \sum_a B_a^{(p)}(\xi)\, w_a \tag{6.71b}$$

defines a projective coordinate. The end values of $w_a$ are always set to unity to ensure a function interpolates the end points. In the rational form it is possible to create an infinite number of curves by varying the value for the interior $w_a$. For example, a circular arc may be created using the quadratic functions by setting $w_3 = \cos(\theta)$ as shown in Fig. 6.30. Setting all the $w_a$ to unity recovers

$$R_a^{(p)}(\xi) = B_a^{(p)}(\xi)$$

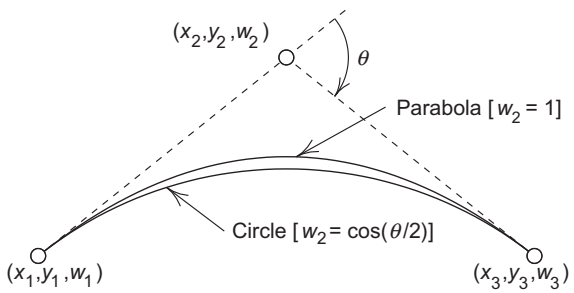



**FIGURE 6.30**
Bèzier curves: parabola $w_3 = 1$; circular arc $w_3 = \cos(\theta)$.

Two- and three-dimensional geometry may be created by using tensor products of the one-dimensional rational Bèzier functions. Thus, for two-dimensional surfaces one can use the form

$$\mathbf{x} = \sum_a \sum_b R_a^{(p)}(\xi) R_b^{(q)}(\eta) \tilde{\mathbf{x}}_{ab} \tag{6.72a}$$

and for three dimensions the form

$$\mathbf{x} = \sum_a \sum_b \sum_c R_a^{(p)}(\xi) R_b^{(q)}(\eta) R_c^{(r)}(\zeta)\, \tilde{\mathbf{x}}_{abc} \tag{6.72b}$$

may be used. Use of these permits the geometry of conic sections to be exactly matched over each Bèzier patch.

Bèzier functions can also be used to define elements, however, to retain a complete linear polynomial in the dependent variable it is necessary to use Bèzier functions for the coordinates also.

The use of the rational form for shape functions may be extended further to create *non-uniform rational B-splines* (NURBS) [37–39]. NURBS are commonly used by many computer-aided design (CAD) systems to create geometric models of physical entities. The output from a CAD system is usually not in a form that can be directly used for analysis. Often features of the design that do not affect analysis results need to be modified or removed. Once the model is in a form for analysis purposes, a "mesh" of nodes and elements must be created. The output of this step gives the finite element mesh for analysis and is followed by additional steps to assign appropriate boundary and initial conditions.

## 6.11 Computational advantage of numerically integrated finite elements

One considerable gain that is possible in numerically integrated finite elements is the versatility that can be achieved in a single computer program [40]. It will be observed that for a *given class of problems* the general matrices are always of the same form [see the example of Eq. (5.16)] in terms of the shape function and its derivatives.

To proceed with evaluation of the element properties it is necessary first to *specify the shape function* and its derivatives and, second, to *specify the order of integration*. The computation of element properties is thus composed of three distinct parts as shown in Fig. 6.31. For a *given class of problems* it is only necessary to change the prescription of the shape functions to achieve a variety of possible elements. Conversely, the *same shape function* routines can be used in many different classes of problem.

Use of different elements, testing the efficiency of a new element in a given context, or extension of programs to deal with new situations can thus be readily achieved, and considerable algebra avoided (with its inherent possibilities of mistakes). The

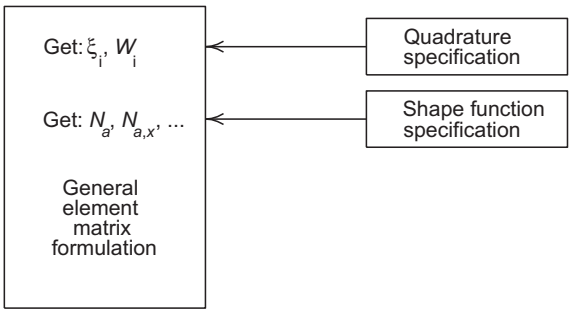



**FIGURE 6.31**
Computational scheme for numerically integrated elements ($N_{a,x} = \partial N_a / \partial x$).

computer is thus placed in the position it deserves, i.e., of being the obedient slave capable of saving routine work.

The greatest practical advantage of the use of universal shape function routines is that they can be checked decisively for errors by a simple program with the patch test playing the crucial role (viz. Chapter 8).

The incorporation of simple, exactly integrable, elements in such a system is, incidentally, not penalized as the time of exact and numerical integration in many cases is almost identical.

## 6.12 Problems

**6.1.** Develop an explicit form of the standard shape functions at nodes 1, 3, and 6 for the element shown in Fig. 6.32a.
Using a Pascal triangle in $\xi$ and $\eta$ show the polynomials included in the element.

**6.2.** Develop an explicit form of the standard shape functions at nodes 2, 3, and 9 for the element shown in Fig. 6.32b.
Using a Pascal triangle in $\xi$ and $\eta$ show the polynomials included in the element.

**6.3.** Develop an explicit form of the standard shape functions at nodes 1, 2, and 5 for the element shown in Fig. 6.32c.
Using a Pascal triangle in $\xi$ and $\eta$ show the polynomials included in the element.

**6.4.** Compute the parent coordinate first derivatives for the shape functions obtained in Problem 6.3.

**6.5.** Develop an explicit form of the standard shape functions at nodes 1, 2, and 5 for the element shown in Fig. 6.33a.
Using a Pascal triangle in $\xi$ and $\eta$ show the polynomials included in the element.

**6.6.** Develop an explicit form of the standard shape functions at nodes 1, 5, and 7 for the element shown in Fig. 6.33b.
Using a Pascal triangle in $\xi$ and $\eta$ show the polynomials included in the element.

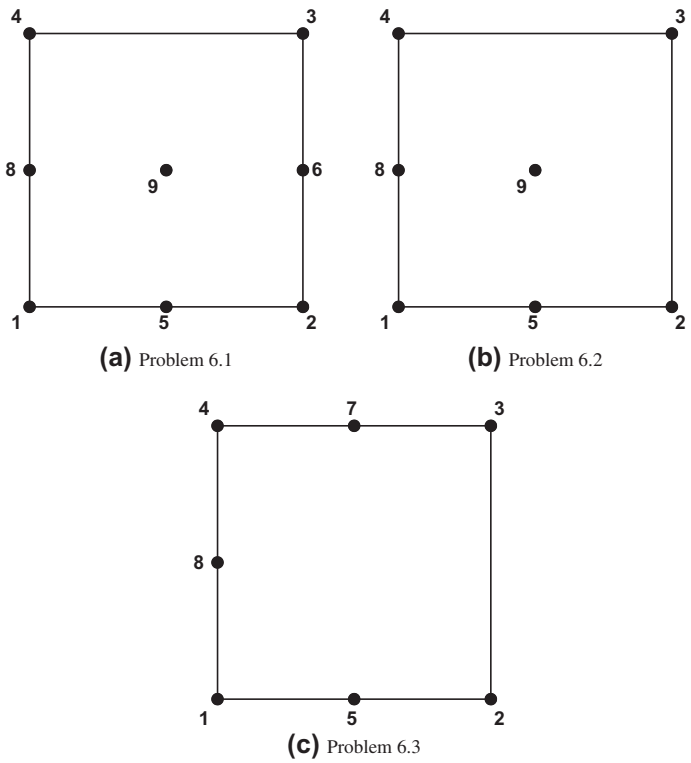
**6.7** The mesh for a problem contains an eight-node quadratic serendipity rectangle adjacent to a six-node quadratic triangle as shown in Fig. 6.34. Show that the coordinates computed from each element satisfy $C^0$ continuity along the edge 3-7-11.

**6.8** Determine an explicit expression for the shape function of node 1 of the linear triangular prism shown in Fig. 6.16a.

**6.9** Determine an explicit expression for the hierarchical shape function of nodes 1, 7, and 10 of the quadratic triangular prism of Lagrangian family shown in Fig. 6.16b.

**6.10** Determine an explicit expression for the shape function of nodes 1, 7 and 10 of the quadratic triangular prism of serendipity family shown in Fig. 6.16.

**6.11** On a sketch show the location of the nodes for the quartic member of the tetrahedron family. Construct an explicit expression for the shape function of the vertex node located at $(L_1, L_2, L_3, L_4) = (1, 0, 0, 0)$ and the mid-edge node located at $(0.25, 0.75, 0, 0)$.




**(a)** Problem 6.1 **(b)** Problem 6.2 **(c)** Problem 6.3

**FIGURE 6.32**
Quadrilateral element for Problems 6.1–6.3.

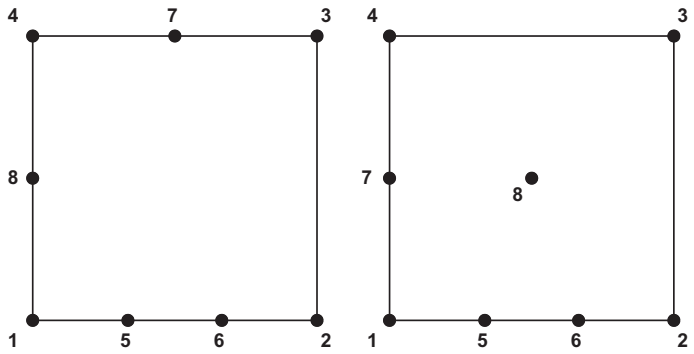



**FIGURE 6.33**
Quadrilateral element for Problems 6.5 and 6.6.

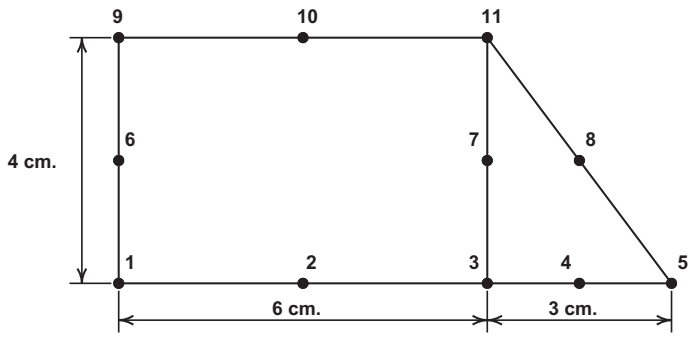



**FIGURE 6.34**
Quadratic rectangle and triangle for Problem 6.7.

**6.12** On a sketch show the location of the nodes for the quartic member of the serendipity family. Construct an explicit expression for the shape function of the vertex node located at $(\xi, \eta, \zeta) = (1, 1, 1)$ and the mid-edge node located at $(0.75, 1, 1)$.

**6.13** On a sketch show the location of the nodes for the cubic member of the triangular prism family shown in Fig. 6.16. Construct an explicit expression for the hierarchical shape function of a vertex node, an edge node of a triangular face, and an edge node of a rectangular face.

**6.14** On a sketch show the location of the nodes for the quadratic member of the triangular prism family in which Lagrangian interpolation is used on rectangular faces (see Fig. 6.16). Construct an explicit expression for the shape function of a

vertex node, an edge node of a triangular face, and an edge node of a rectangular face.

**6.15** On a sketch show the location of the nodes for the cubic member of the triangular prism family in which lagrangian interpolation is used on rectangular faces (see Fig. 6.16). Construct an explicit expression for the shape function of a vertex node, an edge node of a triangular face, an edge node of a rectangular face, a mid-face node of a triangular face, a mid-face node of a rectangular face, and for any internal nodes.

**6.16** A quadratic one-dimensional element is shown in Fig. 6.35 in parent form and in the mapped configuration. Let $a + b = h$ the total length of the mapped element:

- **(a)** Determine the shape functions $N_a(\xi)$ for the three nodes.
- **(b)** Plot $\xi$ vs. $x$ for values of $a$ ranging from $0.2\,h$ to $0.8\,h$ in increments of $0.1\,h$.
- **(c)** Plot $N_a$ vs. $x$ for the range of $a$ given in part (b).
- **(d)** Plot $\mathrm{d}N_a/\mathrm{d}x$ vs. $x$ for the range of $a$ given in part (b).

**6.17** Consider the one-dimensional problem for $0 \le x \le 1$ which is defined by the weak form

$$\int_0^1 \left[ \frac{\mathrm{d}\delta u}{\mathrm{d}x} \frac{\mathrm{d}u}{\mathrm{d}x} - \delta u\, q \right] \mathrm{d}x - \delta u\, \sigma \Big|_{x=1} = 0 \quad \text{with } u(0) = 0$$

with $q = \sigma = 1$:

- **(a)** Deduce the Euler differential equation and boundary conditions for the problem.
- **(b)** Construct an exact solution to the differential equation.
- **(c)** Solve the weak form using a single quadratic order element with nodes placed at $x = 0, 5/16$, and 1 and shape functions $N_a$ defined by:
  - **i.** Lagrange interpolation in $x$ directly
  - **ii.** Isoparametric interpolation for $N_a(\xi)$ with $x = N_a(\xi)\tilde{x}_a$

  Evaluate all integrals using two-point Gaussian quadrature.
- **(d)** Plot $u$ and $\mathrm{d}u/\mathrm{d}x$ for the two solutions. Comment on the differences in quality of the two solutions.

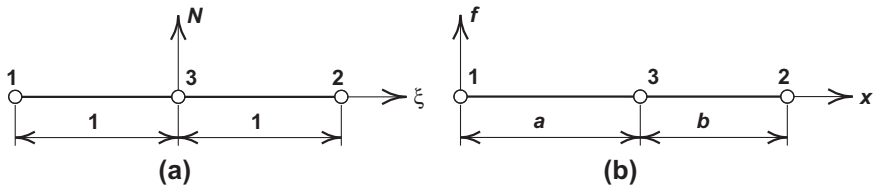



**FIGURE 6.35**
Quadratic element for Problem 6.16: (a) parent element and (b) mapped element.

**6.18** It is proposed to create transition elements for use with four-node quadrilateral element meshes as shown in Fig. 6.36:

- **(a)** Devise the shape functions for the transition element labeled $A$. The shape functions must maintain compatibility along all boundaries. Hint: The element can be a composite form combining more than one four-node element.
- **(b)** Devise the shape functions for the transition element labeled $B$.
- **(c)** On a sketch show the location of quadrature points necessary to integrate each element form.
- **(d)** As an alternative to transition elements, four-node elements may be used for all elements and constraints imposed to maintain compatibility. For the mesh shown in the figure, number all nodes and write the constraint equations necessary to maintain compatibility. The interior node of element $B$ is not needed and can be ignored.

**6.19** Determine the hierarchical interpolation functions in $\xi, \eta$ coordinates for the 16-node cubic order quadrilateral shown in Fig. 6.37a. Express the hierarchic shape functions in a form such that interpolation is given by

$$f(\xi,\eta) = \sum_{a=1}^{4} N_a(\xi,\eta) f_a + \sum_{a=5}^{12} N_a(\xi,\eta)\Delta f_a + \sum_{a=13}^{16} N_a(\xi,\eta)\Delta\Delta f_a$$

**6.20** Determine the hierarchical interpolation functions in $L_1, L_2, L_3$ area coordinates for the 10-node cubic order triangle shown in Fig. 6.37b. Express your

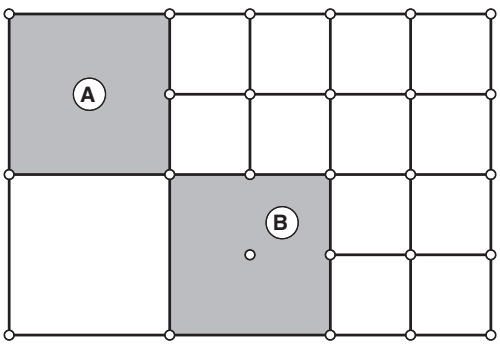



**FIGURE 6.36**
Transition elements for use with four-node quadrilaterals.

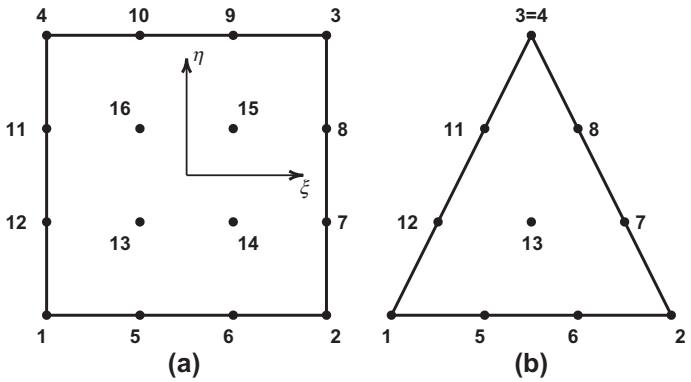



**FIGURE 6.37**
Degeneration of cubic triangle for Problem 6.20: (a) 16-node quadrilateral and (b) 10-node triangle.

hierarchic shape functions in a form such that interpolation is given by

$$f(L_1, L_2, L_3) = \sum_{a=1}^{3} N_a(L_1, L_2, L_3) f_a + \sum_{a=5}^{8} N_a(L_1, L_2, L_3) \Delta f_a + \sum_{a=11}^{12} N_a(L_1, L_2, L_3) \Delta f_a + N_{13}(L_1, L_2, L_3) \Delta\Delta f_{13}$$

**6.21** Using the shape functions developed in Problem 6.19, determine the modified shape functions to degenerate the cubic 16-node quadrilateral into the cubic 10-node triangular element using numbering as shown in Fig. 6.37. The final element must be completely consistent with the shape functions developed in Problem 6.20.

**6.22** Degenerate an eight-node hexahedral element to form a pyramid form with a rectangular base. Write the resulting shape functions for the remaining five nodes.

**6.23** For the triangular element shown in Fig. 6.38 show that the global coordinates may be expressed in local coordinates as

$$\mathbf{x} = \sum_{a=1}^{6} N_a(L_b)\, \tilde{\mathbf{x}}_a = 12\, L_2 + 18\, L_3$$

**6.24** For the triangular element shown in Fig. 6.38 compute the integrals $\int_\Delta N_2\, N_3\, \mathrm{d}\Delta$ and $\int_\Delta N_2\, N_4\, \mathrm{d}\Delta$ using:

**(a)** Eq. (6.47)
**(b)** An appropriate numerical integration using Table 6.3

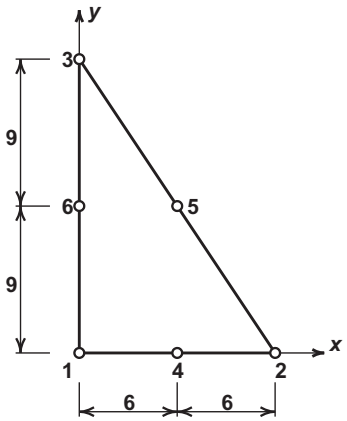



**FIGURE 6.38**
Quadratic triangle for Problems 6.23–6.25.

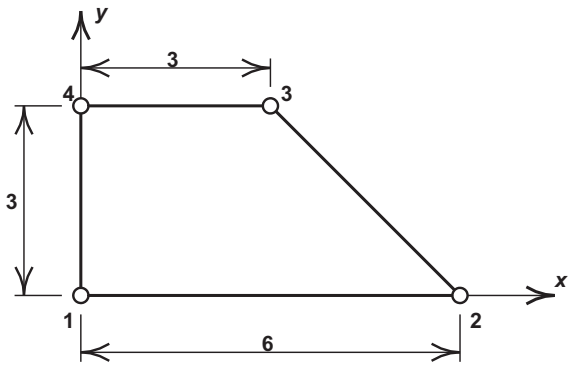



**FIGURE 6.39**
Quadrilateral for Problem 6.26.

**6.25** For the triangular element shown in Fig. 6.38 compute the integrals $\int_\Delta N_a \mathrm{d}\Delta$; $a = 1, 2, \ldots, 6$ using:

- **(a)** Eq. (6.47)
- **(b)** An appropriate numerical integration using Table 6.3

**6.26** The four-node quadrilateral element shown in Fig. 6.39 is used in the solution of a problem in which the dependent variable is a scalar, $u$:

- **(a)** Write the expression for an isoparametric mapping of coordinates in the element.

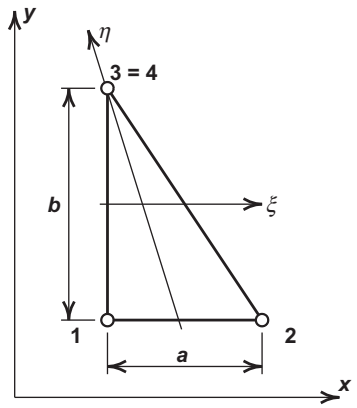



**FIGURE 6.40**
Degenerate triangle for Problem 6.27.

**(b)** Determine the location of the natural coordinates $\xi$ and $\eta$ which define the centroid of the element.
**(c)** Compute the expression for the Jacobian transformation $\mathbf{J}$ of the element. Evaluate the Jacobian at the centroid.
**(d)** Compute the derivatives of the shape function $N_3$ at the centroid.

**6.27** A triangular element is formed by degenerating a four-node quadrilateral element as shown in Fig. 6.40. If node 1 is located at $(x, y) = (10, 8)$ and the sides are $a = 20$ and $b = 30$:

**(a)** Write the expressions for $x$ and $y$ in terms of $\xi$ and $\eta$.
**(b)** Compute the Jacobian $\mathbf{J}(\xi, \eta)$ for the element.
**(c)** Compute the Jacobian determinant $J(\xi, \eta)$.
**(d)** For a one-point quadrature formula given by

$$I = \int_{-1}^{1}\int_{-1}^{1} f(\xi, \eta)\, \mathrm{d}\xi\, \mathrm{d}\eta = f(\xi_i, \eta_i) W_i$$

determine the values of $W_i$, $\xi_i$, and $\eta_i$ which exactly integrate the Jacobian $J$ (and thus also any integral which is a constant times the Jacobian).
**(e)** Is this the same point in the element as that using triangular coordinates $L_a$ and the one-point formula from Table 6.3? If not, why?

**6.28** In some instances it is desirable to perform numerical integration in which quadrature points are located at the end points as well as at interior points. One such formula is Gauss-Lobatto quadrature expressed as

$$\int_{-1}^{1} f(\xi)\, \mathrm{d}\xi = [f(-1) + f(1)]W_0 + \sum_{1}^{n} f(\xi_n) W_n$$

Determine the location of the points $\xi_n$ and the value of the weights $W_n$ which exactly integrate the highest polynomial of $f$ possible. Consider:

**(a)** The three-point formula ($n = 1$)
**(b)** The four-point formula ($n = 2$)

**6.29** Write the blending function mapping for a two-dimensional quadrilateral region which has one circular edge and three straight linear edges. Make a clear sketch of the region defined by the function and a $3 \times 3$ division into four-node quadrilateral elements.

**6.30** Consider a six-node triangular element with straight edges in which two of the mid-side nodes are placed at the quarter point. Show that the interpolation along the edge produces a derivative which varies as $1/\sqrt{r}$ where $r$ is the distance measured from the vertex.

**6.31** Compute the $x$ and $y$ derivatives for the shape function of nodes 1, 7, and 10 of the quadratic triangular prism shown in Fig. 6.16b.

**6.32** Program development project: Extend the program system started in Problem 5.15 to permit mesh generation using as input a four-node isoparametric block and mapping as described in Section 6.5. The input data should be the coordinates of the block vertices and the number of subdivisions in each direction.
Include as an option generation of coordinates in $r, \theta$ coordinates that are then transformed to $x, y$ Cartesian form.
Hint: Once coordinates for all node points are specified, MATLAB can generate a node connection list for three-node triangles using `DELAUNAY`.[4] A plot of the mesh may be produced using `TRIMESH`.
Use your program to generate a mesh for the rectangular beam described in Example 2.3 and the curved beam described in Example 2.4. Note the random orientation of diagonals which is associated with degeneracy in the Delaunay algorithm (viz. Chapter 17).

**6.33** Program development project: Extend the mesh generation scheme developed in Problem 6.32 to permit specification of the block as a blending function. Only allow two cases: (i) Lagrange interpolation which is linear or quadratic; (ii) circular arcs with specified radius and end points.
Test your program for the beam problems described in Examples 2.3 and 2.4.

## References

[1] August Möbius, Der barycentrische, Calcul (1827), vol. 1, pp. 1–388 (Verlag von S. Hirzel, Leipzig, 1885).
[2] B.M. Irons, J.G. Ergatoudis, O.C. Zienkiewicz, Comments on 'complete polynomial displacement fields for finite element method' (by P.C. Dunne), Trans. Roy. Aeronaut. Soc. 72 (1968) 709.

[4] In Chapter 17 we discuss mesh generation and some of the difficulties encountered with the Delaunay method.

[3] J.H. Argyris, I. Fried, D.W. Scharpf, The TET 20 and TEA 8 elements for the matrix displacement method, Aeronaut. J. 72 (1968) 618–625.
[4] R.L. Taylor, On completeness of shape functions for finite element analysis, Int. J. Numer. Methods Eng. 4 (1972) 17–22.
[5] P. Silvester, Higher order polynomial triangular finite elements for potential problems, Int. J. Eng. Sci. 7 (1969) 849–861.
[6] B. Fraeijs de Veubeke, Displacement and equilibrium models in finite element method, in: O.C. Zienkiewicz, G.S. Holister (Eds.), Stress Analysis, John Wiley & Sons, Chichester, 1965, pp. 145–197 (Chapter 9).
[7] J.H. Argyris, Triangular elements with linearly varying strain for the matrix displacement method, J. Roy. Aeronaut. Soc. Tech. Note 69 (1965) 711–713.
[8] J.G. Ergatoudis, B.M. Irons, O.C. Zienkiewicz, Curved, isoparametric, 'quadrilateral' elements for finite element analysis, Int. J. Solids Struct. 4 (1968) 31–42.
[9] O.C. Zienkiewicz, B.M. Irons, J.G. Ergatoudis, S. Ahmad, F.C. Scott, Isoparametric and associated elements families for two and three dimensional analysis, in: Finite Element Methods in Stress Analysis, Tapir Press, Trondheim, 1969 (Chapter 13).
[10] J.H. Argyris, K.E. Buck, H.M. Hilber, G. Mareczek, D.W. Scharpf, Some new elements for matrix displacement methods, in: Proceedings of the Second Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-68-150, Wright Patterson Air Force Base, Ohio, October 1968.
[11] O.C. Zienkiewicz, B.M. Irons, J. Campbell, F.C. Scott, Three dimensional stress analysis, in: IUTAM Symposium on High Speed Computing in Elasticity, Li'ege, 1970.
[12] F.C. Scott, A quartic, two-dimensional isoparametric element, Undergraduate Project, University of Wales, 1968.
[13] J.G. Ergatoudis, B.M. Irons, O.C. Zienkiewicz, Three dimensional analysis of arch dams and their foundations, in: Proceedings of the Symposium on Arch Dams, Institution of Civil Engineers, London, 1968.
[14] I.C. Taig, Structural analysis by the matrix displacement method, Technical Report, Report No. S.0.17, English Electric Aviation Ltd., April 1962. Based on work performed 1957–1958.
[15] E.L. Wachspress, High order curved finite elements, Int. J. Numer. Methods Eng. 17 (1981) 735–745.
[16] M. Crochet, Personal communication, 1988.
[17] Nam-Sua Lee, K.-J. Bathe, Effects of element distortion on the performance of isoparametric elements, Int. J. Numer. Methods Eng. 36 (1993) 3553–3576.
[18] R. Radau, Études sur les formules d'approximation qui servent à calculer la valeur d'une intégrale définie, J. Math. Pures Appl. 6 (1880) 283–336.
[19] R.G. Anderson, B.M. Irons, O.C. Zienkiewicz, Vibration and stability of plates using finite elements, Int. J. Solids Struct. 4 (1968) 1033–1055.
[20] P.C. Hammer, O.P. Marlowe, A.H. Stroud, Numerical integration over symplexes and cones, Math. Tables Aids Comput. 10 (1956) 130–137.
[21] C.A. Felippa, Refined finite element analysis of linear and non-linear two-dimensional structures, Ph.D. dissertation, Department of Civil Engineering,

SEMM, University of California, Berkeley, 1966. Also SEL Report 66-22, Structures Materials Research Laboratory.
[22] M. Abramowitz, I.A. Stegun (Eds.), Handbook of Mathematical Functions, Dover Publications, New York, 1965.
[23] G.R. Cowper, Gaussian quadrature formulas for triangles, Int. J. Numer. Methods Eng. 7 (1973) 405–408.
[24] J.N. Lyness, D. Jespersen, Moderate degree symmetric quadrature rules for the triangle, J. Inst. Math. Appl. 15 (1975) 19–32.
[25] B.M. Irons, Engineering applications of numerical integration in stiffness methods, J. AIAA 4 (1966) 2035–2037.
[26] G. Strang, G.J. Fix, An Analysis of the Finite Element Method, Prentice-Hall, Englewood Cliffs, NJ, 1973.
[27] I. Fried, Accuracy and condition of curved (isoparametric) finite elements, J. Sound Vibr. 31 (1973) 345–355.
[28] I. Fried, Numerical integration in the finite element method, Comput. Struct. 4 (1974) 921–932.
[29] M. Zlamal, Curved elements in the finite element method, SIAM J. Numer. Anal. 11 (1974) 347–362.
[30] D. Kosloff, G.A. Frasier, Treatment of hour glass patterns in low order finite element codes, Int. J. Numer. Anal. Methods Geomech. 2 (1978) 57–72.
[31] T. Belytschko, W.E. Bachrach, The efficient implementation of quadrilaterals with high coarse mesh accuracy, Comput. Methods Appl. Mech. Eng. 54 (1986) 276–301.
[32] R.E. Newton, Degeneration of brick-type isoparametric elements, Int. J. Numer. Methods Eng. 7 (1974) 579–581.
[33] B.M. Irons, A technique for degenerating brick type isoparametric elements using hierarchical midside nodes, Int. J. Numer. Methods Eng. 8 (1974) 209–211.
[34] O.C. Zienkiewicz, D.V. Phillips, An automatic mesh generation scheme for plane and curved surfaces by isoparametric coordinates, Int. J. Numer. Methods Eng. 3 (1971) 519–528.
[35] W.J. Gordon, C.A. Hall, Construction of curvilinear co-ordinate systems and application to mesh generation, Int. J. Numer. Methods Eng. 3 (1973) 461–477.
[36] W.J. Gordon, C.A. Hall, Transfinite element methods – blending-function interpolation over arbitrary curved element domains, Numer. Math. 21 (1973) 109–129.
[37] L. Piegl, W. Tiller, The NURBS Book (Monographs in Visual Communication), second ed., Springer-Verlag, New York, 1997.
[38] G. Farin, Curves and Surfaces for CAGD: A Practical Guide, fifth ed., Morgan Kaufmann Publishers, San Francisco, CA, 2002.
[39] D.F. Rogers, An Introduction to NURBS: With Historical Perspective, Academic Press, San Diego, 2000.
[40] B.M. Irons, Economical computer techniques for numerically integrated finite elements, Int. J. Numer. Methods Eng. 1 (1969) 201–203.

This page is intentionally left blank

CHAPTER

# 7 Elasticity: Two- and Three-Dimensional Finite Elements

## 7.1 Introduction

In Chapter 2 the reader was shown in some detail how linear elasticity and other problems could be formulated as differential equations. In Chapter 3 one-dimensional integral forms were developed and solved using simple finite element approximations. The isoparametric concept and numerical integration were introduced to permit direct numerical calculations. Although the detailed algebra was only concerned with shape functions which arose from one-dimensional forms, it should by now be obvious that once integral weak forms or variational principles are known and appropriate shape functions devised general element forms can equally well be used. Indeed, we will observe that once the integral form together with the element type and its corresponding shape functions are determined, subsequent operations follow a standard, well-defined path as we demonstrated in Chapter 5 for quasi-harmonic problems.

In this chapter we consider the two- and three-dimensional elasticity problems described in Chapter 2 and develop all the necessary steps for a finite element analysis of general solids in terms of a *displacement method* approximation which for the elasticity problem gives the *irreducible form*. In later chapters we will consider other forms in which direct approximation of stress and strain can be employed to achieve better behavior for some classes of problems.

## 7.2 Elasticity problems: Weak form for equilibrium

We begin by considering the weak form for three-dimensional problems in elasticity. The weak form of the equilibrium equations for linear elasticity may be written using variational notation as

$$\mathrm{G}(\delta\mathbf{u},\mathbf{u},\boldsymbol{\sigma}) = \int_{\Omega} \delta\mathbf{u}^{\mathrm{T}} \left(\rho\ddot{\mathbf{u}} - \mathbf{b} - \boldsymbol{\mathcal{S}}^{\mathrm{T}}\boldsymbol{\sigma}\right) \mathrm{d}\Omega = 0 \tag{7.1}$$

where the term within the integral arises from the first two steps of (3.1). Expanding the equations for the three-dimensional problem in Cartesian coordinates gives

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00007-1

$$\mathrm{G}(\delta\mathbf{u},\mathbf{u},\boldsymbol{\sigma}) = \int_\Omega \begin{Bmatrix}\delta u\\ \delta v\\ \delta w\end{Bmatrix}^{\mathrm{T}} \left( \rho \begin{Bmatrix}\ddot{u}\\ \ddot{v}\\ \ddot{w}\end{Bmatrix} - \begin{Bmatrix}b_x\\ b_y\\ b_z\end{Bmatrix} - \begin{Bmatrix} \dfrac{\partial\sigma_x}{\partial x} + \dfrac{\partial\tau_{yx}}{\partial y} + \dfrac{\partial\tau_{zx}}{\partial z}\\ \dfrac{\partial\tau_{xy}}{\partial x} + \dfrac{\partial\sigma_y}{\partial y} + \dfrac{\partial\tau_{zy}}{\partial z}\\ \dfrac{\partial\tau_{xz}}{\partial x} + \dfrac{\partial\tau_{yz}}{\partial y} + \dfrac{\partial\sigma_z}{\partial z}\end{Bmatrix} \right) \mathrm{d}\Omega = 0 \tag{7.2}$$

For the three-dimensional problem in Cartesian coordinates the differential volume is given by $\mathrm{d}\Omega = \mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z$. Integrating the terms with derivatives of stresses using Green's theorem (see Appendix F) yields

$$\begin{aligned}
&\int_\Omega \begin{Bmatrix}\delta u\\ \delta v\\ \delta w\end{Bmatrix}^{\mathrm{T}} \begin{Bmatrix} \dfrac{\partial\sigma_x}{\partial x} + \dfrac{\partial\tau_{yx}}{\partial y} + \dfrac{\partial\tau_{zx}}{\partial z}\\ \dfrac{\partial\tau_{xy}}{\partial x} + \dfrac{\partial\sigma_y}{\partial y} + \dfrac{\partial\tau_{zy}}{\partial z}\\ \dfrac{\partial\tau_{xz}}{\partial x} + \dfrac{\partial\tau_{yz}}{\partial y} + \dfrac{\partial\sigma_z}{\partial z}\end{Bmatrix} \mathrm{d}\Omega \\
&= \int_\Gamma \begin{Bmatrix}\delta u\\ \delta v\\ \delta w\end{Bmatrix}^{\mathrm{T}} \begin{bmatrix} n_x & 0 & 0 & n_y & 0 & n_z\\ 0 & n_y & 0 & n_x & n_z & 0\\ 0 & 0 & n_z & 0 & n_y & n_x\end{bmatrix} \begin{Bmatrix}\sigma_x\\ \sigma_y\\ \sigma_z\\ \tau_{xy}\\ \tau_{yz}\\ \tau_{zx}\end{Bmatrix} \mathrm{d}\Gamma \\
&- \int_\Omega \begin{bmatrix} \dfrac{\partial\delta u}{\partial x} & 0 & 0 & \dfrac{\partial\delta u}{\partial y} & 0 & \dfrac{\partial\delta u}{\partial z}\\ 0 & \dfrac{\partial\delta v}{\partial y} & 0 & \dfrac{\partial\delta v}{\partial x} & \dfrac{\partial\delta v}{\partial z} & 0\\ 0 & 0 & \dfrac{\partial\delta w}{\partial z} & 0 & \dfrac{\partial\delta w}{\partial y} & \dfrac{\partial\delta w}{\partial x}\end{bmatrix} \begin{Bmatrix}\sigma_x\\ \sigma_y\\ \sigma_z\\ \tau_{xy}\\ \tau_{yz}\\ \tau_{zx}\end{Bmatrix} \mathrm{d}\Omega
\end{aligned} \tag{7.3}$$

where $\mathrm{d}\Gamma = \mathrm{d}A$ for which $\mathrm{d}A$ is a differential area of the boundary surface. In the above we have also used balance of angular momentum ($\tau_{xy} = \tau_{yx}$, etc.). Similar to the one-dimensional axisymmetric problem considered in Section 3.7 we note that

$$\begin{bmatrix} \dfrac{\partial\delta u}{\partial x} & 0 & 0\\ 0 & \dfrac{\partial\delta v}{\partial y} & 0\\ 0 & 0 & \dfrac{\partial\delta w}{\partial z}\\ \dfrac{\partial\delta u}{\partial y} & \dfrac{\partial\delta v}{\partial x} & 0\\ 0 & \dfrac{\partial\delta v}{\partial z} & \dfrac{\partial\delta w}{\partial y}\\ \dfrac{\partial\delta u}{\partial z} & 0 & \dfrac{\partial\delta w}{\partial x}\end{bmatrix} = \boldsymbol{\mathcal{S}}\,\delta\mathbf{u} = \delta\boldsymbol{\varepsilon} \tag{7.4}$$

and thus it is merely the variation of the strain expressed as derivatives of displacement. We recall that this is identical to the notion of a virtual displacement in a virtual work form. The recovery of virtual strains in a weak form of equilibrium is fundamental for all types of solid mechanics formulations—including plate and shell problems considered in later chapters. We also note the boundary integral term may be written as

$$\int_\Gamma [\,\delta u \quad \delta v \quad \delta w\,] \begin{bmatrix} n_x & 0 & 0 & n_y & 0 & n_z \\ 0 & n_y & 0 & n_x & n_z & 0 \\ 0 & 0 & n_z & 0 & n_y & n_x \end{bmatrix} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{Bmatrix} \mathrm{d}\Gamma = \int_\Gamma \delta\mathbf{u}^{\mathrm{T}}\,\mathbf{G}^{\mathrm{T}}\boldsymbol{\sigma}\,\mathrm{d}\Gamma = \int_\Gamma \delta\mathbf{u}^{\mathrm{T}}\,\mathbf{t}\,\mathrm{d}\Gamma \tag{7.5}$$

where $\mathbf{t} = \mathbf{G}^{\mathrm{T}}\boldsymbol{\sigma}$ defines the boundary traction vector [viz. (2.26)]. If we split the boundary into two parts that define where tractions and displacement boundary conditions are defined, we may insert specified traction boundary conditions as

$$\int_\Gamma \delta\mathbf{u}^{\mathrm{T}}\,\mathbf{t}\,\mathrm{d}\Gamma = \int_{\Gamma_u} \delta\mathbf{u}^{\mathrm{T}}\,\mathbf{t}\,\mathrm{d}\Gamma + \int_{\Gamma_t} \delta\mathbf{u}^{\mathrm{T}}\,\mathbf{t}\,\mathrm{d}\Gamma = \int_{\Gamma_u} \delta\mathbf{u}^{\mathrm{T}}\,\mathbf{t}\,\mathrm{d}\Gamma + \int_{\Gamma_t} \delta\mathbf{u}^{\mathrm{T}}\,\bar{\mathbf{t}}\,\mathrm{d}\Gamma \tag{7.6}$$

Generally, as described in Chapter 3, we shall impose the displacement boundary condition $\mathbf{u} = \bar{\mathbf{u}}$ directly and set $\delta\mathbf{u} = \mathbf{0}$ on $\Gamma_u$. Thus, the weak form for the linear elastic equilibrium equations may be written compactly in matrix form as

$$\mathrm{G}(\delta\mathbf{u}, \mathbf{u}, \boldsymbol{\sigma}) = \int_\Omega \left[\delta\mathbf{u}^{\mathrm{T}}\left(\rho\ddot{\mathbf{u}} - \mathbf{b}\right) + \delta\boldsymbol{\varepsilon}^{\mathrm{T}}\boldsymbol{\sigma}\right]\mathrm{d}\Omega - \int_{\Gamma_t} \delta\mathbf{u}^{\mathrm{T}}\bar{\mathbf{t}}\,\mathrm{d}\Gamma = 0 \tag{7.7}$$

where $\delta\boldsymbol{\varepsilon} = \boldsymbol{\mathcal{S}}\,\delta\mathbf{u}$.

Equation (7.7) also may be used for two-dimensional plane and axisymmetric problems by appropriately defining the domain and boundary along with the displacement, stress, and strain components as given in Chapter 2. For example, the axisymmetric form has displacements $u(r, z)$, $v(r, z)$ in the domain $\mathrm{d}\Omega = 2\pi r\,\mathrm{d}r\,\mathrm{d}z$ with boundary surface $\mathrm{d}\Gamma = 2\pi r\,\mathrm{d}s$ where $\mathrm{d}s$ is an increment of the boundary curve. The stress and strain components are

$$\boldsymbol{\sigma}^{\mathrm{T}} = [\,\sigma_r,\ \sigma_z,\ \sigma_\theta,\ \tau_{rz}\,] \quad \text{and} \quad \boldsymbol{\varepsilon}^{\mathrm{T}} = [\,\varepsilon_r,\ \varepsilon_z,\ \varepsilon_\theta,\ \gamma_{rz}\,] \tag{7.8}$$

and the strain-displacement operator $\boldsymbol{\mathcal{S}} = \boldsymbol{\mathcal{S}}_a$ as defined in (2.10). A similar substitution may be used to define the plane strain and plane stress forms.

### 7.2.1 Displacement method: Irreducible form

The weak form of linear elasticity may be written completely in terms of displacements by substituting (2.46a) into (7.7) and using (2.8) to compute strains. The result is

$$\begin{aligned} \mathrm{G}(\delta\mathbf{u}, \mathbf{u}) = &\int_\Omega \delta\mathbf{u}^{\mathrm{T}} \left(\rho\ddot{\mathbf{u}} - \mathbf{b}\right) \mathrm{d}\Omega + \int_\Omega \delta\boldsymbol{\varepsilon}^{\mathrm{T}} \left[\mathbf{D}\left(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_0\right) + \boldsymbol{\sigma}_0\right] \Omega \\ &- \int_{\Gamma_t} \delta\mathbf{u}^{\mathrm{T}} \bar{\mathbf{t}} \, \mathrm{d}\Gamma = 0 \end{aligned} \tag{7.9}$$

where it is understood also that strains are expressed in terms of displacement derivatives. This is called the *displacement method* or *irreducible form* of the problem since it is only necessary to define the distribution of the displacement field and its variation to solve a problem. As illustrated for the one-dimensional problem in Chapter 3 one may use an assumed form for the displacement field expressed as

$$\begin{aligned} \mathbf{u}(\mathbf{x}, t) &= \sum_i \phi_i(\mathbf{x}) \, \tilde{\mathbf{u}}_i(t) + \bar{\mathbf{u}}_b(\mathbf{x}, t) \\ \delta\mathbf{u}(\mathbf{x}) &= \sum_i \phi_i(\mathbf{x}) \, \delta\tilde{\mathbf{u}}_i \end{aligned} \tag{7.10}$$

where $\bar{\mathbf{u}}_b$ is introduced to satisfy the displacement boundary condition. To represent the functions $\phi_i$ in a viable manner for multidimensional problems we again use a finite element approach and employ the shape functions described in Chapter 6.

## 7.3 Finite element approximation by the Galerkin method

The functional in (7.9) may be used to construct approximate solutions by inserting appropriate finite element trial and weight functions. Here, since the static problem is self-adjoint, we shall assume that the same functions are used to approximate the displacements $\mathbf{u}$ and virtual displacements $\delta\mathbf{u}$ as indicated in (7.10). To construct the finite element approximation, we shall assume that the domain of analysis is divided into a set of elements and nodes and express the integrals as

$$\int_\Omega (\cdot) \, \mathrm{d}\Omega = \sum_e \int_{\Omega_e} (\cdot) \, \mathrm{d}\Omega \quad \text{and} \quad \int_\Gamma (\cdot) \, \mathrm{d}\Gamma = \sum_e \int_{\Gamma_e} (\cdot) \, \mathrm{d}\Gamma \tag{7.11}$$

This basic structure is fundamental to all finite element methods. Since the strains involve first derivatives of displacement components, the approximation to displacement components need only to be continuous in $\Omega$. A similar result holds for the relationship between the variation of strain and the variation of displacement. Thus, for all the multidimensional elasticity problems we only require $C_0$ continuous functions of the type presented in the previous chapter to approximate the displacement and virtual displacement. Accordingly, we assume that the displacements are given

in each element by

$$\mathbf{u} \approx \hat{\mathbf{u}} = \sum_a N_a(\mathbf{x})\, \tilde{\mathbf{u}}_a^e(t) \tag{7.12}$$

with a similar expression for the variation of displacement. Using this form we are able to satisfy both the weak form and, using some of the nodal parameters, the displacement boundary conditions on $\Gamma_u$.

Inserting the approximation for displacements into an approximation of the weak form yields

$$\mathrm{G}(\delta\mathbf{u}, \mathbf{u}) \approx \widehat{\mathrm{G}}(\delta\hat{\mathbf{u}}, \hat{\mathbf{u}}) = \sum_e \widehat{\mathrm{G}}^e(\delta\hat{\mathbf{u}}, \hat{\mathbf{u}}) = 0 \tag{7.13}$$

The expression for each element domain $\Omega_e$ is given by

$$\begin{aligned} \widehat{\mathrm{G}}^e(\delta\hat{\mathbf{u}}, \hat{\mathbf{u}}) = &\int_{\Omega_e} \delta\hat{\mathbf{u}}^{\mathrm{T}} \left(\rho\ddot{\hat{\mathbf{u}}} - \mathbf{b}\right) \mathrm{d}\Omega + \int_{\Omega_e} \delta\hat{\boldsymbol{\varepsilon}}^{\mathrm{T}} \left[\mathbf{D}\left(\hat{\boldsymbol{\varepsilon}} - \boldsymbol{\varepsilon}_0\right) + \boldsymbol{\sigma}_0\right] \mathrm{d}\Omega \\ &- \int_{\Gamma_{et}} \delta\hat{\mathbf{u}}^{\mathrm{T}} \bar{\mathbf{t}}\, \mathrm{d}\Gamma \end{aligned} \tag{7.14a}$$

in which strains are expressed as

$$\begin{aligned} \delta\hat{\boldsymbol{\varepsilon}} &= \sum_a \boldsymbol{\mathcal{S}}(N_a)\, \delta\tilde{\mathbf{u}}_a^e = \sum_a \mathbf{B}_a\, \delta\tilde{\mathbf{u}}_a^e \\ \hat{\boldsymbol{\varepsilon}} &= \sum_b \boldsymbol{\mathcal{S}}(N_b)\, \tilde{\mathbf{u}}_b^e = \sum_b \mathbf{B}_b\, \tilde{\mathbf{u}}_b^e \end{aligned} \tag{7.14b}$$

with $\mathbf{B}_a$ the three-dimensional *strain-displacement matrix*

$$\mathbf{B}_a = \begin{bmatrix} \dfrac{\partial N_a}{\partial x} & 0 & 0 \\ 0 & \dfrac{\partial N_a}{\partial y} & 0 \\ 0 & 0 & \dfrac{\partial N_a}{\partial z} \\ \dfrac{\partial N_a}{\partial y} & \dfrac{\partial N_a}{\partial x} & 0 \\ 0 & \dfrac{\partial N_a}{\partial z} & \dfrac{\partial N_a}{\partial y} \\ \dfrac{\partial N_a}{\partial z} & 0 & \dfrac{\partial N_a}{\partial x} \end{bmatrix} \tag{7.15}$$

Evaluating the weak form for an individual element gives

$$\widehat{\mathrm{G}}^e = \sum_a \delta\tilde{\mathbf{u}}_a^{e\mathrm{T}} \left[\sum_b \left(\mathbf{M}_{ab}^e \ddot{\tilde{\mathbf{u}}}_b^e + \mathbf{K}_{ab}^e\, \tilde{\mathbf{u}}_b^e\right) - \mathbf{f}_a^e\right] \tag{7.16a}$$

where the element mass, stiffness, and load matrices are computed from

$$\mathbf{M}_{ab}^e = \int_{\Omega_e} N_a\, \rho\, N_b\, \mathrm{d}\Omega\, \mathbf{I}$$

$$\mathbf{K}^e_{ab} = \int_{\Omega_e} \mathbf{B}_a^{\mathrm{T}}\, \mathbf{D}\, \mathbf{B}_b \, \mathrm{d}\Omega$$

$$\mathbf{f}^e_a = \int_{\Omega_e} N_a\, \mathbf{b}\, \mathrm{d}\Omega + \int_{\Gamma_{et}} N_a\, \bar{\mathbf{t}}\, \mathrm{d}\Gamma + \int_{\Omega_e} \mathbf{B}_a^{\mathrm{T}} \left(\mathbf{D}\, \boldsymbol{\varepsilon}_0 - \boldsymbol{\sigma}_0\right) \mathrm{d}\Omega \tag{7.16b}$$

In this form we again obtain the standard discrete system described in Chapter 1 in which we assemble the total problem from the contribution of each element. After assembly of all elements the weak form for the approximation is specified by

$$\widehat{\mathrm{G}} = \delta \tilde{\mathbf{u}}_a^{\mathrm{T}} \left[ \mathbf{M}_{ab}\, \ddot{\tilde{\mathbf{u}}}_b + \mathbf{K}_{ab}\, \tilde{\mathbf{u}}_b - \mathbf{f}_a \right] = 0$$

and since $\delta \tilde{\mathbf{u}}_a$ is arbitrary the governing equation of the problem becomes

$$\mathbf{M}_{ab}\, \ddot{\tilde{\mathbf{u}}}_b + \mathbf{K}_{ab}\, \tilde{\mathbf{u}}_b = \mathbf{f}_a \tag{7.17a}$$

with

$$\mathbf{M}_{ab} = \sum_e \mathbf{M}^e_{ab}, \qquad \mathbf{K}_{ab} = \sum_e \mathbf{K}^e_{ab} \qquad \text{and} \qquad \mathbf{f}_a = \sum_e \mathbf{f}^e_a \tag{7.17b}$$

### *Quasi-static problems*

For problems subjected to loading changes that are very slow, we can often assume that $\ddot{\tilde{\mathbf{u}}}_b \approx \mathbf{0}$ and solve the *quasi-static* problem

$$\sum_b \mathbf{K}_{ab}\, \tilde{\mathbf{u}}_b = \mathbf{f}_a \qquad \text{for all } a$$

or simply

$$\mathbf{K}\, \tilde{\mathbf{u}} = \mathbf{f} \tag{7.18}$$

which, formally after inserting the displacement boundary conditions, yields the solution for nodal displacements

$$\tilde{\mathbf{u}} = \mathbf{K}^{-1}\mathbf{f} \tag{7.19}$$

in which $\mathbf{K}^{-1}$ is the matrix inverse of $\mathbf{K}$.[1]

### *Strains: Two-dimensional case*

The strain-displacement matrix $\mathbf{B}_a$ for plane stress and plane strain may be determined from (7.15) by eliminating the dependence on the $w$ derivative and eliminating the last two shear strains. The result is

$$\mathbf{B}_a\, \tilde{\mathbf{u}}_a = \begin{bmatrix} \dfrac{\partial N_a}{\partial x}, & 0 \\ 0, & \dfrac{\partial N_a}{\partial y} \\ 0, & 0 \\ \dfrac{\partial N_a}{\partial y}, & \dfrac{\partial N_a}{\partial x} \end{bmatrix} \begin{Bmatrix} \tilde{u}_a \\ \tilde{v}_a \end{Bmatrix} \tag{7.20}$$

[1] In practice we do not invert $\mathbf{K}$; instead we solve Eq. (7.18) by a direct or iterative method (see Appendix D).

For the axisymmetric case the strains are given by (2.10) and the reader may verify that the strain matrix is given by

$$\mathbf{B}_a\,\tilde{\mathbf{u}}_a = \begin{bmatrix} \dfrac{\partial N_a}{\partial r} & 0 \\ 0 & \dfrac{\partial N_a}{\partial z} \\ \dfrac{N_a}{r} & 0 \\ \dfrac{\partial N_a}{\partial z} & \dfrac{\partial N_a}{\partial r} \end{bmatrix} \begin{Bmatrix} \tilde{u}_a \\ \tilde{v}_a \end{Bmatrix} \tag{7.21}$$

## 7.4 Boundary conditions

Many types of boundary conditions may exist, however, boundary conditions considered here for elasticity problems are of two types: (a) specified displacements and (b) specified tractions.

### *Displacement boundary conditions*

The displacement conditions specified at each point of the boundary are given by [viz. Section 2.2.5]

$$\mathbf{u}(\mathbf{x}, t) = \bar{\mathbf{u}}(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma_u \qquad \text{and} \qquad t \geq 0$$

where $\bar{\mathbf{u}}$ is a specified quantity. In a finite element problem it is necessary to approximate the displacement boundary condition. There are two aspects of the approximations. First, any curved boundary of the problem is approximated by the boundary mesh of elements as shown in Fig. 7.1 for a two-dimensional edge on which linear elements are used. The boundary of the finite element mesh is denoted as $\Gamma_u^h$. The geometric error between $\Gamma_u$ and $\Gamma_u^h$ is given by

$$e = O(h^p)$$

where $h$ is the characteristic element dimension and we note that for linear order shape functions $p = 2$.

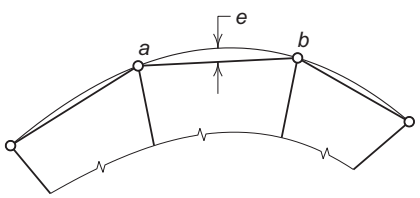



**FIGURE 7.1**

Boundary approximation: $e$ error between $\Gamma$ and $\Gamma^h$.

The second approximation is introduced by the interpolation used to define the shape functions. Thus, the displacement boundary condition becomes

$$\bar{\mathbf{u}}(\mathbf{x}, t) \approx \hat{\mathbf{u}}(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma_u^h \qquad \text{and} \qquad t \geq 0 \tag{7.22}$$

where $\hat{\mathbf{u}}$ denotes the finite element approximation.

We note that using an isoparametric form to define shape functions yields errors in the displacement approximation that are of identical order as the error in matching curved boundaries by isoparametric elements. Furthermore, it is not necessary to introduce any separate interpolation for $\bar{\mathbf{u}}_b$; one merely uses the finite element interpolation for nodes that are on the boundary $\Gamma_u^h$. Thus, an accurate approximation of (7.22) is a *collocation of the specified displacements at the nodes* as

$$\tilde{\mathbf{u}}_a(t) = \bar{\mathbf{u}}(\mathbf{x}_a, t), \quad \mathbf{x}_a \in \Gamma_u^h \qquad \text{and} \qquad t \geq 0 \tag{7.23}$$

A least squares fit to the boundary condition may also be used. However, for the type of interpolations used in this volume, no increase in accuracy is achieved. For other types of interpolations collocation may not be appropriate and a more accurate solution must be used in order to achieve best accuracy in a solution [1].

### *Traction boundary conditions*

Traction boundary conditions are also carried out over the approximation to the boundary $\Gamma_t$, which we denote by $\Gamma_t^h$. Thus, the contribution to the force vector for boundary tractions is computed from

$$\mathbf{f}_a = \int_{\Gamma_t^h} N_a \, \bar{\mathbf{t}} \, \mathrm{d}\Gamma \tag{7.24}$$

For three-dimensional problems, the integral may be transformed to parent coordinates by using two-dimensional shape functions to approximate the boundary surface. Accordingly we obtain

$$\mathbf{f}_a = \int_{-1}^{1} \int_{-1}^{1} N_a(\xi, \eta) \, \bar{\mathbf{t}} \, j(\xi, \eta) \, \mathrm{d}\xi \, \mathrm{d}\eta \tag{7.25a}$$

for quadrilateral surface facets and

$$\mathbf{f}_a = \int_0^1 \left[ \int_0^{1-L_2} N_a(L_1, L_2) \, \bar{\mathbf{t}} \, j(L_1, L_2) \, \mathrm{d}L_1 \right] \mathrm{d}L_2 \tag{7.25b}$$

for triangular surface facets. Generally the above integrals are best computed using numerical integration as described in Section 6.8.

For two-dimensional problems the above surface integrals become one-dimensional integrals along the edges of elements on the boundary approximation $\Gamma_t^h$.

### *Normal traction on surface*

A useful special case for a boundary traction is application of a normal traction loading by a "pressure" given as

$$\bar{\mathbf{t}} = \bar{p}\,\mathbf{n}$$

This is a particularly simple condition to impose on the boundary facet since, for three-dimensional quadrilateral faces we have

$$\mathbf{n}\,\mathrm{d}\Gamma = \left(\frac{\partial \mathbf{x}}{\partial \xi} \times \frac{\partial \mathbf{x}}{\partial \eta}\right) \mathrm{d}\xi\,\mathrm{d}\eta \tag{7.26}$$

In the above $\mathbf{a} \times \mathbf{b}$ denotes the vector (cross) product of the two vectors $\mathbf{a}$ and $\mathbf{b}$. The vector product also may be written in a matrix form as

$$\mathbf{a} \times \mathbf{b} = \hat{\mathbf{a}}\,\mathbf{b} \qquad \text{where } \hat{\mathbf{a}} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{7.27}$$

Using this result the nodal force for each quadrilateral boundary facet is computed from the matrix form

$$\mathbf{f}_a = \int_{-1}^{1}\int_{-1}^{1} N_a(\xi,\eta)\,\bar{p}\,\widehat{\frac{\partial \mathbf{x}}{\partial \xi}}\,\frac{\partial \mathbf{x}}{\partial \eta}\,\mathrm{d}\xi\,\mathrm{d}\eta \tag{7.28}$$

The reader can deduce the form for triangular facets from (7.25b). Quadrature may be used to evaluate the integrals for specific forms of $\bar{p}$.

For boundaries of two-dimensional problems the edge is a line and one-dimensional interpolation for shape functions as given in Chapter 3 may be used. In this case the nodal force for plane problems is computed from

$$\mathbf{f}_a = \int_{-1}^{1} N_a(\xi)\,\bar{p}\,\widehat{\frac{\partial \mathbf{x}}{\partial \xi}}\,\mathbf{e}_z\,\mathrm{d}\xi \tag{7.29}$$

where $\mathbf{e}_z$ is a unit normal in the $z$-coordinate direction. For axisymmetric problems the boundary force is computed from

$$\mathbf{f}_a = 2\pi \int_{-1}^{1} N_a(\xi)\,\bar{p}\,\widehat{\frac{\partial \mathbf{x}}{\partial \xi}}\,\mathbf{e}_\theta\,r(\xi)\,\mathrm{d}\xi \tag{7.30}$$

where $\mathbf{e}_\theta$ is a unit vector normal to the $rz$ plane.

## 7.5 Numerical integration and alternate forms

Generally, the evaluation of the integrals appearing in the arrays given in (7.16b) is performed using numerical quadrature as described in Section 6.8. Indeed it is

possible to prepare a single-element program module that carries out the evaluation of $\mathbf{M}^e_{ab}$, $\mathbf{K}^e_{ab}$, and $\mathbf{f}^e_a$ for each nodal pair.

The basic structure shown in Fig. 6.31 requires only a *quadrature module* to provide the quadrature points and weights and a *shape function module* to provide the matching $N_a$ and their first derivatives.

For applications in which the material parameters entering the arrays are constant in each element a formal expansion will show that products of shape functions and their derivatives appear multiple times and this permits a more efficient evaluation of the mass and stiffness matrices.

**Example 7.1.** Mass matrix evaluation

For a constant element density, $\rho^e$, the element mass array is given by

$$\mathbf{M}^e_{ab} = \rho^e \int_{\Omega_e} N_a \, N_b \, \mathrm{d}\Omega \, \mathbf{I}$$

and thus one only need compute the scalar products

$$M_{ab} = \int_{\Omega_e} N_a \, N_b \, \mathrm{d}\Omega$$

with

$$\mathbf{M}^e_{ab} = \rho^e \, M_{ab} \, \mathbf{I}$$

**Example 7.2.** Stiffness matrix evaluation

For the element stiffness evaluation even greater simplification occurs. For problems expressed in Cartesian coordinates with $\mathbf{D}$ constant in each element the integrals are all of the form

$$\mathbf{W}^{ab} = \int_{\Omega_e} \begin{bmatrix} \dfrac{\partial N_a}{\partial x}\dfrac{\partial N_b}{\partial x} & \dfrac{\partial N_a}{\partial x}\dfrac{\partial N_b}{\partial y} & \dfrac{\partial N_a}{\partial x}\dfrac{\partial N_b}{\partial z} \\ \dfrac{\partial N_a}{\partial y}\dfrac{\partial N_b}{\partial x} & \dfrac{\partial N_a}{\partial y}\dfrac{\partial N_b}{\partial y} & \dfrac{\partial N_a}{\partial y}\dfrac{\partial N_b}{\partial z} \\ \dfrac{\partial N_a}{\partial z}\dfrac{\partial N_b}{\partial x} & \dfrac{\partial N_a}{\partial z}\dfrac{\partial N_b}{\partial y} & \dfrac{\partial N_a}{\partial z}\dfrac{\partial N_b}{\partial z} \end{bmatrix} \mathrm{d}\Omega$$

For a two-dimensional plane strain or plane stress problem the final element matrix is then given by

$$\mathbf{K}^e_{ab} = \begin{bmatrix} K^{ab}_{11} & K^{ab}_{12} \\ K^{ab}_{21} & K^{ab}_{22} \end{bmatrix}$$

where

$$\begin{aligned} K^{ab}_{11} &= D_{11} W^{ab}_{11} + D_{14} W^{ab}_{12} + D_{41} W^{ab}_{21} + D_{44} W^{ab}_{22} \\ K^{ab}_{12} &= D_{12} W^{ab}_{12} + D_{14} W^{ab}_{11} + D_{42} W^{ab}_{22} + D_{44} W^{ab}_{21} \\ K^{ab}_{21} &= D_{21} W^{ab}_{21} + D_{24} W^{ab}_{22} + D_{41} W^{ab}_{11} + D_{44} W^{ab}_{12} \\ K^{ab}_{22} &= D_{22} W^{ab}_{22} + D_{24} W^{ab}_{21} + D_{42} W^{ab}_{12} + D_{44} W^{ab}_{11} \end{aligned}$$

The savings in computer time to compute the array can be quite significant. However, for large problems the overall gain is modest due to the time needed to solve the large set of algebraic equations [e.g., Eq. (7.18)].

## 7.6 Infinite domains and infinite elements

In many problems of engineering and physics infinite or semi-infinite domains exist. A typical example from structural mechanics may, for instance, be that of three-dimensional (or axisymmetric) excavation, illustrated in Fig. 7.2. Here the problem is one of determining the deformations in a semi-infinite half-space due to the removal of loads with the specification of zero displacements at infinity. Similar problems abound in electromagnetics and fluid mechanics but the situation illustrated is typical. The question arises as to how such problems can be dealt with by a method of approximation in which elements of decreasing size are used in the modeling process. The first intuitive answer is the one illustrated in Fig. 7.2a where the infinite boundary condition is specified at a finite boundary placed at a *large distance* from the object. This, however, begs the question of what constitutes a "large distance" and obviously substantial errors may arise if this boundary is not placed far enough away. On the other hand, pushing this out excessively far necessitates the introduction of a large number of elements to model regions of relatively little interest to the analyst.

To overcome such "infinite" difficulties many methods have been proposed. In some a sequence of nesting grids is used and a recurrence relation derived [2,3].

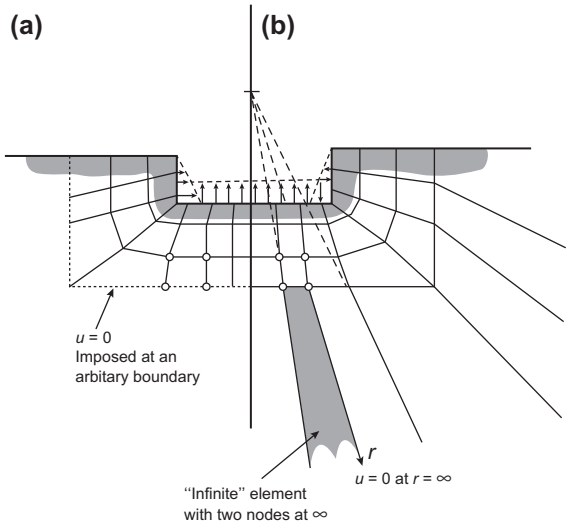



**FIGURE 7.2**

A semi-infinite domain. Deformations of a foundation due to removal of load following an excavation: (a) conventional treatment and (b) use of infinite elements.

In others a boundary-type exact solution is used and coupled to the finite element domain [4,5]. However, without doubt, the most effective and efficient treatment is the use of "infinite elements" [6–9] pioneered originally by Bettess [10]. In this process the conventional, finite elements are coupled to elements of the type shown in Fig. 7.2b which model in a reasonable manner the material stretching to infinity.

The shape of such two-dimensional elements and their treatment is best accomplished by mapping [8–10] these onto a finite square (or a finite line in one dimension or cube in three dimensions). However, it is essential that the sequence of trial functions introduced in the mapped domain be such that it is complete and capable of modeling the true behavior as the radial distance $r$ increases. Here it would be advantageous if the mapped shape functions could approximate a sequence of the decaying form

$$\frac{C_1}{r} + \frac{C_2}{r^2} + \frac{C_3}{r^3} + \cdots \tag{7.31}$$

where $C_a$ are arbitrary constants and $r$ is the radial distance from the "focus" of the problem.

### 7.6.1 The mapping function

Figure 7.3 illustrates the principles for generation of the derived mapping function.

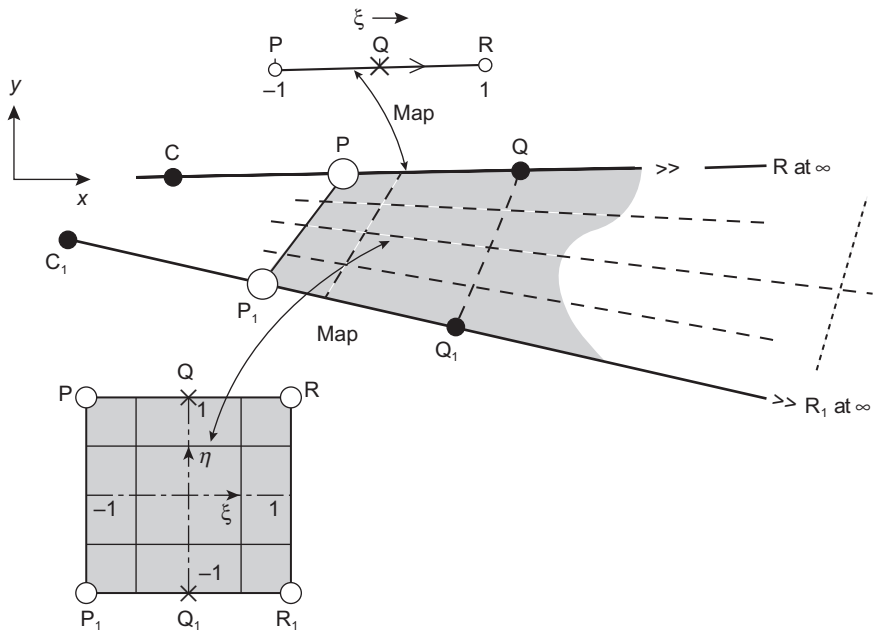



**FIGURE 7.3**
Infinite line and element map. Linear $\eta$ interpolation.

We start with a one-dimensional mapping along a line CPQ coinciding with the $x$ direction. Consider the function

$$x = -\frac{\xi}{1-\xi}x_C + \left(1 + \frac{\xi}{1-\xi}\right)x_Q = \bar{N}_C x_C + \bar{N}_Q x_Q \tag{7.32a}$$

and we immediately observe that

$$\begin{aligned} \xi &= -1 \quad \text{corresponds to } x = \frac{x_Q + x_C}{2} \equiv x_P \\ \xi &= \;\; 0 \quad \text{corresponds to } x = x_Q \\ \xi &= \;\; 1 \quad \text{corresponds to } x = \infty \end{aligned}$$

where $x_P$ is a point midway between $Q$ and $C$.

Alternatively the above mapping could be written directly in terms of the $Q$ and $P$ coordinates by simple elimination of $x_C$. This gives, using our previous notation,

$$\begin{aligned} x &= N_Q x_Q + N_P x_P \\ &= \left(1 + \frac{2\xi}{1-\xi}\right)x_Q - \frac{2\xi}{1-\xi}x_P \end{aligned} \tag{7.32b}$$

Both forms give a mapping that is independent of the origin of the $x$-coordinate as

$$N_Q + N_P = 1 = \bar{N}_C + \bar{N}_Q \tag{7.33}$$

The significance of the point $C$ is, however, of great importance. It represents the center from which the "disturbance" originates and, as we shall now show, allows the expansion of the form of Eq. (7.31) to be achieved on the assumption that $r$ is measured from $C$. Thus

$$r = x - x_C \tag{7.34}$$

If, for instance, the unknown function $u$ is approximated by a polynomial function using, say, hierarchical shape functions and giving

$$u = \alpha_0 + \alpha_1\xi + \alpha_2\xi^2 + \alpha_3\xi^3 + \cdots \tag{7.35}$$

we can easily solve Eq. (7.32a) for $\xi$, obtaining

$$\xi = 1 - \frac{x_Q - x_C}{x - x_C} = 1 - \frac{x_Q - x_C}{r} \tag{7.36}$$

Substitution into Eq. (7.35) shows that a series of the form given by Eq. (7.31) is obtained with the linear shape function in $\xi$ corresponding to $1/r$ terms, quadratic to $1/r^2$, etc.

In one dimension the objectives specified have thus been achieved and the element will yield convergence as the degree of the polynomial expansion, $p$, increases. Now a generalization to two or three dimensions is necessary. It is easy to see that this can be achieved by simple products of the one-dimensional infinite mapping with a "standard" type of shape function in the $\eta$ (and $\zeta$) directions in the manner indicated in Fig. 7.3.

First we generalize the interpolation of Eqs. (7.32a) and (7.32b) for any straight line in $x, y, z$ space and write (for such a line as $C_1 P_1 Q_1$ in Fig. 7.3)

$$\begin{aligned} x &= -\frac{\xi}{1-\xi} x_{C_1} + \left(1 + \frac{\xi}{1-\xi}\right) x_{Q_1} \\ y &= -\frac{\xi}{1-\xi} y_{C_1} + \left(1 + \frac{\xi}{1-\xi}\right) y_{Q_1} \\ z &= -\frac{\xi}{1-\xi} z_{C_1} + \left(1 + \frac{\xi}{1-\xi}\right) z_{Q_1} \qquad \text{(in three dimensions)} \end{aligned} \tag{7.37}$$

Secondly we complete the interpolation and map the whole $\xi\eta(\zeta)$ domain by adding a "standard" interpolation in the $\eta(\zeta)$ directions. Thus for the linear interpolation shown in Fig. 7.3 we can write for element $P P_1 Q Q_1 R R_1$

$$\begin{aligned} x = N_1(\eta) &\left[ -\frac{\xi}{1-\xi} x_C + \left(1 + \frac{\xi}{1-\xi}\right) x_Q \right] \\ + N_0(\eta) &\left[ -\frac{\xi}{1-\xi} x_{C_1} + \frac{\xi}{1-\xi} x_{Q_1} \right] \end{aligned} \tag{7.38}$$

with

$$N_1(\eta) = \frac{1}{2}(1+\eta), \qquad N_0(\eta) = \frac{1}{2}(1-\eta)$$

and map the points as shown. Similar expressions may be written for $y$ and $z$.

In a similar manner we could use quadratic interpolations and map an element as shown in Fig. 7.4 by using quadratic functions in $\eta$.

Thus it is an easy matter to create infinite elements and join these to a standard element mesh as shown in Fig. 7.2b. In the generation of such element properties only the transformation Jacobian matrix differs from standard forms, hence only this has to be altered in conventional programs. Moreover, integration is again over the usual "parent" element.

The "origin" or "pole" of the coordinates $C$ can be fixed arbitrarily for each radial line, as shown in Fig. 7.3. This will be done by taking account of the knowledge of the physical solution expected.

## 7.7 Singular elements by mapping: Use in fracture mechanics, etc.

Many problems in elasticity exhibit a singular solution where stresses are infinite. For example, corners and adjacent edges with different boundary conditions can lead to the appearance of singularities. In classical elasticity this was studied by Williams [11]. Such a situation obviously cannot be easily treated by numerical means since computers do not allow the existence of an infinite value. Thus some special treatments are often needed when such situations arise.

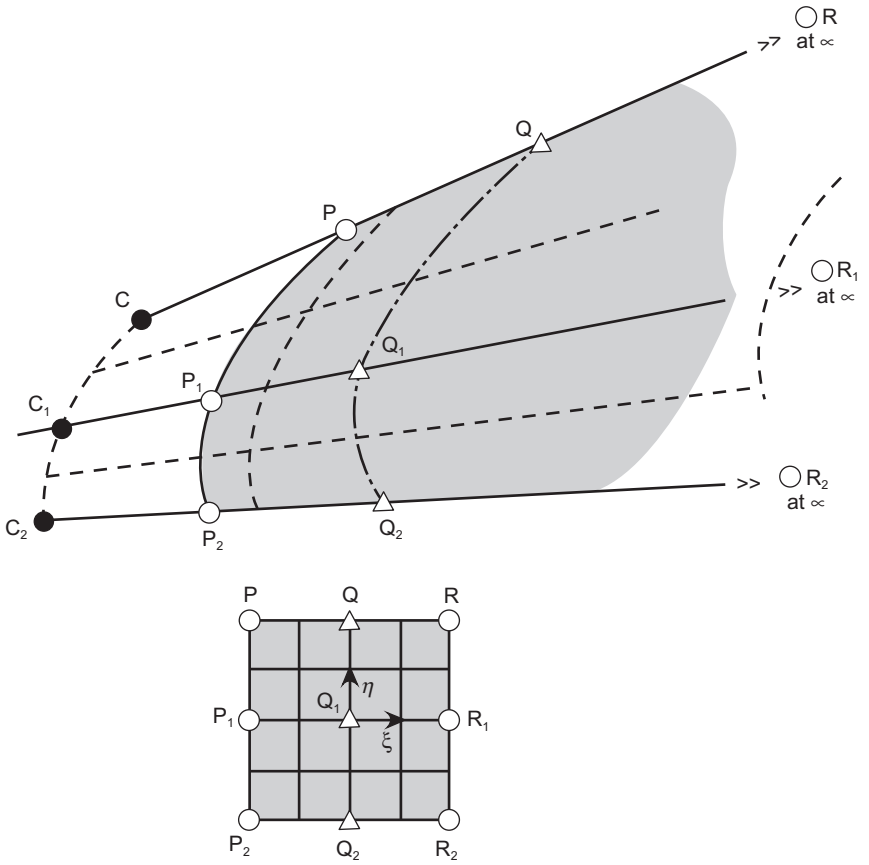



**FIGURE 7.4**
Infinite element map. Quadratic $\eta$ interpolation.

In the study of fracture mechanics interest is often focused on the singularity point where quantities such as stress become (mathematically, but not physically) infinite. Near such singularities normal, polynomial-based, finite element approximations perform badly and attempts have frequently been made here to include special functions within an element which can model the analytically known singularity. References [12–27] give an extensive literature survey of the problem and finite element solution techniques. An alternative to the introduction of special functions within an element—which frequently poses problems of enforcing continuity requirements with adjacent, standard, elements—lies in the use of special mapping techniques.

An element of this kind, shown in Fig. 7.5a, was introduced almost simultaneously by Henshell and Shaw [23] and Barsoum [24,25] for quadrilaterals by a simple shift of the mid-side node to the quarter point in quadratic, isoparametric elements.

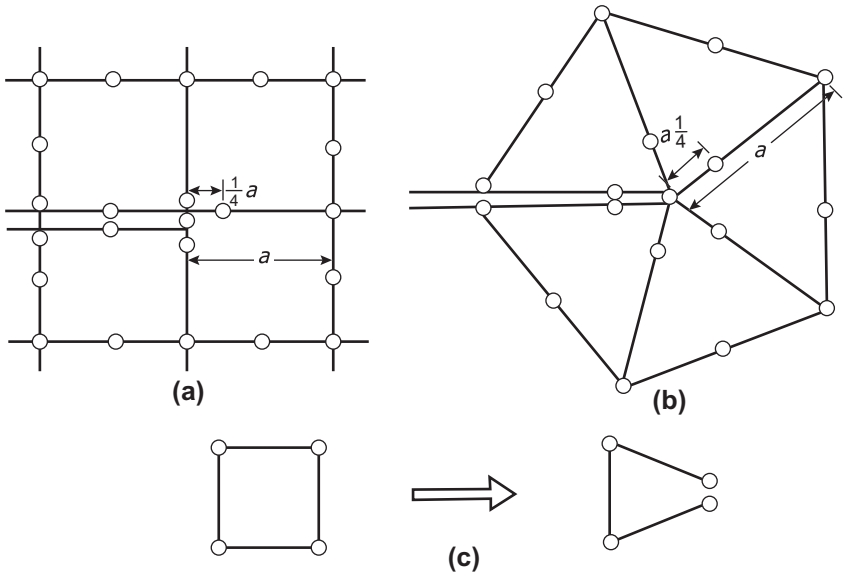



**FIGURE 7.5**
Singular elements from degenerate isoparametric elements (a)–(c).

It can now be shown (and we leave this exercise to the curious reader) that along the element edges the derivatives $\partial u/\partial x$ (or strains) vary as $1/\sqrt{r}$ where $r$ is the distance from the corner node at which the singularity develops. Although good results are achievable with such elements the singularity is, in fact, not well modeled on lines other than element edges. A development suggested by Hibbitt [26] achieves a better result by using triangular second-order elements for this purpose (Fig. 7.5b).

Indeed, the use of distorted or degenerate isoparametric elements is not confined to elastic singularities. Rice [17] shows that in the case of plasticity a shear strain singularity of $1/r$ type develops and Levy et al. [28] use an isoparametric, linear quadrilateral to generate such a singularity by the simple device of coalescing two nodes but treating these displacements independently. A variant of this is developed by Rice and Tracey [13].

The elements just described are simple to implement without any changes in a standard finite element program.

## 7.8 Reporting results: Displacements, strains, and stresses

The reader will now have observed that, while the finite element representation of displacements is in a sense optimal as it is the primary variable, both the strains and the stresses are not realistic. In particular, in ordinary engineering problems both strains and stresses tend to be continuous within a single material. The answers which are

obtained by the finite element calculation result in discontinuities of both strains and stresses between adjacent elements. Thus if the direct calculation of these quantities were presented the answers would be deemed unrealistic. For this reason, from the beginning of the finite element method it was sought to establish these rather important quantities in a more realistic, and possibly more accurate, way. In the very early days of finite element calculation with simple elements an averaging of element strains and stresses, which are constant in triangular elements, was made at each node. This of course gave improved results at most points—except at those which were on the boundary.

Since the early days of averaging further attention was given to this subject and other methods were developed. The first of these methodologies was developed by Brauchli and Oden [29] in 1971 and consisted of assuming that a continuous representation of either strain or stress using the same $C_0$ functions as for displacements could be found by solving a least squares representation of the corresponding discontinuous (finite element) one. This method proved quite expensive but often gave results which were superior to the simple averaging—at least for some sets of problems. However, higher accuracy was not achieved despite the additional cost of solving a full set of algebraic equations. A alternative local procedure to improve results was proposed by Hinton and Campbell [30] and was once quite widely used.

The methods of recovery of strain and stress have progressed much further in recent years and in Chapter 15 we discuss these fully. We find that currently an optimal procedure, which generally gives higher order accuracy and has similar cost to simple averaging, is the patch recovery method [31–33]. In this the process of determining values of recovered strains or stresses assumes that:

1. At some points of the domain or each element, the strains and stresses calculated by the direct differentiation of the shape functions are more accurate than elsewhere. Indeed on many occasions at such points "superconvergence" is demonstrated which can make the accuracy at least one order higher than that of the finite element values computed from derivatives of shape functions.
2. A continuous representation of such strains and stresses can be given by finding nodal values which in the least squares sense approximate those computed by the optimal points. Now the increased accuracy will exist over the entire domain.

The discussion of the existence of such points at which higher order may exist is deferred to Chapter 15 but here we show how this can be easily incorporated into standard programs dealing with elasticity as well as those for any other similar problem form, including the quasi-harmonic problem.

Basically in the procedure we will assume a strain exists for an element and can be expressed by

$$\hat{\varepsilon} = N_a \tilde{\varepsilon}_a \tag{7.39}$$

where now $\varepsilon$ is any component of strain. A similar expression may be written for a stress component. The goal is to find appropriate values for $\tilde{\varepsilon}_a$ which give improved results. To do this we use a least squares method in which the strain in a patch

surrounding a *vertex* node on elements may be expressed in *global coordinates* by a polynomial expression of higher order, suitable for the number of unknown parameters in the strain expression. This polynomial expression is given by

$$\varepsilon^\star = \begin{bmatrix} 1 & (x - x_a) & (y - y_a) & \cdots \end{bmatrix} \begin{Bmatrix} \tilde{\varepsilon}_a \\ \alpha_1 \\ \alpha_2 \\ \vdots \end{Bmatrix} = \mathbf{P}_a(\mathbf{x})\,\boldsymbol{\alpha}_a \tag{7.40}$$

For three-node triangles or four-node quadrilaterals in two dimensions and four-node tetrahedron or eight-node brick elements in three dimensions a linear interpolation is used. For higher order elements, the polynomial in $\mathbf{P}_a$ is also raised to the order of the complete displacement approximation.

The parameters in $\varepsilon^\star$ are determined using the least squares problem given by

$$\Pi = \frac{1}{2} \sum_{e=1}^{n_a} \sum_{l} \left[ \mathbf{P}_a^{\mathrm{T}} \left( \mathbf{x}_l^e \right) \boldsymbol{\alpha}_a - \hat{\varepsilon}(\mathbf{x}_l) \right]^2 = \min \tag{7.41}$$

where $n_a$ is the number of elements attached to node $a$. The minimization condition results in

$$\mathbf{M}_a \boldsymbol{\alpha}_a = \mathbf{f}_a \tag{7.42}$$

where

$$\mathbf{M}_a = \sum_{e=1}^{n_a} \sum_{l} \mathbf{P}_a^{\mathrm{T}} \left( \mathbf{x}_l^e \right) \mathbf{P}_a \left( \mathbf{x}_l^e \right) \qquad \text{and} \qquad \mathbf{f}_a = \sum_{e=1}^{n_a} \sum_{l} \mathbf{P}_a^{\mathrm{T}} \left( \mathbf{x}_l^e \right) \hat{\varepsilon} \left( \mathbf{x}_l^e \right) \tag{7.43}$$

The values for the remaining nodes (e.g., at boundary, mid-side, and face locations) may be computed by averaging the extrapolated values computed from (7.40). For example, from a patch, the result at node $b$ $(b \neq a)$ is given by

$$\varepsilon^\star(\mathbf{x}_b) = \mathbf{P}_a(\mathbf{x}_b) \boldsymbol{\alpha}_a$$

and averaging the result from all patches which contain node $b$ gives the final result for $\tilde{\varepsilon}_b$.

An identical process may be used to compute stress values or any discontinuous field of interest. We recommend that a patch recovery method be used to report all strain or stress values. In addition, the method serves as the basis for error assessment and methods to efficiently construct adaptive solutions to a specified accuracy as we shall present in Chapter 16.

## 7.9 Discretization error and convergence rate

In the foregoing sections we have assumed that the approximation to the dependent variable as represented by (7.12) will yield the exact solution in the limit as the size

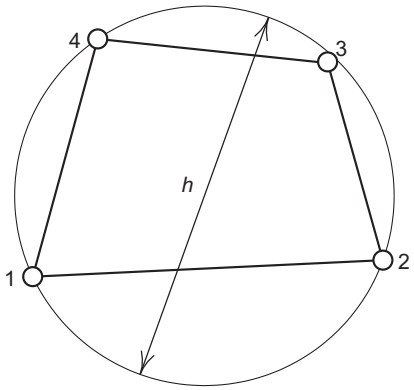



**FIGURE 7.6**
Element size parameter $h$ for four-node quadrilateral.

$h$ of elements decreases. The parameter $h$ is defined as the smallest ball that will contain the element domain $\Omega_e$. An example for a four-node quadrilateral is shown in Fig. 7.6. The arguments for this are simple: if the expansion is capable, in the limit, of exactly reproducing any solution form conceivable in the continuum, then as the solution of each approximation is unique it must approach, in the limit of $h \to 0$, the unique exact solution. In some cases the exact solution is indeed obtained with a finite number of subdivisions (or even with one element only) if the *shape function expansion used in that element fits the exact solution*. Thus, for instance, if the exact solution is of the form of a quadratic polynomial *and* the shape functions include all the polynomials of that order, the approximation will yield the exact answer.

The last argument helps in determining the order of convergence of the finite element procedure as the exact solution can always be expanded in a Taylor series in the vicinity of any point (or node) $a$ as a polynomial:

$$\mathbf{u} = \mathbf{u}_a + \left(\frac{\partial \mathbf{u}}{\partial x}\right)_a (x - x_a) + \left(\frac{\partial \mathbf{u}}{\partial y}\right)_a (y - y_a) + \cdots \tag{7.44}$$

If within an element of "size" $h$ a polynomial expansion complete to degree $p$ is employed, this can fit locally the Taylor expansion up to that degree and, as $x - x_i$ and $y - y_i$ are of the order of magnitude $h$, the error in $\mathbf{u}$ will be of the order $O(h^{p+1})$. Thus, using linear elements we use a complete linear expansion and $p = 1$. We should therefore expect a *convergence* rate of order $O(h^2)$, i.e., the error in the solution being reduced to $1/4$ for a halving of the mesh spacing.

By a similar argument the gradients in the solution, which are given by the $m$th derivatives of the dependent variable, should converge with an error of $O(h^{p+1-m})$, i.e., as $O(h^p)$ in elasticity and field problems, where $m = 1$. The elastic energy, being given by the square of stresses, will show an error of $O(h^{2(p+1-m)})$ or $O(h^2)$ for linear order elements in any elasticity example.

The arguments given here are perhaps "heuristic" from a mathematical viewpoint—they are, however, true [34,35] and correctly give the orders of convergence, which can be expected to be achieved asymptotically as the element size tends to zero and *if the exact solution does not contain singularities*. Such singularities may result in infinite values of the coefficients in terms omitted in the Taylor expansion of Eq. (7.44) and invalidate the arguments. However, in many well-behaved problems the mere determination of the order of convergence often suffices to extrapolate the solution to the correct result. Thus, for instance, if the displacement converges at $O(h^2)$ and we have two approximate solutions $u^1$ and $u^2$ obtained with meshes of sizes $h$ and $h/2$, we can write, with $u$ being the exact solution,

$$\frac{u^1 - u}{u^2 - u} = \frac{O(h^2)}{O(h/2)^2} \approx 4 \tag{7.45}$$

From the above an (almost) exact solution $u$ can be predicted. This type of extrapolation was first introduced by Richardson [36] and is of use if convergence is monotonic and nearly asymptotic.

We shall return to the important question of estimating errors due to the discretization process in Chapter 15 and will show that much more precise methods than those arising from convergence rate considerations are possible today. Indeed automatic mesh refinement processes can be introduced so that the specified accuracy can be achieved (viz. Chapters 16 and 17).

Discretization error is not the only error possible in a finite element computation. In addition to obvious mistakes which can occur when introducing data into computers, errors due to *round-off* are always possible. With the computer operating on numbers rounded to a finite number of digits, a reduction of accuracy occurs every time differences between "like" numbers are being formed. In the process of equation solving many subtractions are necessary and accuracy decreases. Problems of matrix conditioning, etc., enter here and the user of the finite element method must at all times be aware of accuracy limitations which simply do not allow the exact solution ever to be obtained. Fortunately in many computations, by using modern machines which carry a large number of significant digits, these errors are often small.

Another error that is often encountered occurs in approximation of curved boundaries by polynomials on faces of elements. For example, we observed that use of linear triangles or quadrilaterals to approximate a curved boundary causes an error of $O(h^2)$ to be introduced.

## 7.10 Minimization of total potential energy

The weak form approach used in the previous sections is identical to the principle of virtual work and ensures satisfaction of equilibrium conditions within the limits prescribed by the assumed displacement pattern. Only if the virtual work equality for all arbitrary variations of displacement is ensured would the equilibrium be satisfied at every point in the domain.

As the number of parameters of $\tilde{\mathbf{u}}$ which prescribes the displacement increases without limit then ever closer approximation of all equilibrium conditions can be ensured.

For the quasi-static problem the weak form given by (7.7) becomes

$$\mathrm{G}(\delta\mathbf{u}, \mathbf{u}, \boldsymbol{\sigma}) = \int_\Omega \delta\boldsymbol{\varepsilon}^{\mathrm{T}}\boldsymbol{\sigma}\,\mathrm{d}\Omega - \int_\Omega \delta\mathbf{u}^{\mathrm{T}}\mathbf{b}\,\mathrm{d}\Omega - \int_{\Gamma_t} \delta\mathbf{u}^{\mathrm{T}}\bar{\mathbf{t}}\,\mathrm{d}\Gamma = 0 \tag{7.46}$$

and can be restated in a different form if the virtual quantities $\delta\mathbf{u}$ and $\delta\boldsymbol{\varepsilon}$ are considered as *variations* of the real quantities [37,38].

Thus, for instance, we can write the first term of Eq. (7.46), for elastic materials, as

$$\delta U = \int_\Omega \delta\boldsymbol{\varepsilon}^{\mathrm{T}}\boldsymbol{\sigma}\,\mathrm{d}\Omega \tag{7.47}$$

where $U$ is the *strain energy* of the system. For the linear elastic material described by Eq. (2.55) the strain energy is given by

$$U = \frac{1}{2}\int_\Omega \boldsymbol{\varepsilon}^{\mathrm{T}}\mathbf{D}\,\boldsymbol{\varepsilon}\,\mathrm{d}\Omega + \int_\Omega \boldsymbol{\varepsilon}^{\mathrm{T}}\left(\boldsymbol{\sigma}_0 - \mathbf{D}\,\boldsymbol{\varepsilon}_0\right)\,\mathrm{d}\Omega \tag{7.48}$$

and will, after variation, yield the correct expression providing $\mathbf{D}$ is a symmetric matrix (this is a necessary condition for a single-valued $U$ to exist) [37,38].

The last two terms of Eq. (7.46) can be written as

$$\delta W = -\delta\left(\int_\Omega \mathbf{u}^{\mathrm{T}}\mathbf{b}\,\mathrm{d}\Omega + \int_\Gamma \mathbf{u}^{\mathrm{T}}\bar{\mathbf{t}}\,\mathrm{d}\Gamma\right) \tag{7.49}$$

where $W$ is the *potential energy of the external loads*. The above is certainly true if $\mathbf{b}$ and $\bar{\mathbf{t}}$ are conservative (or independent of displacement) where we obtain simply

$$W = -\int_\Omega \mathbf{u}^{\mathrm{T}}\mathbf{b}\,\mathrm{d}\Omega - \int_\Gamma \mathbf{u}^{\mathrm{T}}\bar{\mathbf{t}}\,\mathrm{d}\Gamma \tag{7.50}$$

Thus, instead of Eq. (7.46), we can write the *total potential energy*, $\Pi$, as

$$\Pi = U + W \tag{7.51}$$

in which $U$ is given by (7.48) and $W$ by (7.50) and require

$$\delta\Pi = \delta(U + W) = 0 \tag{7.52}$$

In this form $\Pi$ is known as a *functional* and (7.52) is a requirement which renders the functional *stationary*. What we have devised is a natural variational principle of the type described in Chapter 4.

The above statement means that for equilibrium to be ensured the *total potential energy must be stationary* for variations of the admissible displacements. The finite element equations for the total potential energy are obtained by substituting the approximation for displacements [viz. Eq. (7.12)] into Eqs. (7.48) and (7.50) giving

$$\Pi = \frac{1}{2}\tilde{\mathbf{u}}^{\mathrm{T}}\mathbf{K}\,\tilde{\mathbf{u}} - \tilde{\mathbf{u}}^{\mathrm{T}}\mathbf{f} \tag{7.53}$$

in which $\mathbf{K}$ (where $\mathbf{K} = \mathbf{K}^{\mathrm{T}}$) and $\mathbf{f}$ are given by Eq. (7.16b). The variation with respect to displacements with the finite number of parameters $\tilde{\mathbf{u}}$ is now written as

$$\frac{\partial \Pi}{\partial \tilde{\mathbf{u}}} = \begin{Bmatrix} \dfrac{\partial \Pi}{\partial \tilde{\mathbf{u}}_1} \\ \dfrac{\partial \Pi}{\partial \tilde{\mathbf{u}}_2} \\ \vdots \end{Bmatrix} = \mathbf{K}\,\tilde{\mathbf{u}} - \mathbf{f} = \mathbf{0} \tag{7.54}$$

It can also be shown that in stable elastic situations the total potential energy is not only stationary but is a minimum when all integrals are exactly evaluated [37]. *Thus the finite element process seeks such a minimum within the constraint of an assumed displacement pattern*.

The greater the number of degrees of freedom, the more closely the solution will approximate the true one, ensuring complete equilibrium, providing the true displacement can, in the limit, be represented. The necessary convergence conditions for the finite element process could thus be derived. Discussion of these will, however, be deferred to subsequent sections.

It is of interest to note that if true equilibrium requires an absolute minimum of the total potential energy, $\Pi$, a finite element solution by the displacement approach will always provide an approximate $\Pi$ greater than the correct one. *Thus a bound on the value of the total potential energy is always achieved*.

If the functional $\Pi$ could be specified, *a priori*, then the finite element equations could be derived directly by the differentiation specified by Eq. (7.54).

The well-known Rayleigh [39]-Ritz [40] process of approximation frequently used in elastic analysis is based precisely on this approach. The total potential energy expression is formulated and the displacement pattern is assumed to vary with a finite set of undetermined parameters. A set of simultaneous equations minimizing the total potential energy with respect to these parameters is set up. Thus the finite element process as described so far also can be considered to be the Rayleigh-Ritz procedure.

### 7.10.1 Bound on strain energy in a displacement formulation

While the approximation obtained by the finite element displacement approach always overestimates the true value of the total potential energy $\Pi$ (the absolute minimum corresponding to the exact solution), this is not directly useful in practice. It is, however, possible to obtain a more useful limit in special cases.

Consider the problem in which no initial strains $\boldsymbol{\varepsilon}_0$ or initial stresses $\boldsymbol{\sigma}_0$ exist. Now by the principle of energy conservation the strain energy will be equal to the work done by the external loads which increase uniformly from zero [41]. This work done is equal to $-W/2$ where $W$ is the potential energy of the loads. Thus,

$$U + \frac{1}{2}W = 0 \tag{7.55}$$

or

$$\Pi = U + W = -U \tag{7.56}$$

whether an exact or approximate displacement field is assumed.

If only one external concentrated load $R_a$ is present, the strain energy bound immediately informs us that the finite element deflection under this load has been underestimated (as $U = -W/2 = -R_a u_a/2$, where $u_a$ is the deflection at the load point). In more complex loading cases the usefulness of this bound is limited as neither local displacements nor local stresses, i.e., the quantities of real engineering interest, can be bounded. It is also important to remember that this bound on strain energy is only valid in the absence of any initial stresses or strains.

The expression for $U$ in this case can be obtained from Eq. (7.48) as

$$U = \frac{1}{2}\int_{\Omega} \boldsymbol{\varepsilon}^{\mathrm{T}}\mathbf{D}\,\boldsymbol{\varepsilon}\,\mathrm{d}\Omega \tag{7.57}$$

which by substituting $\boldsymbol{\varepsilon} = \mathbf{B}\,\tilde{\mathbf{u}}$ becomes simply

$$U = \frac{1}{2}\tilde{\mathbf{u}}^{\mathrm{T}}\int_{\Omega} \mathbf{B}^{\mathrm{T}}\mathbf{D}\,\mathbf{B}\,\mathrm{d}\Omega\,\tilde{\mathbf{u}} = \frac{1}{2}\tilde{\mathbf{u}}^{\mathrm{T}}\mathbf{K}\,\tilde{\mathbf{u}} \tag{7.58}$$

a *quadratic matrix form* in which $\mathbf{K}$ is the stiffness matrix previously discussed.

### 7.10.2 Direct minimization

The fact that the finite element approximation reduces to the problem of minimizing the total potential energy $\Pi$ defined in terms of a finite number of nodal parameters led us to the formulation of the simultaneous set of equations given symbolically by Eq. (7.54). This is the most usual and convenient approach, especially in linear solutions, but other search procedures, now well developed in the field of optimization, could be used to estimate the lowest value of $\Pi$. In this text we shall continue with the simultaneous equation process but the interested reader could well bear the alternative possibilities in mind [42,43].

## 7.11 Finite element solution process

By now it should be clear to the reader that the finite element solution of a problem always follows a standard methodology. The solution process to any problem type is always performed by the following steps:

1. Define the problem to be solved in terms of differential equations. Construct the integral form for the problem as a virtual work, variational, or weak formulation.
2. Select the type and order of finite elements to be used in the analysis.
3. Define the mesh for the problem. This involves the description of the node and element layout, as well as the specification of boundary conditions and parameters for the formulation used. Simple forms using subdivision of blocks were indicated

in Section 6.10. A process for general mesh generation will be described in more detail in Chapter 17.

4. Compute and assemble the element arrays. The particular virtual work, variational, or weak form provides the basis for computing the specific relationships of each element.
5. Solve the resulting set of linear algebraic equations for the unknown parameters. For static problems this requires the solution to (7.18). See Appendix D for a brief discussion on solution of linear algebraic equations. For transient problems it is necessary to also select the method to treat the time variations, such as described in Chapter 3.
6. Output the results for the nodal and element variables. Graphical outputs also are useful for this step. An accurate procedure to project element values to nodes is described in Section 7.8.

Much of the discussion in this and following chapters is concerned with the development of the theory needed to compute element arrays. For a steady-state problem the two arrays are a coefficient array $\mathbf{K}$, which we refer to as a "stiffness" matrix, and a force array $\mathbf{f}$. For transient problems it is necessary to also compute $\mathbf{M}$, which we call the "mass" matrix.

## 7.12 Numerical examples

Let us now consider the solution to a set of problems for which an exact solution is known. This will enable us to see how the finite element results compare to the known solution and to demonstrate the convergence properties for different element types. Of course, the power of the finite element method is primarily for use on problems for which no alternative solution is possible using results from classical books on elasticity. Later we will include a few results for such example problems.

**Example 7.3.** Patch test

To validate the correctness of any implementation of a finite element method into a computer system, a test for a simple mesh of elements for a specified *linear* order polynomial solution should be performed. Such a test is called a *patch test* and was originally proposed by Irons and co-workers [44]. As an example we consider a two-dimensional plane stress solution given by

$$u(x, y) = 0.01\, x \qquad \text{and} \qquad v = 0$$

The strains are $\varepsilon_x = 0.01$ and all other strains are zero. The stresses are given by

$$\sigma_x = \frac{E}{1 - \nu^2}\, \varepsilon_x \qquad \text{and} \qquad \sigma_y = \frac{\nu\, E}{1 - \nu^2}\, \varepsilon_x$$

and all other stresses are zero. To solve the problem we consider the mesh of five quadrilateral elements shown in Fig. 7.7 in which restrained boundary conditions are imposed on nodes 1 and 4 as shown and nodal forces on nodes 2, 3, and 4. The patch

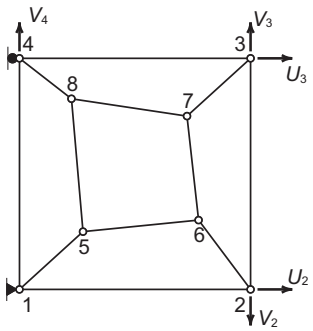



**FIGURE 7.7**
Patch test for two-dimensional plane stress.

**Table 7.1** Coordinates and Nodal Displacement in Fig. 7.7

| | Node Numbers | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| $x$ | 0.00 | 10.00 | 10.00 | 0.00 | 2.75 | 7.75 | 8.25 | 2.25 |
| $y$ | 0.00 | 0.00 | 10.00 | 10.00 | 2.50 | 3.00 | 7.50 | 8.25 |
| $u$ | 0.0000 | 0.1000 | 0.1000 | 0.0000 | 0.0275 | 0.0775 | 0.0825 | 0.0225 |
| $v$ | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |

is square with a side length of 10 units and we let $E = 1500$ and $\nu = 0.25$ which gives the exact stresses $\sigma_x = 16$ and $\sigma_y = 4$. The traction on the right edge is also 16 and thus, nodal forces in the $x$ direction are $U_3 = U_4 = 80$ and those in the $y$ direction are $V_2 = -20$, $V_3 = V_4 = 20$. The location for nodes and the expected nodal displacements are given in Table 7.1. Results for displacements and stresses must be obtained to within computer round-off for the patch test to be satisfied—this is achieved for properly coded isoparametric four-node quadrilateral elements. The patch test is an important concept and we devote more attention to the matter in Chapter 8.

**Example 7.4.** Beam subjected to end moment

In this example we consider the pure bending of a beam. The solution consists of quadratic displacements and, thus, has an exact polynomial solution. Two meshes are considered as shown in Fig. 7.8a. The properties are taken as $E = 30{,}000$, $\nu = 0.3$, and $M = 600$ which gives a vertical tip displacement of $-2$ units. Since the solution is only quadratic we expect that nine-node Lagrangian elements should give the exact solution for both regular and sub-parametric mapped shapes. On the other hand using eight-node serendipity elements gives the exact solution only for the case where elements are rectangular. In Fig. 7.8b we show the displacement along the centerline

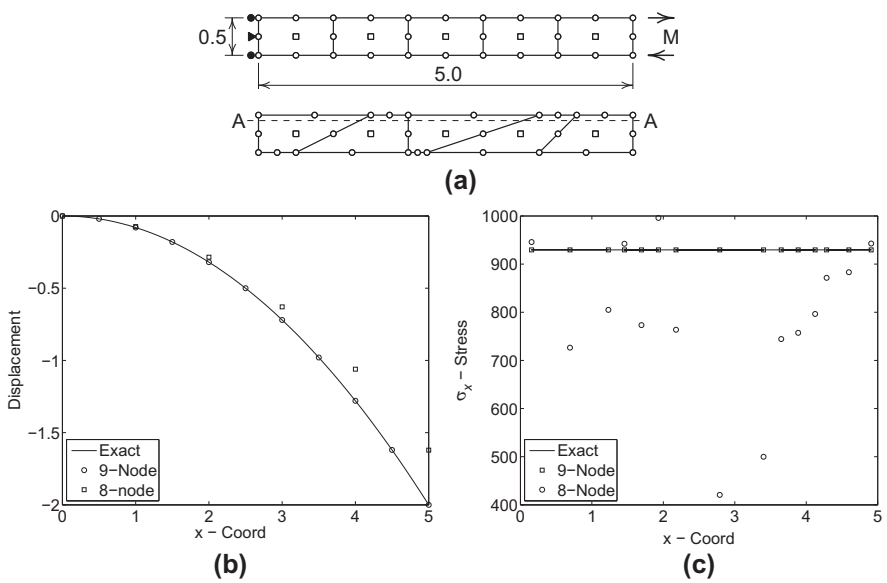



**FIGURE 7.8**
Beam with end loaded moment: (a) meshes, (b) displacement, and (c) $\sigma_x$ stress on AA.

for the distorted mesh and observe the expected result. Each element stiffness matrix is computed using a $3 \times 3$ Gaussian quadrature. Similarly, in Fig. 7.8c we show the $\sigma_x$ stress along the upper line of quadrature points (AA) for both cases. Here the Lagrangian nine-node element again gives the expected exact result while the serendipity eight-node element exhibits wild oscillations and very inaccurate stress results. Based on this result we recommend use of serendipity elements only for cases where mesh distortion is small in high bending regions. When we consider incompressible problems in Chapter 10 we will find some additional recommendations on use of Lagrangian and serendipity elements.

**Example 7.5.** Beam subjected to end shear

We next consider a rectangular beam in a state of plane stress. The geometric properties are shown in Fig. 7.9a. The exact solution to the problem is given in Timoshenko and Goodier based on use of an Airy stress function solution [45]. The solution for stresses is given by

$$\sigma_x = -\frac{3}{2}\frac{Pxy}{c^3}$$
$$\sigma_y = 0$$
$$\tau_{xy} = -\frac{3P}{4c}\left[1 - \left(\frac{y}{c}\right)^2\right]$$

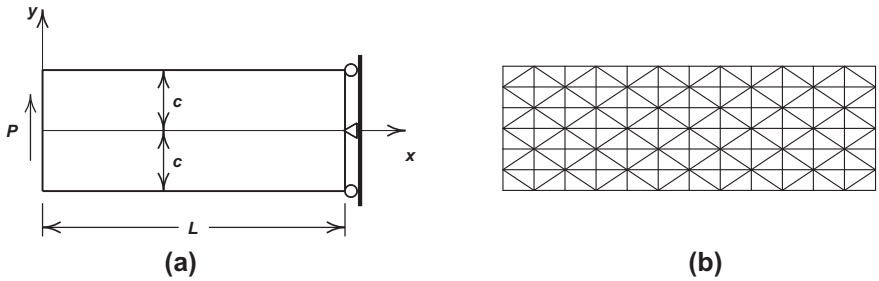



**FIGURE 7.9**
End loaded beam: (a) problem geometry and (b) coarse three-node mesh.

where $P$ is the applied end load and $c$ is the half-depth of the beam. For the displacement boundary conditions

$$u(L,0) = v(L,0) = 0 \quad \text{and} \quad u(L,c) = u(L,-c) = 0$$

as shown in Fig. 7.9a, the solution for displacements is given by

$$u = -\frac{P(x^2 - L^2)y}{2EI} - \frac{\nu P y(y^2 - c^2)}{6EI} + \frac{Py(y^2 - c^2)}{6GI}$$
$$v = \frac{\nu P x y^2}{2EI} + \frac{P(x^3 - L^3)}{6EI} - \left(\frac{PL^2}{2EI} + \frac{\nu P c^2}{6EI} + \frac{Pc^2}{3GI}\right)(x - L)$$

In the above $E$ and $\nu$ are the elastic modulus and Poisson ratio, $G$ is the shear modulus given by $E/[2(1+\nu)]$, and $I$ is the area moment of inertia which is equal to $2wc^3/3$ where $w$ is a constant beam thickness.

For this solution the tractions on the boundaries become

$$\begin{Bmatrix} t_x \\ t_y \end{Bmatrix} = w \begin{Bmatrix} 0 \\ -\tau_{xy} \end{Bmatrix} \quad \text{for } x = 0,\ -c \le y \le c$$
$$\begin{Bmatrix} t_x \\ t_y \end{Bmatrix} = w \begin{Bmatrix} -\sigma_x \\ \tau_{xy} \end{Bmatrix} \quad \text{for } x = L,\ -c \le y \le c$$

For the numerical solution we choose the properties

$$c = 10, \quad L = 100, \quad w = 1, \quad P = 80, \quad E = 1000, \quad \text{and} \quad \nu = 0.25$$

In order to perform a finite element solution to the problem we need to compute the nodal forces for the tractions using a two-dimensional form of Eq. (7.24). When many elements are used in an analysis this step can be quite tedious and it is best to write a small computer program to carry out the integrations (e.g., using MATLAB [46], GNU Octave [47,48], or any other programming language).[2]

[2]For the triangular and quadrilateral elements discussed in this chapter, the program *FEAPpv* available as a companion to this book includes automatic computation of nodal forces for this type of loading [49].

The solution to the problem is carried out using a uniform mesh of (a) three-node triangular elements and (b) four-node quadrilateral elements and the results for the error in energy given by

$$\eta_E = \frac{|E_{ex} - E_{fe}|}{E_{ex}} \approx C\,h^2$$

are plotted versus the element size $h/h_1$ in Fig. 7.10. Here $E_{ex}$ is the energy of the exact solution and $E_{fe}$ that of the finite element solution. Results for the energy are given in Table 7.2 and the exact value for the geometry and properties selected is 3296 (energy here is work done which is twice the stored elastic strain energy). The element size is normalized to that of the coarsest mesh ($h_1$) for each element type and the energy error is computed using the exact value. The expected slope $p = 2$ is achieved for both element types with the four-node element giving a smaller constant $C$ due to the presence of the $xy$ term in each shape function.

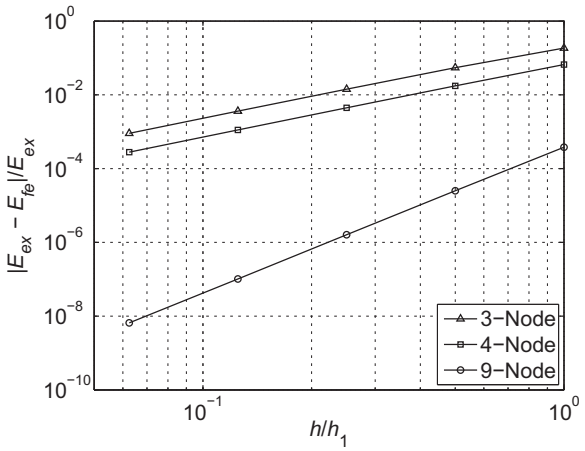



**FIGURE 7.10**
Convergence in energy error for three-node, four-node, and nine-node elements.

**Table 7.2** Mesh Size and Energy for End-Loaded Beam

| | Three-Node Triangles | | Four-Node Rectangles | | Nine-Node Rectangles | |
|---|---|---|---|---|---|---|
| **Nodes** | **Elements** | **Energy** | **Elements** | **Energy** | **Elements** | **Energy** |
| 91 | 144 | 2685.6650 | 72 | 3077.4986 | 18 | 3294.7512 |
| 325 | 576 | 3116.8360 | 288 | 3238.2915 | 72 | 3295.9174 |
| 1225 | 2304 | 3249.0252 | 1152 | 3281.3465 | 288 | 3295.9947 |
| 4753 | 9216 | 3284.0969 | 4608 | 3292.3206 | 1152 | 3295.9997 |
| 18721 | 36864 | 3293.0129 | 18432 | 3295.0790 | 4608 | 3296.0000 |
| Exact | – | 3296.0000 | – | 3296.0000 | – | 3296.0000 |

The rectangular beam is solved again using Lagrangian 9-node (bi-quadratic) and 16-node (bi-cubic) elements. The mesh for the bi-quadratic model initially has 3 elements in the depth direction and 6 along the length for a total of 18 elements and 91 nodes. This is subsequently subdivided to form meshes with $6 \times 12$, $12 \times 24$, $24 \times 48$, and $48 \times 96$ elements. All other data is as defined above.

The analysis is repeated using 16-node bi-cubic elements with an initial mesh of $2 \times 4$ elements, which again gives a mesh with 91 nodes. Since the elements are rectangular (sub-parametric mapping) and the exact solution for displacements given above contains all polynomial terms of degree 3 or less, the solution with this coarse mesh is exact and no refinement is needed.

In Table 7.2 we present the results for the energy obtained from each mesh and in Fig. 7.10 we show the convergence behavior for the nine-node element form. Again, the expected rate of convergence is attained as indicated by the slope 4 in the figure. The very rapid convergence with nine-node elements indicates their superiority compared to use of three- or four-node elements.

**Example 7.6.** Circular beam subjected to end shear

We consider a circular beam in a state of plane stress. The geometric properties and loading for the problem are shown in Fig. 7.11a. The solution to the problem is given in Timoshenko and Goodier based on use of an Airy stress function [45]. The solution for stresses is given by

$$\sigma_r = \frac{P}{N}\left[\, r + \frac{a^2 b^2}{r^3} - \frac{a^2 + b^2}{r} \right] \sin\theta$$

$$\sigma_\theta = \frac{P}{N}\left[ 3r - \frac{a^2 b^2}{r^3} - \frac{a^2 + b^2}{r} \right] \sin\theta$$

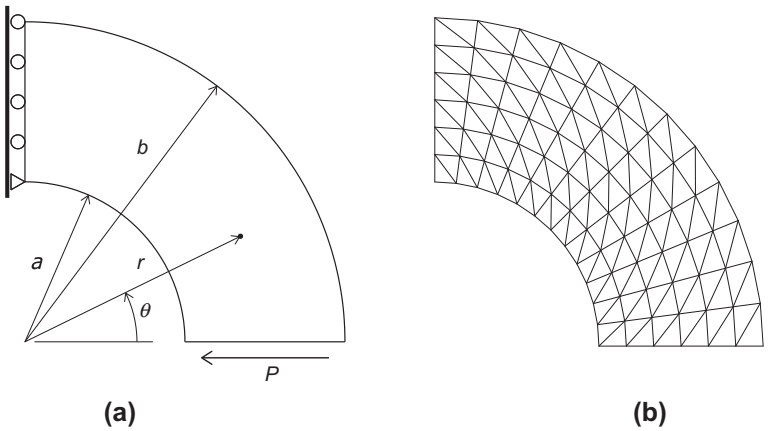



**FIGURE 7.11**
End-loaded circular beam: (a) problem geometry and (b) coarse mesh.

$$\tau_{r\theta} = -\frac{P}{N}\left[\, r + \frac{a^2b^2}{r^3} - \frac{a^2+b^2}{r}\right]\cos\theta$$

where $N = a^2 - b^2 + (a^2 + b^2)\ln b/a$. For the restraints shown in Fig. 7.11a the solution for displacements is given by

$$\begin{aligned} u_r = \frac{P}{NE}\Bigg\{ &\left[\frac{1}{2}(1-3\nu)r^2 - \frac{a^2b^2(1+\nu)}{2r^2} - (a^2+b^2)(1-\nu)\ln r\right]\sin\theta \\ &+ (a^2+b^2)(2\theta-\pi)\cos\theta\Bigg\} - K\sin\theta \\ u_\theta = -\frac{P}{NE}\Bigg\{ &\left[\frac{1}{2}(5+\nu)r^2 - \frac{a^2b^2(1+\nu)}{2r^2}\right. \\ &\left. + (a^2+b^2)[(1-\nu)\ln r + (1+\nu)]\right]\cos\theta \\ &+ (a^2+b^2)(2\theta-\pi)\sin\theta\Bigg\} - K\cos\theta \end{aligned}$$

where, for $u_r(a, \pi/2) = 0$, we obtain

$$K = \frac{P}{NE}\left[\frac{1}{2}(1-3\nu)a^2 - \frac{b^2(1+\nu)}{2} - (a^2+b^2)(1-\nu)\ln a\right]$$

In the above $E$ and $\nu$ are the elastic modulus and Poisson ratio; $a$ and $b$ are the inner and outer radii, respectively (see Fig. 7.11a).

For this solution the displacement $u_r$ at $\theta = 0$ is constant and given by

$$u_r(r, 0) = -\frac{\pi P}{2NE}(a^2 + b^2) = u_0$$

Thus, instead of computing the nodal forces for the traction on this boundary we merely set all the nodal displacements in the $x$ direction ($r$ direction for $\theta = 0$) to a constant value.

For the numerical solution we choose the properties

$$a = 5, \quad b = 10, \quad w = 1, \quad u_0 = -0.01, \quad E = 10{,}000 \quad \text{and} \quad \nu = 0.25$$

In addition the displacement boundary conditions in Cartesian coordinates are prescribed as

$$u(x, 0) = u_0 \qquad \text{and} \qquad u(0, y) = v(0, a) = 0$$

The finite element solution to the problem is carried out first using a uniform mesh of three-node triangular elements oriented as shown in Fig. 7.11b. The initial mesh has three radial spaces and six in the circumferential direction (a $3 \times 6$ mesh). Results for the energy and its error are listed in Table 7.3 with the exact value given by

$$E_{ex} = \frac{1}{\pi}\left[\ln 2 - 0.6\right] = 0.0296496684424$$

for the geometry and properties selected.

**Table 7.3** Mesh Size and Energy for Curved Beam

| | Three-Node Triangles | | |
|---|---|---|---|
| **Nodes** | **Elements** | **Energy** | **Error (%)** |
| 91 | 144 | 0.0334113422626 | 12.687 |
| 325 | 576 | 0.0306058930879 | 3.225 |
| 1225 | 2304 | 0.0298900009824 | 0.811 |
| 4753 | 9216 | 0.0297098490031 | 0.203 |
| 18721 | 36864 | 0.0296647207994 | 0.051 |
| 74305 | 147456 | 0.0296534320547 | 0.013 |
| Exact | – | 0.0296496684424 | – |

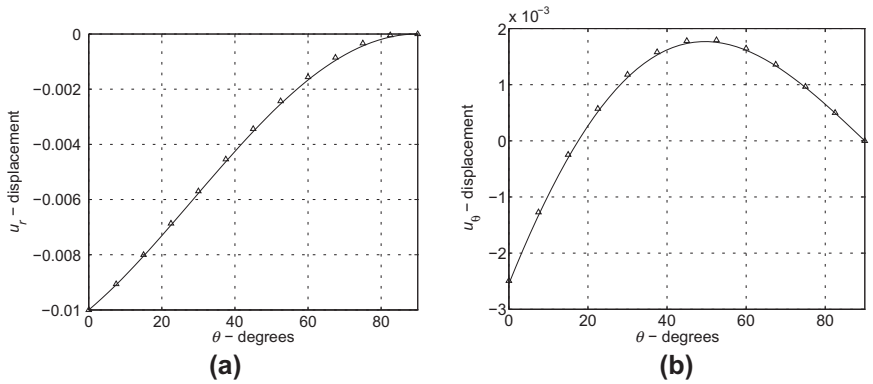



**FIGURE 7.12**
End-loaded circular beam: (a) $u_r$ displacement and (b) $u_\theta$ displacement for $r = a$.

Finally, in Fig. 7.12 we compare the $u_r$ and $u_\theta$ displacements from the finite element solution of the coarsest three-node triangular mesh to the exact values. We observe that even with this coarse distribution of elements the solution is quite good. Unfortunately, the stress distribution is not as accurate and quite fine meshes are needed to obtain good values. Thus, the solution to the problem is performed again using isoparametric 4-node bi-linear quadrilaterals, 9-node Lagrangian quadrilaterals, and 16-node Lagrangian quadrilaterals.

The initial mesh for all element types uses a regular subdivision division of the domain that produces initial element patterns with $6 \times 12$ 4-node elements, $3 \times 6$ 9-node elements, and $2 \times 4$ 16-node elements. The mesh for each element form is shown in Fig. 7.13.

Results for the energy are given in Table 7.4 and compared to the exact value. The element size is normalized to that of the coarsest mesh (shown in Fig. 7.13a) and the energy error computed from Table 7.4 has the expected slope $p = 2$ for four-node elements, $p = 4$ for nine-node elements, and $p = 6$ for cubic elements as shown in Fig. 7.14.

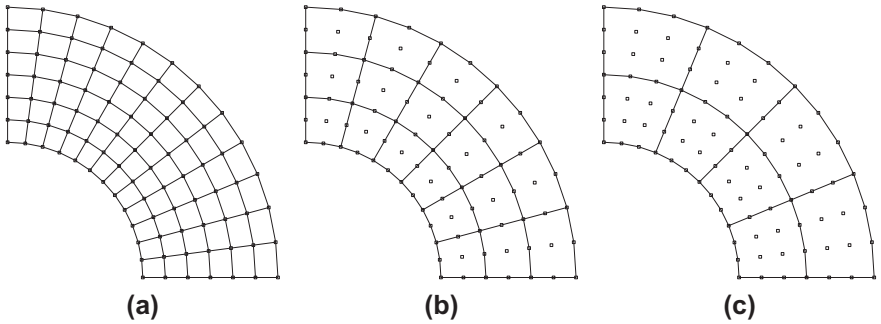



**FIGURE 7.13**
End-loaded circular beam: coarse mesh for four-node, nine-node, and 16-node Lagrangian elements.

**Table 7.4** Mesh Size and Energy for Curved Beam

| | 4-Node Quadrilateral | | 9-Node Quadrilateral | | 16-Node Quadrilateral | |
|---|---|---|---|---|---|---|
| **Nodes** | **Elements** | **Energy** | **Elements** | **Energy** | **Elements** | **Energy** |
| 91 | 72 | 0.030420381 | 18 | 0.029701013 | 8 | 0.029653273 |
| 325 | 288 | 0.029843513 | 72 | 0.029653181 | 32 | 0.029649752 |
| 1225 | 1152 | 0.029698207 | 288 | 0.029649894 | 128 | 0.029649669 |
| 4753 | 4608 | 0.029661808 | 1152 | 0.029649682 | 512 | 0.029649668 |
| 18721 | 18432 | 0.029652703 | 4608 | 0.029649669 | 2048 | 0.029649668 |
| Exact | – | 0.029649668 | – | 0.029649668 | – | 0.029649668 |

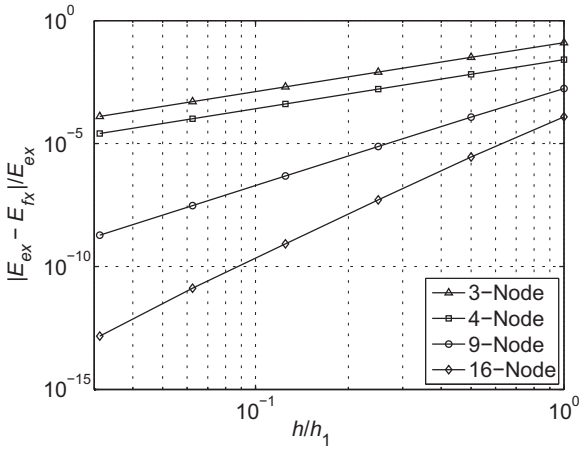



**FIGURE 7.14**
Curved beam: convergence in energy error for three-node to 16-node elements.

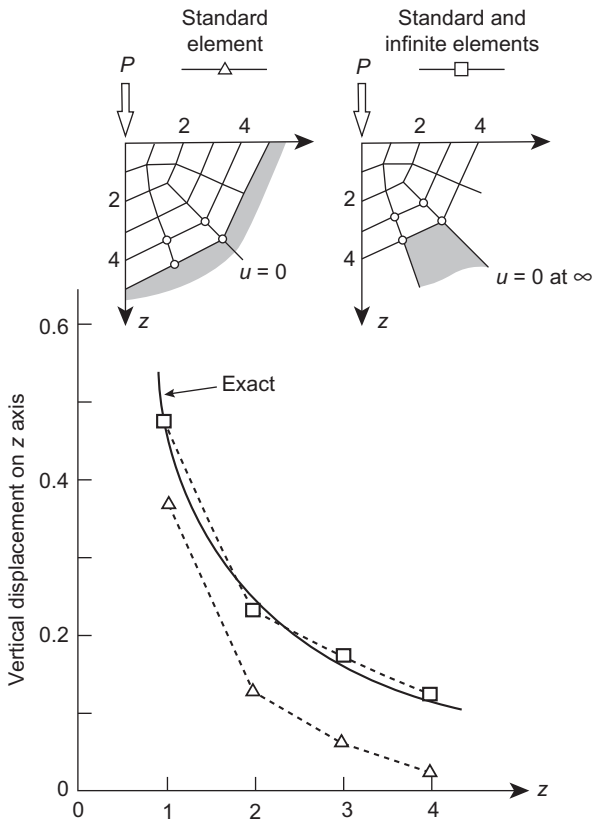



**FIGURE 7.15**

A point load on an elastic half-space (Boussinesq problem). Standard linear elements and infinite line elements ($E = 1, \nu = 0.1, p = 1$).

**Example 7.7.** Infinite element examples

In Fig. 7.15 we show a solution of the Boussinesq problem (a point load on an elastic half-space). Here results of using a fixed displacement or infinite elements are compared and the big changes in the solution noted. In this example the pole of each element was taken at the load point for obvious reasons [9].

It should be remarked that the use of infinite elements (as indeed of any other finite elements) must be tempered by background analytical knowledge and "miracles" should not be expected. Thus the user should not expect, for instance, such excellent results as those shown in Fig. 7.15 for the displacement of a plane elasticity problem. It is "well known" that in this case the displacements under any load which is not self-equilibrated will be infinite everywhere and the numbers obtained from the computation will not be, whereas for the three-dimensional or axisymmetric case a displacement is infinite only at a point load.

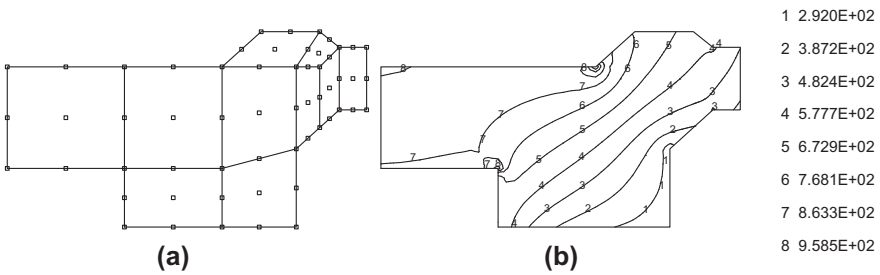



**FIGURE 7.16**
A rotating disc, analyzed with quadratic Lagrangian elements.

Further use of infinite elements is made in the context of the solution of wave problems in fluids in Ref. [50].

### 7.12.1 Practical examples

We now consider some practical examples for problems which have been solved using the finite element method.

**Example 7.8.** Rotating disk

Here only 10 quadratic Lagrangian elements are needed to define the initial mesh arranged as shown in Fig. 7.16. These may then be subdivided to provide a fine mesh of any degree required. The problem requires the specification of body forces caused by the centrifugal effects of the rotating disk. Here,

$$b_r = -\rho\, r\, \omega^2$$

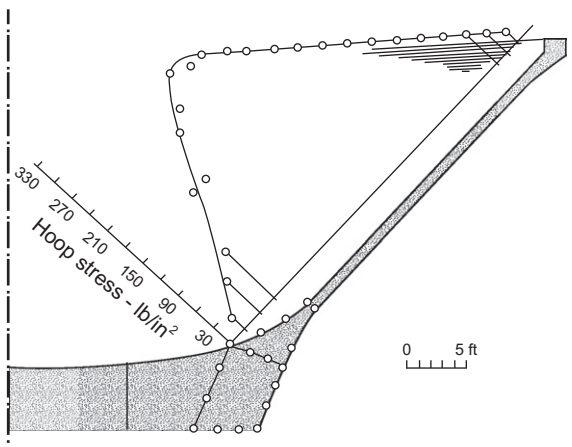



**FIGURE 7.17**
Conical water tank.

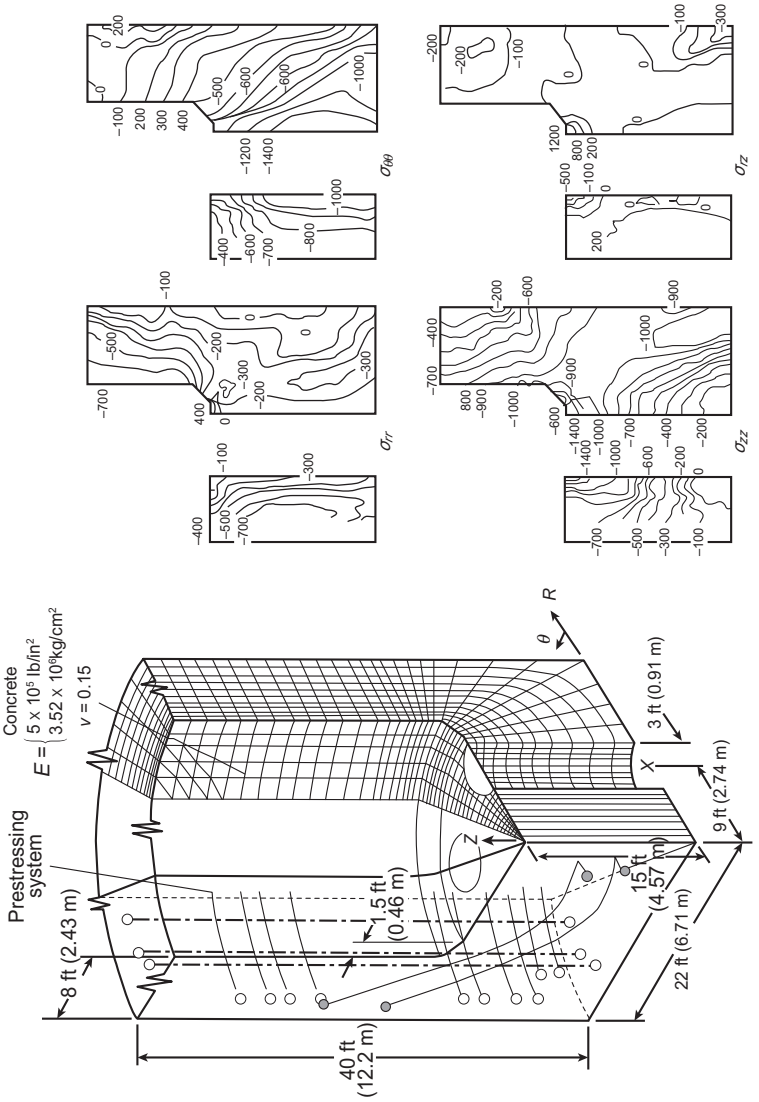



**FIGURE 7.18**
A nuclear pressure vessel analysis using simple tetrahedral elements. Geometry, subdivision, and some stress results. N.B. Not all edges are shown.

where $\rho$ is the mass density of the material and $\omega$ is the angular velocity in rad/s. The material properties are $E = 206$ GPa, $\nu = 0.3$, and $\rho = 7.85 \times 10^{-3}$ kg/m$^3$. The disk rotates at a speed of 22,500 rpm. Figure 7.16b shows contours for the hoop stress $\sigma_\theta$. Here we can notice strong stress gradients at the sharp corners. Indeed, as further refinements are made the computed stresses at sharp corners continue to increase with no upper bound. This is caused by "singular" stresses at these corners thus requiring special treatments. The behavior of stress at sharp corners was studied in some detail in Ref. [11].

**Example 7.9.** Conical water tank

In this problem cubic serendipity elements are used as shown in Fig. 7.17. It is worth noting that a single-element thickness throughout is adequate to represent the bending effects in both the thick and thin parts of the container. With simple four-node quadrilateral elements, several layers of elements are needed to give an adequate solution.

**Example 7.10.** Pressure vessel problem

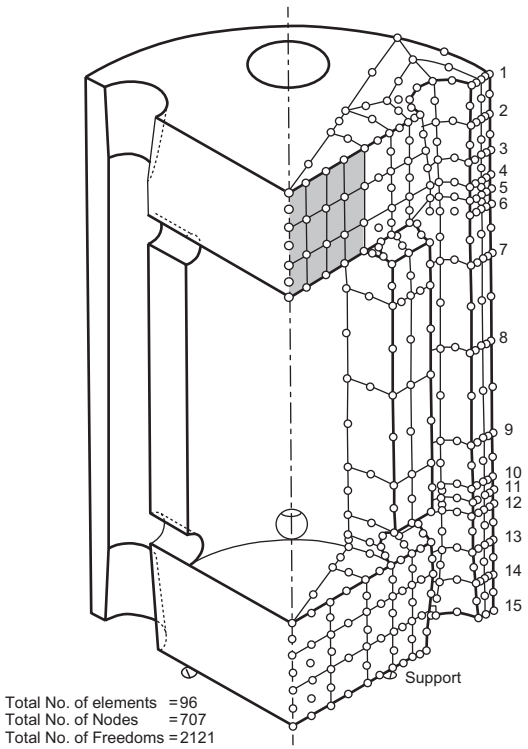



**FIGURE 7.19**
Three-dimensional analysis of a pressure vessel.

Figure 7.18 illustrates an analysis of a complex pressure vessel. Some 10,000 degrees of freedom are involved in this analysis [51]. A similar problem using higher order isoparametric elements permits a sufficiently accurate analysis for a very similar problem to be performed with only 2121 degrees of freedom (Fig. 7.19).

## 7.13 Concluding remarks

The "displacement" approach to the analysis of elastic solids is still undoubtedly the most popular and easily understood procedure. In many of the following chapters we shall use the general formulas developed here in the context of other situations for elastic bodies such as for incompressible problems and mixed problems in which strains and stresses are also approximated. These are also applicable in the context of nonlinear analysis, the main variants being the definitions of the stresses, generalized strains, and other associated quantities [52].

## 7.14 Problems

**7.1** For the triangular element shown in Fig. 7.20a, the dimensions are $a = 3$ cm, and $b = 4$ cm. Compute the shape functions $\mathbf{N}$ for the three nodes of the element.

**7.2** For the rectangular element shown in Fig. 7.20b, the dimensions are $a = 6$ cm, and $b = 4$ cm. Compute the shape functions $\mathbf{N}$ for the four nodes of the element.

**7.3** Use the results from Problem 7.1 to compute the strain-displacement matrix $\mathbf{B}$ for the triangular element shown in Fig. 7.20a.

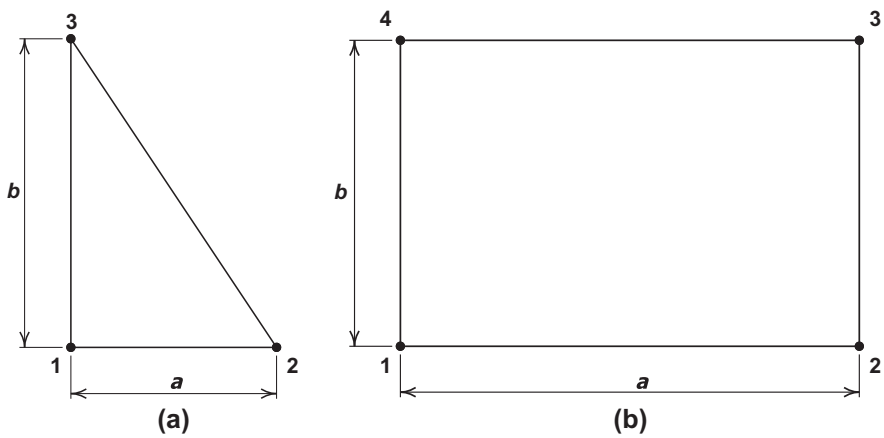



**FIGURE 7.20**
Elements for Problems 7.1–7.4: (a) triangle and (b) rectangle.

**7.4** Use the results from Problem 7.2 to compute the strain-displacement matrix **B** for the rectangular element shown in Fig. 7.20b.

**7.5** The body force vector in a plane stress problem is given by $b_x = 5$ and $b_y = 0$. Using the shape functions determined in Problem 7.1 compute the body force vector for the triangular element shown in Fig. 7.20a.

**7.6** Repeat Problem 7.5 using $b_x = 0$ and $b_y = -30$.

**7.7** The body force vector in a plane stress problem is given by $b_x = 5$ and $b_y = 0$. Using the shape functions determined in Problem 7.2 compute the body force vector for the rectangular element shown in Fig. 7.20b.

**7.8** Repeat Problem 7.6 using $b_x = 0$ and $b_y = -30$.

**7.9** The edge of the triangular element defined by nodes 2 and 3 shown in Fig. 7.20a is to be assigned boundary conditions $u_n = 0$ and $t_s = 0$ where $n$ is a direction normal to the edge and $s$ tangential to the edge. Determine the transformation matrix **L** [viz. Eq. (1.23)] required to transform the nodal degrees of freedom at nodes 2 and 3 to be able to impose the boundary conditions.

**7.10** A concentrated load, $F$, is applied to the edge of a two-dimensional plane strain problem as shown in Fig. 7.21a:

- **(a)** Use equilibrium conditions to compute the statically equivalent forces acting at nodes 1 and 2.
- **(b)** Use the weak form to compute the equivalent forces acting on nodes 1 and 2.

**7.11** A triangular traction load is applied to the edge of a two-dimensional plane strain problem as shown in Fig. 7.21b:

- **(a)** Use equilibrium conditions to compute the statically equivalent forces acting at nodes 1 and 2.
- **(b)** Use the weak form to compute the equivalent forces acting on nodes 1 and 2.

**7.12** For the rectangular and triangular element shown in Fig. 7.22, compute and assemble the stiffness matrices associated with nodes 2 and 5 (i.e., $\mathbf{K}_{22}$, $\mathbf{K}_{25}$, and $\mathbf{K}_{55}$). Let $E = 1000$, $\nu = 0.25$ for the rectangle and $E = 1200$, $\nu = 0$ for the triangle. The thickness for the assembly is constant with $t = 0.2$ cm.

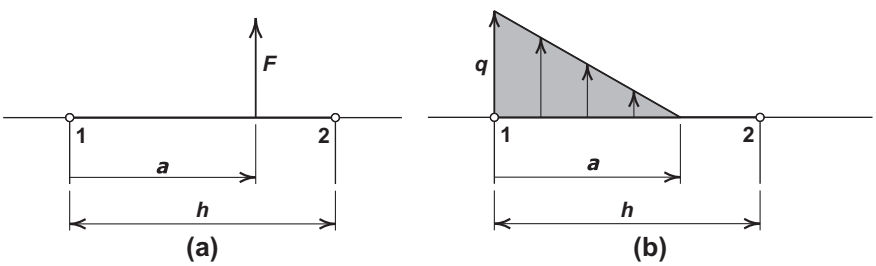



**FIGURE 7.21**
Traction loading on boundary for Problems 7.10 and 7.11.

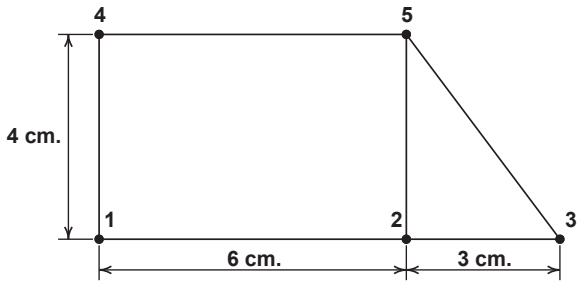



**FIGURE 7.22**
Element assembly for Problem 7.12.

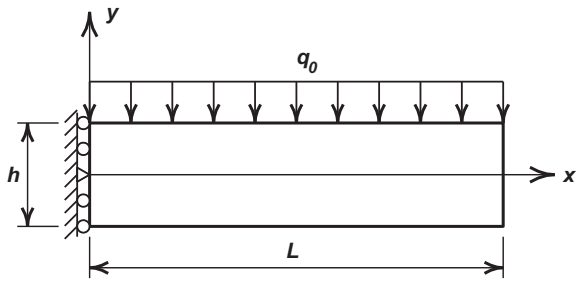



**FIGURE 7.23**
Uniformly loaded cantilever beam for Problem 7.16.

**7.13** Download the program *FEAPpv* and user manual from a website given in Chapter 18. Note that both source code for the program and an executable version for Windows-based systems are available at the site.
If source code is used it is necessary to compile the program to obtain an executable version. Also download a test problem from the web site. Install the program and verify that the test problem is correctly solved.

**7.14** Use *FEAPpv* (or any available program) to solve the rectangular beam problem with an end-loaded shear given in Example 7.5—verify the results shown in Table 7.2.

**7.15** Use *FEAPpv* (or any available program) to solve the curved beam problem given in Example 7.6—verify the results shown in Tables 7.3 and 7.4.

**7.16** The uniformly loaded cantilever beam for shown in Fig. 7.23 has the properties

$$L = 2 \text{ m}, \quad h = 0.4 \text{ m}, \quad t = 0.05 \text{ m} \quad \text{and} \quad q_0 = 100 \text{ N/m}$$

Use *FEAPpv* (or any available program) to perform a plane stress analysis of the problem assuming linear isotropic elastic behavior with $E = 200$ GPa and $\nu = 0.3$.

In your analysis:

- **(a)** Use three-node triangular elements with an initial mesh of two elements in the depth and 10 elements in the length directions.
- **(b)** Compute consistent nodal forces for the uniform loading.
- **(c)** Compute nodal forces for a parabolically distributed shear traction at the restrained end which balances the uniform loading $\mathbf{q_0}$.
- **(d)** Report results for the centerline displacement in the vertical direction and the stored energy in the beam.
- **(e)** Repeat the analysis three additional times using meshes of $4 \times 20$, $8 \times 40$, and $16 \times 80$ elements. Tabulate the tip vertical displacement and stored energy for each solution.
- **(f)** If the energy error is given by

$$\Delta E = E_n - E_{n-1} = C\, h^p$$

estimate $C$ and $p$ for your solution.

- **(g)** Repeat the above analysis using four-node quadrilateral elements.
- **(h)** Repeat the above analysis using eight-node quadrilateral elements.
- **(i)** Repeat the above analysis using nine-node quadrilateral elements.

**7.17** Use the transformation array given by

$$\mathbf{T} = \begin{bmatrix} \cos\theta & sin\,\theta & 0 \\ -\sin\theta & cos\,\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

with $\theta = 45°$ to transform stress and strain components from the $x, y, z$ components to their $x', y', z'$ components. Let the material be linearly elastic with material parameters given by $E$ and $\nu$. Show that $G = E/[2(1+\nu)]$.

**7.18** For an isotropic material expressed in $E$ and $\nu$ compute the mean stress $p = (\sigma_x + \sigma_y + \sigma_z)/3$. If the bulk modulus is given by

$$p = K\, \varepsilon_v$$

where $\varepsilon_v = \varepsilon_x + \varepsilon_y + \varepsilon_z$ is the volume strain, show that $K = E/[3(1-2\nu)]$.

**7.19** The cross-section of two tubular sections is shown in Fig. 7.24. The parts are to be assembled by heating the outer part until it just passes over the inner part as shown in the figure. Let $r_i = 10$ cm, $t = 5$ cm, and $h = 10$ cm and take elastic properties as $E = 200$ GPa, $\nu = 0.3$, and $\alpha = 12 \times 10^{-6}/°\text{C}$. The parts are stress free at the room temperature 20 °C. The parts just fit when the outer bar is heated to 220 °C (while the inner part is maintained at room temperature):

- **(a)** What is the correct inner radius of the outer part at room temperature?
- **(b)** Solve the problem using *FEAPpv* (or any other available program). Use a mesh of four-node quadrilateral elements to compute the final solution for the assembled part at room temperature assuming complete contact at the mating surface and no slip during cooling. Plot the radial displacement at the interaction surface.

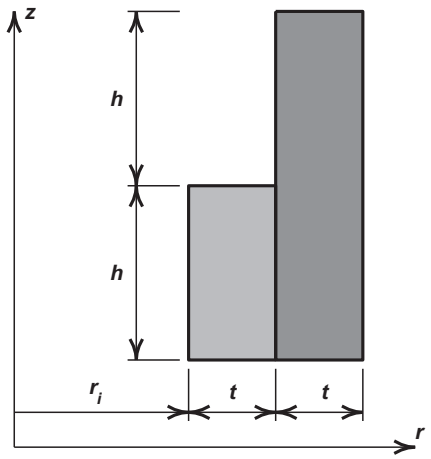



**FIGURE 7.24**
Thermal assembly of tubular sections for Problem 7.19.

**(c)** Compute an estimate of the traction components at the interaction surface. Do you think there will be slip? Why?

**7.20** Program development project:[3] Extend the program started in Problem 7.12 to write a MATLAB [46] or GNU Octave [47] program[4] to solve plane stress problems.
Your program system should have the following features:

**(a)** Input module that describes:
- **(i)** Nodal coordinate values, $\mathbf{x}_a$
- **(ii)** Nodes connected to each element and the material properties of each element
- **(iii)** Node and degree-of-freedom (dof) for each applied nodal force
- **(iv)** Node and dof for a fixed (essential) boundary condition—also the value if nonzero.

**(b)** Module to compute the stiffness matrix for a three-node triangular element.
**(c)** Module to compute the stiffness matrix for a four-node quadrilateral element.
**(d)** Module to assemble element arrays into global arrays and specified nodal forces and displacements.

[3] If programming is included as a part of your study, it is recommended that this problem be solved. Several extensions will be suggested later to create a solution system capable of performing all steps of finite element analysis.
[4] Another programming language may be used, however, MATLAB and GNU Octave offer many advantages to write simple programs and are also useful to easily complete later exercises.

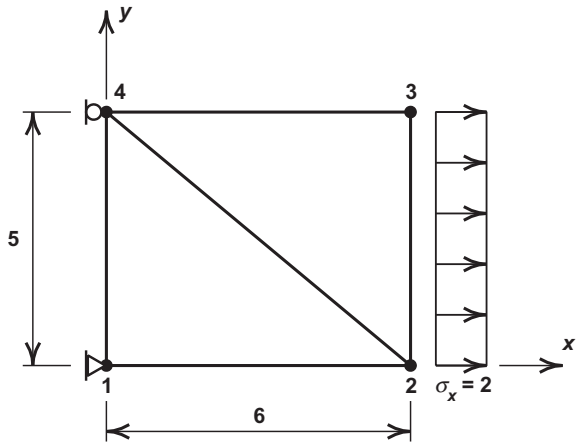



**FIGURE 7.25**
Patch test for triangles for Problem 7.20.

**(e)** Module to solve $\mathbf{K}\,\tilde{\mathbf{u}} + \mathbf{f} = \mathbf{0}$.
**(f)** Module to output nodal displacements and element stress and strains.

Use your program to solve the *patch test* problem shown in Fig. 7.25. Use the properties $E = 2 \times 10^5$, $\nu = 0.3$, and $t = 1$ ($t$ is thickness of slab). You can verify the correctness of your answer by computing an exact solution to the problem. The correctness of computed arrays may be obtained using results from *FEAPpv* (or any available plane stress program).

**7.21** Program development project: Add a graphics capability to the program developed in Problem 7.20 to plot contours of the computed finite element displacements. (Hint: MATLAB has `contour` and `surf` options to easily perform this operation.)

Solve the curved beam problem for the mesh of four-node quadrilaterals shown in Fig. 7.13a. Plot contours for $u$ and $v$ displacements. (Hint: Write a separate MATLAB or GNU Octave program to generate the nodal coordinates and element connections for the simple geometry of the curved beam.) Refine the mesh by increasing the number of segments in each direction by a factor of 2 and repeat the solution of the curved beam problem.

## References

[1] T.J. Mitchell, S. Govindjee, R.L. Taylor, A method for enforcement of Dirichlet boundary conditions in isogeometric analysis, in: Dana Mueller-Hoeppe, Stefan Loehnert, Stefanie Reese (Eds.), Recent Developments and Innovative Applica-

tions in Computational Mechanics, Springer-Verlag, Berlin, Heidelberg, 2010, pp. 283–293.
[2] R.W. Thatcher, On the finite element method for unbounded regions, SIAM J. Numer. Anal. 15 (3) (1978) 466–476.
[3] P. Silvester, D.A. Lowther, C.J. Carpenter, E.A. Wyatt, Exterior finite elements for 2-dimensional field problems with open boundaries, Proc. IEEE 123 (12) (1977).
[4] S.F. Shen, An aerodynamicist looks at the finite element method, in: R.H. Gallagher et al. (Eds.), Finite Elements in Fluids, vol. 2, John Wiley & Sons, New York, 1975, pp. 179–204.
[5] O.C. Zienkiewicz, D.W. Kelley, P. Bettess, The coupling of the finite element and boundary solution procedures, Int. J. Numer. Methods Eng. 11 (1977) 355–375.
[6] P. Bettess, Infinite Elements, Penshaw Press, Cleadon, UK, 1992.
[7] P. Bettess, O.C. Zienkiewicz, Diffraction and refraction of surface waves using finite and infinite elements, Int. J. Numer. Methods Eng. 11 (1977) 1271–1290.
[8] G. Beer, J.L. Meek, Infinite domain elements, Int. J. Numer. Methods Eng. 17 (1981) 43–52.
[9] O.C. Zienkiewicz, C. Emson, P. Bettess, A novel boundary infinite element, Int. J. Numer. Methods Eng. 19 (1983) 393–404.
[10] P. Bettess, Infinite elements, Int. J. Numer. Methods Eng. 11 (1977) 53–64.
[11] M.L. Williams, Stress singularities resulting from various boundary conditions in angular corners of plates in extension, J. Appl. Mech., ASME, 19 (1952) 526–528.
[12] J.J. Oglesby, O. Lomacky, An evaluation of finite element methods for the computation of elastic stress intensity factors, J. Eng. Ind. 95 (1973) 177–183.
[13] J.R. Rice, D.M. Tracey, Computational fracture mechanics, in: S.J. Fenves et al. (Eds.), Numerical and Computer Methods in Structural Mechanics, Academic Press, New York, 1973, pp. 555–624.
[14] A.A. Griffiths, The phenomena of flow and rupture in solids, Philos. Trans. Roy. Soc. (London), Ser. A 221 (1920) 163–198.
[15] D.M. Parks, A stiffness derivative finite element technique for determination of elastic crack tip stress intensity factors, Int. J. Fract. 10 (1974) 487–502.
[16] T.K. Hellen, On the method of virtual crack extensions, Int. J. Numer. Methods Eng. 9 (1975) 187–208.
[17] J.R. Rice, A path-independent integral and the approximate analysis of strain concentration by notches and cracks, J. Appl. Mech., ASME 35 (1968) 379–386.
[18] P. Tong, T.H.H. Pian, On the convergence of the finite element method for problems with singularity, Int. J. Solids Struct. 9 (1972) 313–321.
[19] T.A. Cruse, W. Vanburen, Three-dimensional elastic stress analysis of a fracture specimen with edge crack, Int. J. Fract. Mech. 7 (1971) 1–15.
[20] P.F. Walsh, The computation of stress intensity factors by a special finite element technique, Int. J. Solids Struct. 7 (1971) 1333–1342.
[21] P.F. Walsh, Numerical analysis in orthotropic linear fracture mechanics, Trans. Inst. Eng. Aust. Civ. Eng. 15 (1973) 115–119.

[22] D.M. Tracey, Finite elements for determination of crack tip elastic stress intensity factors, Eng. Fract. Mech. 3 (1971) 255–265.
[23] R.D. Henshell, K.G. Shaw, Crack tip elements are unnecessary, Int. J. Numer. Methods Eng. 9 (1975) 495–509.
[24] R.S. Barsoum, On the use of isoparametric finite elements in linear fracture mechanics, Int. J. Numer. Methods Eng. 10 (1976) 25–38.
[25] R.S. Barsoum, Triangular quarter point elements as elastic and perfectly elastic crack tip elements, Int. J. Numer. Methods Eng. 11 (1977) 85–98.
[26] H.D. Hibbitt, Some properties of singular isoparametric elements, Int. J. Numer. Methods Eng. 11 (1977) 180–184.
[27] S.E. Benzley, Representation of singularities with isoparametric finite elements, Int. J. Numer. Methods Eng. 8 (1974) 537–545.
[28] N. Levy, P.V. Marçal, W.J. Ostergren, J.R. Rice, Small scale yielding near a crack in plane strain: a finite element analysis, Int. J. Fract. Mech. 7 (1967) 143–157.
[29] H.J. Brauchli, J.T. Oden, On the calculation of consistent stress distributions in finite element applications, Int. J. Numer. Methods Eng. 3 (1971) 317–325.
[30] E. Hinton, J. Campbell, Local and global smoothing of discontinuous finite element function using a least squares method, Int. J. Numer. Methods Eng. 8 (1974) 461–480.
[31] O.C. Zienkiewicz, J.Z. Zhu, The superconvergent patch recovery (SPR) and adaptive finite element refinement, Comput. Methods Appl. Mech. Eng. 101 (1992) 207–224.
[32] O.C. Zienkiewicz, J.Z. Zhu, The superconvergent patch recovery (SPR) and a posteriori error estimates. Part 1: The recovery technique, Int. J. Numer. Methods Eng. 33 (1992) 1331–1364.
[33] S. Govindjee, J. Strain, T.J. Mitchell, R.L. Taylor, Convergence of an efficient local least-squares fitting method for bases with compact support, Comput. Methods Appl. Mech. Eng. 213–216 (2012) 84–92, http://dx.doi.org/10.1016/j.cma.2011.11.017.
[34] E.R. de Arantes e Oliveira, Theoretical foundations of the finite element method, Int. J. Solids Struct. 4 (1968) 929–952.
[35] G. Strang, G.J. Fix, An Analysis of the Finite Element Method, Prentice-Hall, Englewood Cliffs, NJ, 1973.
[36] L.F. Richardson, The approximate arithmetical solution by finite differences of physical problems, Trans. Roy. Soc. (London), Ser. A 210 (1910) 307–357.
[37] K. Washizu, Variational Methods in Elasticity and Plasticity, third ed., Pergamon Press, New York, 1982.
[38] F.B. Hildebrand, Methods of Applied Mathematics, second ed., Prentice-Hall, 1965. Reprinted by Dover Publishers, 1992.
[39] Lord Rayleigh (J.W. Strutt), On the theory of resonance, Philos. Trans. Roy. Soc. (London), Ser. A 161 (1870) 77–118.
[40] W. Ritz, Über eine neue Methode zur Lösung gewisser variationsproblem der mathematischen physik, J. Reine Angew. Math. 135 (1908) 1–61.

[41] B. Fraeijs de Veubeke, Displacement and equilibrium models in finite element method, in: O.C. Zienkiewicz, G.S. Holister (Eds.), Stress Analysis, John Wiley & Sons, Chichester, 1965, pp. 145–197 (Chapter 9).
[42] R.L. Fox, E.L. Stanton, Developments in structural analysis by direct energy minimization, J. AIAA 6 (1968) 1036–1044.
[43] F.K. Bogner, R.H. Mallett, M.D. Minich, L.A. Schmit, Development and evaluation of energy search methods in non-linear structural analysis, in: Proceedings of the First Conference Matrix Methods in Structural Mechanics, vol. AFFDL-TR-66-80, Wright Patterson Air Force Base, Ohio, October 1966.
[44] G.P. Bazeley, Y.K. Cheung, B.M. Irons, O.C. Zienkiewicz, Triangular elements in bending—conforming and non-conforming solutions, in: Proceedings of the First Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-66-80, Wright Patterson Air Force Base, Ohio, October 1966, pp. 547–576.
[45] S.P. Timoshenko, J.N. Goodier, Theory of Elasticity, third ed., McGraw-Hill, New York, 1969.
[46] MATLAB, 2012. <www.mathworks.com>.
[47] GNU Octave, 2012. <www.gnu.org/software/octave>.
[48] GNU Octave – Wikipedia, 2012. <http://en.wikipedia.org/wiki/GNUOctave>.
[49] R.L. Taylor, FEAP – A Finite Element Analysis Program: Personal Version, User Manual, University of California, Berkeley. <http://www.ce.berkeley.edu/feap/feappv>.
[50] O.C. Zienkiewicz, R.L. Taylor, P. Nithiarasu, The Finite Element Method for Fluid Dynamics, seventh ed., Elsevier, Oxford, 2013.
[51] Y.R. Rashid, W. Rockenhauser, Pressure vessel analysis by finite element techniques, in: Proceedings of Conference on Prestressed Concrete Pressure Vessels, Institute of Civil Engineering, 1968.
[52] O.C. Zienkiewicz, R.L. Taylor, D. Fox, The Finite Element Method for Solid and Structural Mechanics, seventh ed., Elsevier, Oxford, 2013.

This page is intentionally left blank

CHAPTER

# 8 The Patch Test, Reduced Integration, and Nonconforming Elements

## 8.1 Introduction

We have briefly referred in Chapter 7 to the patch test as a means of assessing convergence of displacement-type elements for elasticity problems in which the shape functions violate continuity requirements. In this chapter we shall deal in more detail with this test which is *applicable to all finite element forms* and will show that

**(a)** it is a *necessary* condition for assessing the convergence of any finite element approximation and further that, if properly extended and interpreted, it can provide,
**(b)** a *sufficient* requirement for convergence,
**(c)** an assessment of the (asymptotic) convergence rate of the element tested,
**(d)** a check on the robustness of the algorithm, and
**(e)** a means of developing new finite element forms which can violate compatibility (continuity) requirements.

While for elements which *a priori* satisfy all the continuity requirements, have correct polynomial expansions, and are exactly integrated such a test is superfluous in principle, it is nevertheless useful as it gives

**(f)** a check that correct programming was achieved.

For all the reasons cited above the patch test since its inception has been and continues to be the most important check for practical finite element codes.

The original test was introduced by Irons et al. [1–3] in a physical way and could be interpreted as a check which ascertained whether a patch of elements (Fig. 8.1) subject to a constant strain reproduced exactly the constitutive behavior of the material and resulted in correct stresses when it became infinitesimally small. If it did, it could then be argued that the finite element model represented the real material behavior and, in the limit, as the size of the elements decreased would therefore reproduce exactly the behavior of the real structure.

Clearly, although this test would only have to be passed when the size of the element patch became infinitesimal, for most elements in which polynomials are used the patch size did not in fact enter the consideration and the requirement that the patch test be passed for any element size became standard.

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00008-3

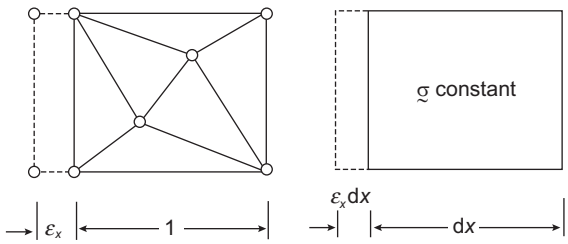



**FIGURE 8.1**
A patch of element and a volume of continuum subject to constant strain $\varepsilon_x$. A physical interpretation of the constant strain or linear displacement field patch test.

Quite obviously a rigid body displacement of the patch would cause no strain, and if the proper constitutive laws were reproduced no stress changes would result. The patch test thus guarantees that no rigid body motion straining will occur.

When curvilinear coordinates are used the patch test is still required to be passed in the limit but generally will not do so for a finite size of the patch. (An exception here is the isoparametric coordinate system in problems discussed in Chapters 3 and 6 since it is guaranteed to contain linear polynomials in the global coordinates.) Thus for many problems such as shells, where local curvilinear coordinates are used, this test has to be restricted to infinitesimal patch sizes and, on physical grounds alone, appears to be a *necessary and sufficient condition* for convergence.

Numerous publications on the theory and practice of the test have followed the original publications cited [4–6] and mathematical respectability was added to those by Strang [7,8]. Although some authors have cast doubts on its validity [9,10] these have been fully refuted [11–13] and if the test is used as described here it fulfills the requirements (a)–(f) stated above.

In the present chapter we consider the patch test applied to irreducible forms (see Chapters 5 and 7) but an extension to mixed forms is more important. This has been studied in Refs. [13–15] and made use of in many subsequent publications. The matter of mixed form patch tests will be fully discussed in the next chapter; however, the consistency and stability tests developed in the present chapter are *always* required.

One additional use of the patch test was suggested by Babuška et al. [16] with a shorter description given by Boroomand and Zienkiewicz [17]. This test can establish the efficiency of gradient (stress) recovery processes which are so important in error estimation as will be discussed in Chapter 15.

## 8.2 Convergence requirements

We shall consider in the following the patch test as applied to a finite element solution of a set of differential equations

$$\mathcal{A}(\mathbf{u}) \equiv \mathcal{L}\mathbf{u} + \mathbf{b} = \mathbf{0} \tag{8.1}$$

in the domain $\Omega$ together with the conditions

$$\mathcal{B}(\mathbf{u}) = \mathbf{0} \tag{8.2}$$

on the boundary of the domain, $\Gamma$.

The finite element approximation is given in the form

$$\mathbf{u} \approx \hat{\mathbf{u}} = \mathbf{N}\tilde{\mathbf{u}} \tag{8.3}$$

where $\mathbf{N}$ are shape functions defined in each element, $\Omega_e$, and $\tilde{\mathbf{u}}$ are unknown parameters.

By applying standard procedures of finite element approximation the problem reduces in a linear case to a set of algebraic equations

$$\mathbf{K}\tilde{\mathbf{u}} = \mathbf{f} \tag{8.4}$$

which when solved give an approximation to the differential equation and its boundary conditions.

What is meant by "convergence" in the approximation sense is that the approximate solution, $\hat{\mathbf{u}}$, should tend to the exact solution $\mathbf{u}$ when the size of the elements $h$ approaches zero (with some specified subdivision pattern). Stated mathematically we must find that the error at any point becomes (when $h$ is sufficiently small)

$$|\mathbf{u} - \hat{\mathbf{u}}| = O(h^q) \leq Ch^q \tag{8.5}$$

where $q > 0$ and $C$ is a positive constant, depending on the position. This must also be true for all the derivatives of $\mathbf{u}$ defined in the approximation.

By the order of convergence in the variable $\mathbf{u}$ we mean the value of the index $q$ in the above definition. To ensure convergence it is necessary that the approximation fulfill both consistency and stability conditions [18].

The *consistency requirement* ensures that as the size of the elements $h$ tends to zero, the approximation equation (8.4) will represent the exact differential equation (8.1) and the boundary conditions (8.2) (at least in the weak sense).

The *stability condition* is simply translated as a requirement that the solution of the discrete equation system (8.4) be unique and avoid spurious mechanisms which may pollute the solution for all sizes of elements. For linear problems in which we solve the system of algebraic equations (8.4) as

$$\tilde{\mathbf{u}} = \mathbf{K}^{-1}\mathbf{f} \tag{8.6}$$

this means simply that the matrix $\mathbf{K}$ must be nonsingular for all possible element assemblies (subject to imposing minimum stable boundary conditions).

The patch test traditionally has been used as a procedure for verifying the consistency requirement; the stability was checked independently by ensuring nonsingularity of matrices [19]. Further, it generally tested only the consistency in satisfaction of the differential equation (8.1) but not of its natural boundary conditions. In what follows we shall show how all the necessary requirements of convergence can be tested by a properly conceived patch test.

A "weak" singularity of a single element may on occasion be permissible and some elements exhibiting it have been, and still are, successfully used in practice. One such case is given by the eight-node isoparametric element with a $2 \times 2$ Gauss quadrature, to which we shall refer later here. This element is on occasion observed to show peculiar behavior (though its use has advantages as discussed in Chapter 9). An element that occasionally fails is termed *nonrobust* and the patch test provides a means of assessing the *degree of robustness*.

## 8.3 The simple patch test (Tests A and B): A necessary condition for convergence

We shall first consider the consistency condition which requires that in the limit (as $h$ tends to zero) the finite element approximation of Eq. (8.4) should model exactly the differential equation (8.1) and the boundary conditions (8.2). If we consider a "small" region of the domain (of size $2h$) we can expand the unknown function $\mathbf{u}$ and the essential derivatives entering the weak approximation in a Taylor series. From this we conclude that for convergence of the function and its first derivative in typical problems of a second-order equation and two dimensions, we require that around a point $a$ assumed to be at the coordinate origin,

$$\begin{aligned} \mathbf{u} &= \mathbf{u}_a + \left(\frac{\partial \mathbf{u}}{\partial x}\right)_a x + \left(\frac{\partial \mathbf{u}}{\partial y}\right)_a y + \cdots + O(h^p) \\ \frac{\partial \mathbf{u}}{\partial x} &= \left(\frac{\partial \mathbf{u}}{\partial x}\right)_a + \cdots + O(h^{p-1}) \\ \frac{\partial \mathbf{u}}{\partial y} &= \left(\frac{\partial \mathbf{u}}{\partial y}\right)_a + \cdots + O(h^{p-1}) \end{aligned} \tag{8.7}$$

with $p \geq 2$. The finite element approximation should therefore reproduce exactly the problem posed for *any linear forms* of $\mathbf{u}$ as $h$ tends to zero. Similar conditions can obviously be written for higher order problems. This requirement is tested by the current interpretation of the patch test illustrated in Fig. 8.2. We refer to this as the *base solution*.

For problems involving $C_0$ approximation we compute first an arbitrary solution of the differential equation using a linear polynomial as the base solution and set the corresponding parameters $\tilde{\mathbf{u}}$ [see Eq. (8.3)] at all "nodes" of a *patch* which assembles completely the nodal variable $\tilde{\mathbf{u}}_a$ (i.e., provides all the equation terms corresponding to it).

In "Test A" we simply insert the exact value of the parameters $\tilde{\mathbf{u}}$ into the $a$th equation and verify that

$$\mathbf{K}_{ab}\tilde{\mathbf{u}}_b - \mathbf{f}_a \equiv \mathbf{0} \tag{8.8}$$

where $\mathbf{f}_a$ is a force which results from any "body force" required to satisfy the differential equation (8.1) for the base solution. Generally in problems given in Cartesian coordinates the required body force is zero; however, in curvilinear coordinates (e.g., axisymmetric elasticity problems) it can be nonzero.

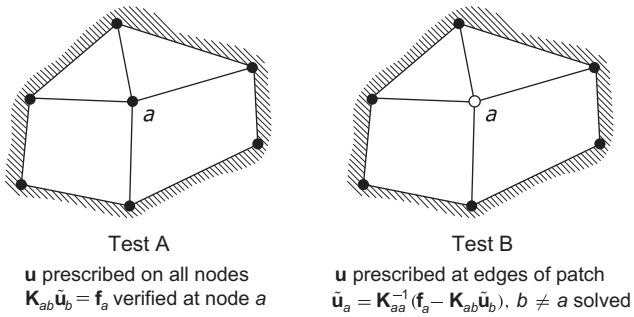



**FIGURE 8.2**
Simple patch test (Tests A and B).

In "Test B" only the values of $\tilde{\mathbf{u}}$ corresponding to the boundaries of the "patch" are inserted and $\tilde{\mathbf{u}}_a$ is found as

$$\tilde{\mathbf{u}}_a = \mathbf{K}_{aa}^{-1}(\mathbf{f}_a - \mathbf{K}_{ab}\tilde{\mathbf{u}}_b), \quad b \neq a \tag{8.9}$$

and compared against the exact value.

Both patch tests verify only the satisfaction of the basic differential equation and not of the boundary approximations, as these have been explicitly excluded here.

We mentioned earlier that the test is, in principle, required only for an infinitesimally small patch of elements; however, for differential equations with constant coefficients the size of the patch is immaterial and the test can be carried out on a patch of arbitrary dimensions.

Indeed, if the coefficients are not constant the same size independence exists providing that a constant set of such coefficients is used in the formulation of the test. This applies, for instance, in axisymmetric problems where coefficients of the type $1/r$ (radius) enter the equations and when the patch test is here applied, it is simply necessary to enter the computation with such quantities assumed constant. Alternatively, a body force can be computed which allows the base solution to satisfy the differential equation exactly.

If mapped curvilinear elements are used it is not obvious that the patch test posed in global coordinates needs to be satisfied. Here, in general, convergence in the mapping coordinates may exist but a finite patch test may not be satisfied. However, once again if we specify the nature of the subdivision without changing the mapping function, in the limit the Jacobian becomes locally constant and the previous remarks apply. To illustrate this point consider, for instance, a set of elements in which local coordinates are simply the polar coordinates as shown in Fig. 8.3. With shape functions using polynomial expansions in the $r, \theta$ terms the patch test of the kind we have described above will not be satisfied with elements of finite size—nevertheless in the limit as the element size tends to zero it will become true. Thus it is evident that patch test satisfaction is a *necessary condition* which always has to be achieved *providing the size of the patch is infinitesimal*.

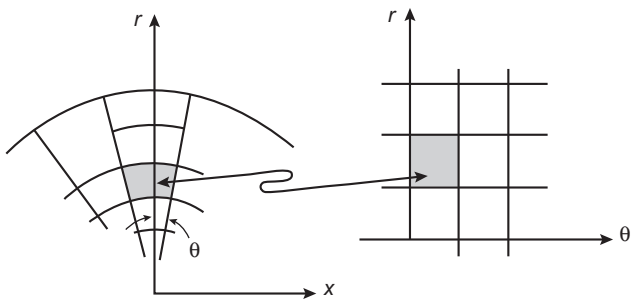



**FIGURE 8.3**
Polar coordinate mapping.

This proviso, which we shall call *weak patch test satisfaction*, is not always simple to verify, particularly if the element coding does not easily permit the insertion of constant coefficients or a constant Jacobian. It is indeed fortunate that the standard isoparametric element form reproduces exactly the linear polynomial global coordinates (see Chapter 6) and for this reason does not require special treatment unless some other *crime* (such as selective or reduced integration) is introduced.

## 8.4 Generalized patch test (Test C) and the single-element test

The patch test described in the preceding section was shown to be a necessary condition for convergence of the formulation but did not establish sufficient conditions for it. In particular, it omitted the testing of the boundary "load" approximation for the case when the "natural" (e.g., "traction of elasticity") conditions are specified. Further it did not verify the stability of the approximation. A test including a check on both of the above conditions is easily constructed. We show this in Fig. 8.4 for a two-dimensional

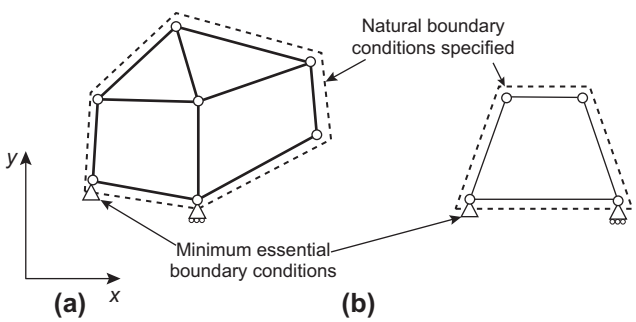



**FIGURE 8.4**
(a) Generalized patch test (Test C). (b) The single-element test.

plane problem as "Test C." In this the patch of elements is assembled as before but subject to prescribed natural boundary conditions (or tractions around its perimeter) corresponding to the base function. The assembled matrix of the whole patch is written as

$$\mathbf{K}\tilde{\mathbf{u}} = \mathbf{f}$$

Fixing only the minimum number of parameters $\tilde{\mathbf{u}}$ necessary to obtain a physically valid solution (e.g., eliminating the rigid body motion in an elasticity example or enforcing a single value of temperature in a heat conduction problem), a solution is sought for the remaining $\tilde{\mathbf{u}}$ values and compared with the exact base solution assumed.

Now any singularity of the $\mathbf{K}$ matrix will be immediately observed and, as the vector $\mathbf{f}$ includes all necessary source and boundary traction terms, the formulation will be completely tested (providing of course a sufficient number of test states is used). The test described is now not only *necessary* but *sufficient* for convergence.

With boundary traction included it is of course possible to reduce the size of the patch to a single element and an alternative form of "Test C" is illustrated in Fig. 8.4b, which is termed the *single-element test* [11]. This test is indeed one requirement of a good finite element formulation as, on occasion, a larger patch may not reveal the inherent instabilities of a single element. This happens in the well-documented case of the plane strain-stress eight-node isoparametric element with (reduced) four-point Gauss quadrature, i.e., where the singular deformation mode of a single element (see Fig. 8.5) disappears when several elements are assembled.[1] *It should be noted, however, that satisfaction of a single-element test is not a sufficient condition for convergence. For sufficiency we require at least one internal element boundary to test that consistency of a patch solution is maintained between elements.*

## 8.5 The generality of a numerical patch test

In the previous section we have defined in some detail the procedures for conducting a patch test. We have also asserted the fact that such tests if passed guarantee that convergence will occur. However, all the tests are numerical and it is impractical to test all possible combinations.

In particular, let us consider the base solutions used. These will invariably be a set of polynomials given in two dimensions as

$$\mathbf{u} = \sum_i \boldsymbol{\alpha}_i P_i(x, y) \tag{8.10}$$

where $P_i$ are a suitable set of low order polynomials (e.g., $1, x, y$ for Galerkin forms possessing only first-order derivatives) and $\boldsymbol{\alpha}_i$ are parameters. It is fairly obvious that if patch tests are conducted on each of these polynomials individually any base function of the form given in Eq. (8.10) can be reproduced and the generality preserved for the particular combination of elements tested. This must always be done and is

[1] This figure also shows a similar singularity for a patch of four four-node (bilinear interpolation) elements with single-point quadrature, and we note the similar shape of zero-energy modes.

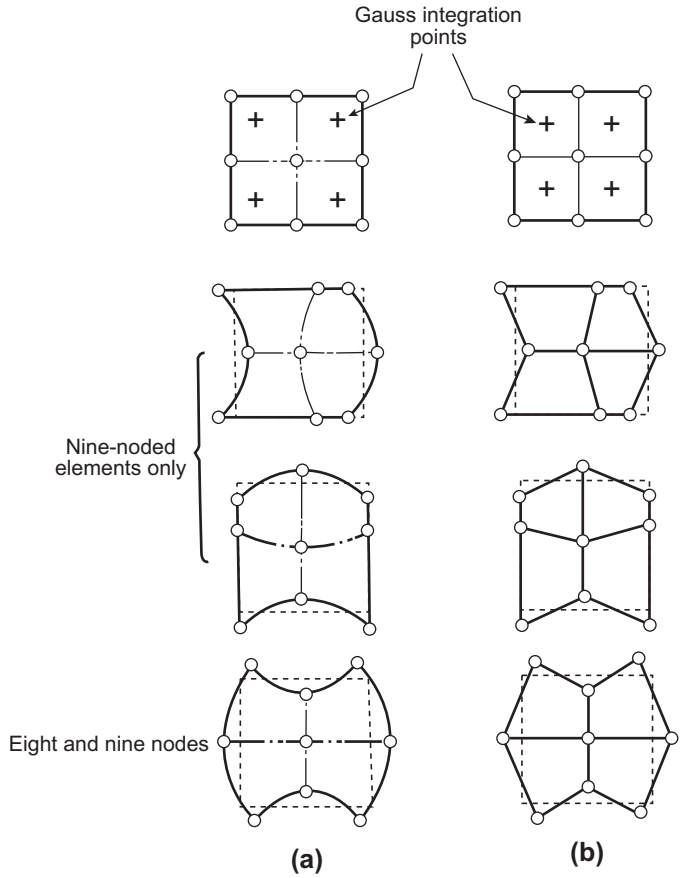



**FIGURE 8.5**

(a) Zero-energy (singular) modes for eight- and nine-node quadratic elements and (b) for a patch of bilinear elements with single integration points.

almost a standard procedure in engineering tests, necessitating only a limited number of combinations.

However, as various possible patterns of elements can occur and it is possible to increase the size without limit the reader may well ask whether the test is complete from the geometrical point of view. We believe it is necessary in a numerical test to consider the possibility of several pathological arrangements of elements but that if the test is purely limited to a single element and a complete patch around a node we can be confident about the performance on more general geometric patterns.

Indeed even mathematical assessments of convergence are subject to limits often imposed *a posteriori*. Such limits may arise if for instance a singular mapping is used.

The procedures referred to in this section should satisfy most readers as to the validity and generality of the test.

On some limited occasions it is possible to perform the test purely algebraically and then its validity cannot be doubted. Some such algebraic tests will be referred to later in connection with incompatible elements.

In this chapter we have only considered linear differential equations and linear material behavior; however, the patch test can well be used and extended to cover nonlinear problems.

## 8.6 Higher order patch tests

While the patch tests discussed in the last three sections ensure (when satisfied) that convergence will occur, they did not test the order of this convergence, beyond assuring us that in the case of Eq. (8.7) the errors were, at least, of order $O(h^2)$ in $\mathbf{u}$. It is an easy matter to determine the actual highest asymptotic rate of convergence of a given element by simply imposing, instead of a linear solution, exact higher order polynomial solutions [6,8]. The highest value of such polynomials for which complete satisfaction of the patch test is achieved automatically evaluates the corresponding convergence rate. It goes without saying that for such exact solutions generally nonzero source (e.g., body force) terms in the original equation (8.1) will need to be involved.

In addition, "Test C" in conjunction with a higher order patch test may be used to illustrate any tendency for "locking" to occur (see Chapter 9). Accordingly, element robustness with regard to various parameters (e.g., Poisson's ratios near one-half for elasticity problems in plane strain) may be established.

In such higher order patch tests it will of course first be assumed that the patch is subject to the base expansion solution as described. Thus, for higher order terms it will be necessary to start and investigate solutions of the type

$$\boldsymbol{\alpha}_3 x^2 + \boldsymbol{\alpha}_4 xy + \boldsymbol{\alpha}_5 y^2 + \cdots$$

Each of these should be applied individually or as linearly independent combinations and for each the solution should be appropriately tested.

In particular, we shall expect higher order elements to exactly satisfy certain order solutions. However in Chapter 15 we shall use this idea to find the error between the exact solution and the recovery using precisely the same type of formulation.

## 8.7 Application of the patch test to plane elasticity elements with "standard" and "reduced" quadrature

In the next few sections we consider several applications of the patch test in the evaluation of finite element models. In each case we consider only one of the necessary tests which need to be implemented. For a complete evaluation of a formulation it is necessary to consider all possible independent base polynomial solutions as well as a variety of patch configurations which test the effects of element distortion or alternative meshing interconnections which will be commonly used in analysis. As we

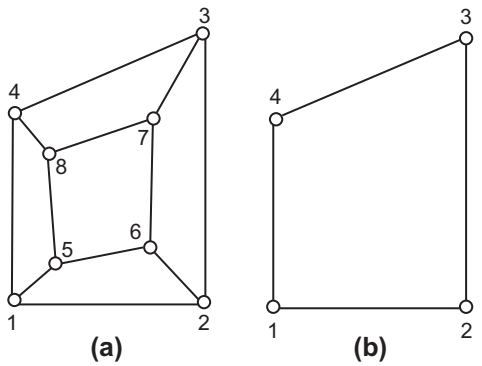



**FIGURE 8.6**
Patch for evaluation of numerically integrated plane stress problems: (a) five-element patch and (b) one-element patch.

shall emphasize, it is important that both consistency and stability be evaluated in a properly conducted test.

In Chapter 6 (Section 6.8) we discussed the minimum required order of numerical integration for various finite element problems which results in no loss of convergence rate. However, it was also shown that for some elements such a minimum integration order results in singular matrices. If we define the *standard* integration as one which evaluates the stiffness of an element exactly[2] (at least in the undistorted form) then any lower order of integration is called *reduced*.

Such *reduced* integration has some merits in certain problems for reasons which we shall discuss in Chapter 10 (Section 10.5), but it can cause singularities which should be discovered by a patch test (which supplements and verifies the arguments of Section 6.8.5).

Application of the patch test to some typical problems will now be shown.

**Example 8.1.** Patch test for base solution

We consider first a plane stress problem on the patch shown in Fig. 8.6a. The material is linear, isotropic elastic with properties $E = 1000$ and $\nu = 0.3$. The finite element procedure used is based on the displacement form using four-node isoparametric shape functions and numerical integration as described in Chapter 6. Since the stiffness computation includes only first derivatives of displacements, the formulation converges provided that the patch test is satisfied for all linear polynomial solutions of displacements in the base solution. Here we consider only one of the six independent linear polynomial solutions necessary to verify satisfaction of the patch test. The solution considered is

$$\begin{aligned} u &= 0.0020\, x \\ v &= -0.0006\, y \end{aligned} \tag{8.11a}$$

[2] An alternate definition for standard integration is the lowest order of integration for which the rank of the stiffness matrix does not increase.

**Table 8.1** Patch Solution for Fig. 8.6

| Node $a$ | Coordinates | | Computed Displacements | | Forces | |
|---|---|---|---|---|---|---|
| | $x_a$ | $y_a$ | $u_a$ | $v_a$ | $F_{x_a}$ | $F_{y_a}$ |
| 1 | 0.0 | 0.0 | 0.0 | 0.0 | −2 | 0 |
| 2 | 2.0 | 0.0 | 0.0040 | 0.0 | 3 | 0 |
| 3 | 2.0 | 3.0 | 0.0040 | −0.00186 | 2 | 0 |
| 4 | 0.0 | 2.0 | 0.0 | −0.00120 | −3 | 0 |
| 5 | 0.4 | 0.4 | 0.0008 | −0.00024 | 0 | 0 |
| 6 | 1.4 | 0.6 | 0.0028 | −0.00036 | 0 | 0 |
| 7 | 1.5 | 2.0 | 0.0030 | −0.00120 | 0 | 0 |
| 8 | 0.3 | 1.6 | 0.0006 | −0.00096 | 0 | 0 |

which produces zero body forces and zero stresses except for

$$\sigma_x = 2 \tag{8.11b}$$

The solution given in Table 8.1 is obtained for the nodal displacements and satisfies Eq. (8.11a) exactly.

The patch test is performed first using $2 \times 2$ Gaussian "standard" quadrature to compute each element stiffness and resulting reaction forces at nodes. For Test A all nodes are restrained and nodal displacement values are specified according to Table 8.1. Stresses are computed at specified Gauss points ($1 \times 1, 2 \times 2$, and $3 \times 3$ Gauss points were sampled) and all are exact to within round-off error (double precision was used which produced round-off errors less than $10^{-15}$ in the quantities computed). Reactions were also computed at all nodes and again produced the force values shown in Table 8.1 to within round-off limits. This approximation satisfies all conditions required for a finite element procedure (i.e., conforming shape functions and standard order quadrature). Accordingly, the patch test merely verifies that the programming steps used contain no errors. Test A does not require explicit use of the stiffness matrix to compute results; consequently, the above patch test was repeated using Test B where only nodes 1–4 are restrained with their displacements specified according to Table 8.1. This tests the accuracy of the stiffness matrix and, as expected, exact results are once again recovered to within round-off errors. Finally, Test C was performed with node 1 fully restrained and node 4 restrained only in the $x$-direction. Nodal forces were applied to nodes 2 and 3 in accordance with the values generated through the boundary tractions by $\sigma_x$ (i.e., nodal forces shown in Table 8.1). This test also produced exact solutions for all other nodal quantities in Table 8.1 and recovered $\sigma_x$ of 2 at all Gauss points in each element.

The above test was repeated for Tests A, B, and C but using a $1 \times 1$ "reduced" Gauss quadrature to compute the element stiffness and nodal force quantities. Test C indicated that the global stiffness matrix contained two global "zero-energy modes" (i.e., the global stiffness matrix was rank deficient by 2), thus producing incorrect nodal displacements whose results depend solely on the round-off errors in the calculations.

These in turn produced incorrect stresses except at the $1 \times 1$ Gauss point used in each element to compute the stiffness and forces. Thus, based upon stability considerations, the use of $1 \times 1$ quadrature on four-node elements produces a failure in the patch test. The element does satisfy consistency requirements, however, and provided a proper stabilization scheme is employed (e.g., stiffness or viscous methods are used in practice) this element may be used for practical calculations [20,21].

It should be noted that a one-element patch test may be performed using the mesh shown in Fig. 8.6b. The results are given by nodes 1–4 in Table 8.1. For the one-element patch, Tests A and B coincide and neither evaluates the accuracy nor stability of the stiffness matrix. On the other hand, a Type C patch test leads to the conclusions reached using the five-element patch: namely, $2 \times 2$ Gaussian quadrature passes a patch test whereas $1 \times 1$ quadrature fails the stability part of the test (as indeed we would expect by the arguments of Chapter 6, Section 6.8.5).

A simple test on cancellation of a diagonal during the triangular decomposition step is sufficient to warn of rank deficiencies in the stiffness matrix.

**Example 8.2.** Patch test for quadratic elements: Quadrature effects

In Fig. 8.7 we show a two-element patch of quadratic isoparametric quadrilaterals. Both eight-node serendipity and nine-node Lagrangian types are considered and Test C is performed for load case 1. For the eight-node element both $2 \times 2$ ("reduced") and $3 \times 3$ ("standard") Gaussian quadrature satisfy the patch test, whereas for the nine-node element only $3 \times 3$ quadrature is satisfactory, with $2 \times 2$ reduced quadrature leading to failure in rank of the stiffness matrix. However, if we perform a one-element test for the eight-node and $2 \times 2$ quadrature element, we discover the spurious zero-energy mode shown in Fig. 8.5 and thus the one-element test has failed. We consider such elements suspect and to be used only with the greatest of care. To illustrate what can happen in practice we consider the simple problem shown in Fig. 8.8a. In this example the "structure" modeled by a single element is considered rigid and interest

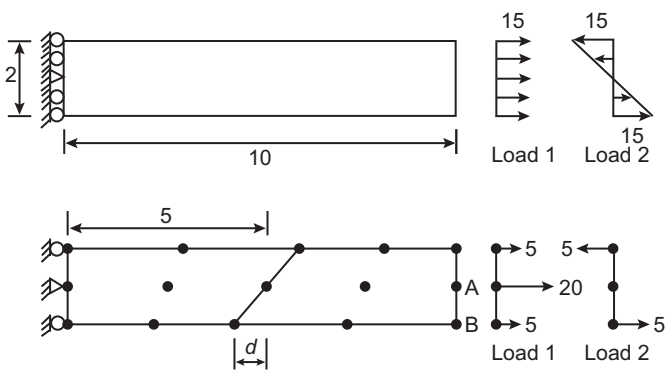



**FIGURE 8.7**
Patch test for eight- and nine-node isoparametric quadrilaterals.

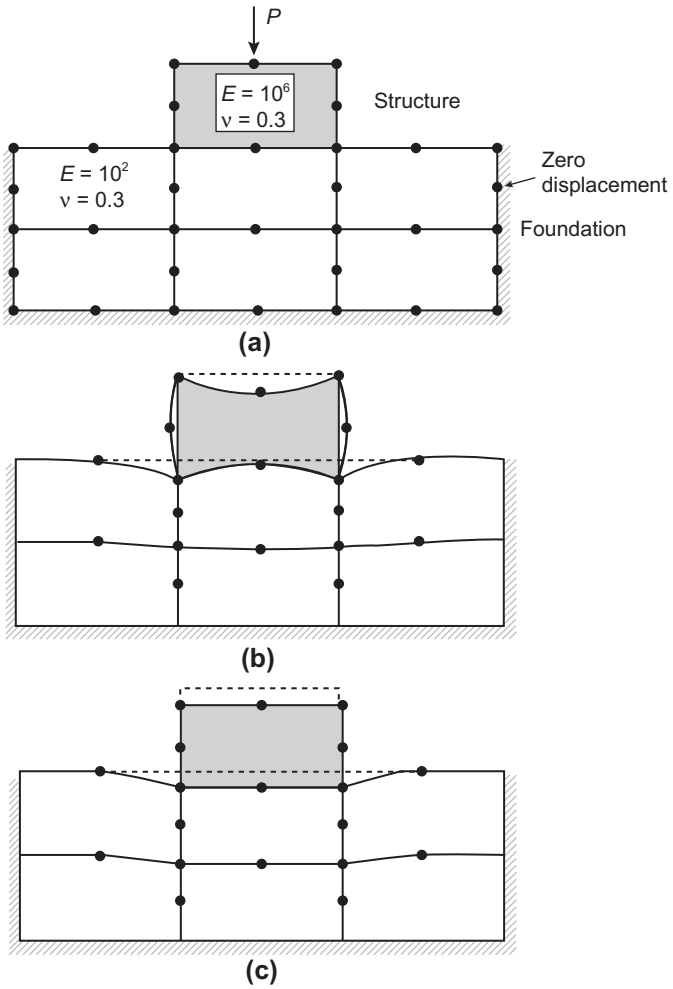



**FIGURE 8.8**
A propagating spurious mode from a single unsatisfactory element: (a) problem and mesh, (b) $2 \times 2$ integration, and (c) $3 \times 3$ integration.

is centered on the "foundation" response. Accordingly only one element is used to model the structure. Use of $2 \times 2$ quadrature throughout leads to answers shown in Fig. 8.8b while results for $3 \times 3$ quadrature are shown in Fig. 8.8c. It should be noted that no zero-energy mode exists since more than one element is used. There is, however, a spurious response due to the large modulus variation between structure and foundation. This suggests that problems in which nonlinear response may lead to a large variation in material parameters could also induce such performance, and thus use of the eight-node $2 \times 2$ integrated element should always be closely monitored to detect such anomalous behavior.

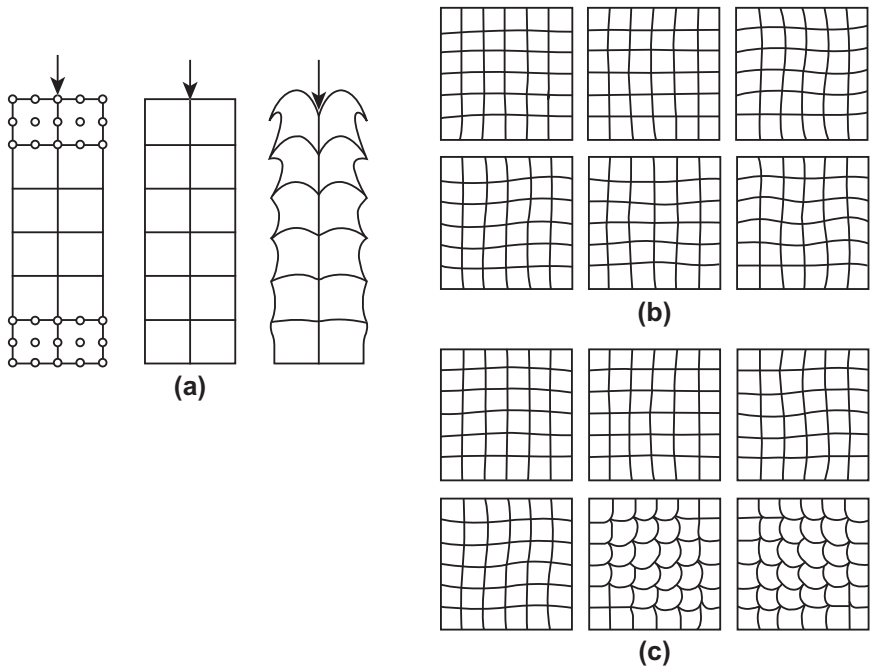



**FIGURE 8.9**

Peculiar response of near singular assemblies of elements. (a) A column of nine-node elements with point load response of full $3\times3$ and $2\times2$ integration. The whole assembly is nonsingular but singular element modes are apparent. (b) A fully constrained assembly of nine-node elements with no singularity—first six eigenmodes with full $(3\times3)$ integration. (c) Same as (b) but with $2\times2$ integration. Note the appearance of "wild" modes called "Escher" modes after the graphic artist.

Indeed, support or loading conditions may themselves induce very suspect responses for elements in which near singularity occurs. Figure 8.9 shows some amusing peculiarities which can occur for reduced integration elements and which disappear entirely if full integration is used [22]. In all cases the *assembly* of elements is nonsingular even though individual elements are rank deficient.

**Example 8.3.** Higher order patch test—assessment of order

In order to demonstrate a higher order patch test we consider the two-element plane stress problem shown in Fig. 8.7 subjected to bending loading shown as Load 2. As above, two different types of element are considered: (a) an eight-node serendipity quadrilateral element and (b) a nine-node Lagrangian quadrilateral element. In our test we wish to demonstrate a feature for nine-node element mapping discussed in Chapter 6 (see Section 6.6) and first shown by Wachspress [23]. In particular we restrict the mapping into the $xy$ plane to be that produced by the four-node isoparametric bilinear

**Table 8.2** Bending Load Case ($E = 100, \nu = 0.3$)

| Element | Quadrature | $d$ | $v_A$ | $u_B$ | $v_B$ |
|---|---|---|---|---|---|
| 8-node | $3 \times 3$ | 0 | 0.750 | 0.150 | 0.75225 |
| 8-node | $2 \times 2$ | | 0.750 | 0.150 | 0.75225 |
| 9-node | $3 \times 3$ | | 0.750 | 0.150 | 0.75225 |
| 8-node | $3 \times 3$ | 1 | 0.7448 | 0.1490 | 0.74572 |
| 8-node | $2 \times 2$ | | 0.750 | 0.150 | 0.75100 |
| 9-node | $3 \times 3$ | | 0.750 | 0.150 | 0.75225 |
| 8-node | $3 \times 3$ | 2 | 0.6684 | 0.1333 | 0.66364 |
| 8-node | $2 \times 2$ | | 0.750 | 0.150 | 0.75225 |
| 9-node | $3 \times 3$ | | 0.750 | 0.150 | 0.75225 |
| Exact | – | – | 0.750 | 0.150 | 0.75225 |

element, but permit the dependent variable to assume the full range of variations consistent with the eight- or nine-node shape functions. In Chapter 6 we showed that the nine-node element can approximate a complete quadratic displacement function in $x, y$ whereas the eight-node element cannot. Thus we expect that the nine-node element when restricted to the isoparametric mappings of the four-node element will pass a higher order patch test for all arbitrary quadratic displacement fields. The pure bending solution in elasticity is composed of polynomial terms up to quadratic order. Furthermore, no body force loadings are necessary to satisfy the equilibrium equations. For the mesh considered the nodal loadings are equal and opposite on the top and bottom nodes as shown in Fig. 8.7. The results for the two elements are shown in Table 8.2 for the indicated quadratures with $E = 100$ and $\nu = 0.3$.

From this test we observe that the nine-node element does pass the higher order test performed. Indeed, provided the mapping is restricted to the four-node shape it will always pass a patch test for displacements with terms no higher than quadratic. On the other hand, the eight-node element passes the higher order patch test performed only for rectangular element (or constant Jacobian) mappings. Moreover, the accuracy of the eight-node element deteriorates very rapidly with increased distortions defined by the parameter $d$ in Fig. 8.7.

The use of $2 \times 2$ reduced quadrature improves results for the higher order patch test performed. Indeed, two of the points sampled give exact results and the third is only slightly in error. As noted previously, however, a single-element test for the $2 \times 2$ integrated eight-node element will fail the stability part of the patch test and it should thus be used with great care.

## 8.8 Application of the patch test to an incompatible element

In order to demonstrate the use of the patch test for a finite element formulation which violates the usually stated requirements for shape function continuity, we consider the plane strain incompatible modes first introduced by Wilson et al. [24] and discussed by

Taylor et al. [25] The specific incompatible formulation considered uses the element displacement approximations

$$\hat{\mathbf{u}} = \mathbf{N}_a \tilde{\mathbf{u}}_a + N_1^n \boldsymbol{\alpha}_1 + N_2^n \boldsymbol{\alpha}_2 \tag{8.12}$$

where $\mathbf{N}_a$ $(a = 1, \ldots, 4)$ are the usual conforming bilinear shape functions and the last two terms are *incompatible modes of deformation* defined by the hierarchical functions

$$N_1^n = 1 - \xi^2 \quad \text{and} \quad N_2^n = 1 - \eta^2 \tag{8.13}$$

defined independently for each element.

The shape functions used are shown in Fig. 8.10. The first, a set of standard bilinear type (Fig. 8.10a), gives a displacement pattern that introduces spurious shear strains in pure bending as shown in the right part of Fig. 8.10b. The second, in which the parameters $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$ for shape functions (8.12) of a specific element (Fig. 8.10c), introduces incompatibility but assures correct bending behavior in an individual rectangular element (Fig. 8.10b, left side). The excellent performance of this element in the bending situation is illustrated in Fig. 8.11.

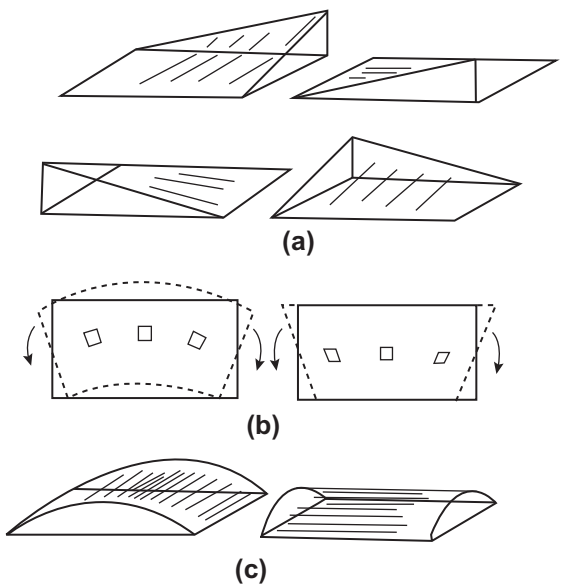
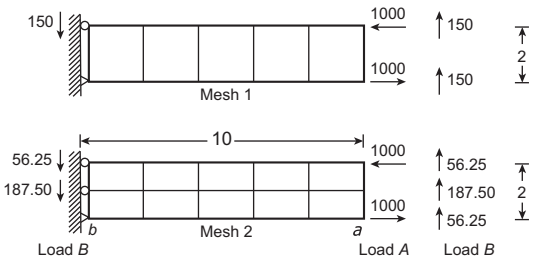



**FIGURE 8.10**

(a) Linear quadrilateral with auxiliary incompatible shape functions; (b) pure bending and linear displacements causing shear; and (c) auxiliary "bending" shape functions with internal variables.




| | | Displacement at *a* | | Stress at *b* | |
|---|---|---|---|---|---|
| | | Load *A* | Load *B* | Load *A* | Load *B* |
| Beam theory | | 10.00 | 103.0 | 300.0 | 4050 |
| (a) | Mesh 1 | 6.81 | 70.1 | 218.2 | 2945 |
| | Mesh 2 | 7.06 | 72.3 | 218.8 | 2954 |
| (b) | Mesh 1 | 10.00 | 101.5 | 300.0 | 4050 |
| | Mesh 2 | 10.00 | 101.3 | 300.0 | 4050 |

**FIGURE 8.11**

Performance of the nonconforming quadrilateral in beam bending treated as plane stress: (a) conforming linear quadrilateral and (b) nonconforming quadrilateral.

In Ref. [25] the finite element approximation is computed by summing the potential energies of each element and computing the nodal loads due to boundary tractions from the conforming part of the displacement field only. Thus for the purposes of conducting patch tests we compute the strains using all parts of the displacement field leading to a generalization of (8.4) which may be written as

$$\begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}} \\ \boldsymbol{\alpha} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{Bmatrix} \tag{8.14}$$

Here $\mathbf{K}_{11}$ and $\mathbf{f}_1$ are the stiffness and loads of the four-node (conforming) bilinear element, $\mathbf{K}_{12}$ and $\mathbf{K}_{21}$ $(=\mathbf{K}_{12}^{\mathrm{T}})$ are coupling stiffnesses between the conforming and nonconforming displacements, and $\mathbf{K}_{22}$ and $\mathbf{f}_2$ are the stiffness and loads of the nonconforming displacements. We note that, according to the algorithm of Ref. [24], $\mathbf{f}_2$ must vanish from the patch test solutions.

For a patch test in plane strain or plane stress, only linear polynomials need be considered for which all nonconforming displacements must vanish. Thus for a successful patch test we must have

$$\mathbf{K}_{11}\tilde{\mathbf{u}} = \mathbf{f}_1 \tag{8.15a}$$

and

$$\mathbf{K}_{21}\tilde{\mathbf{u}} = \mathbf{f}_2 \tag{8.15b}$$

If we carry out a patch test for the mesh shown in Fig. 8.12a we find that all three forms (i.e., patch tests A, B, and C) satisfy these conditions and thus pass the

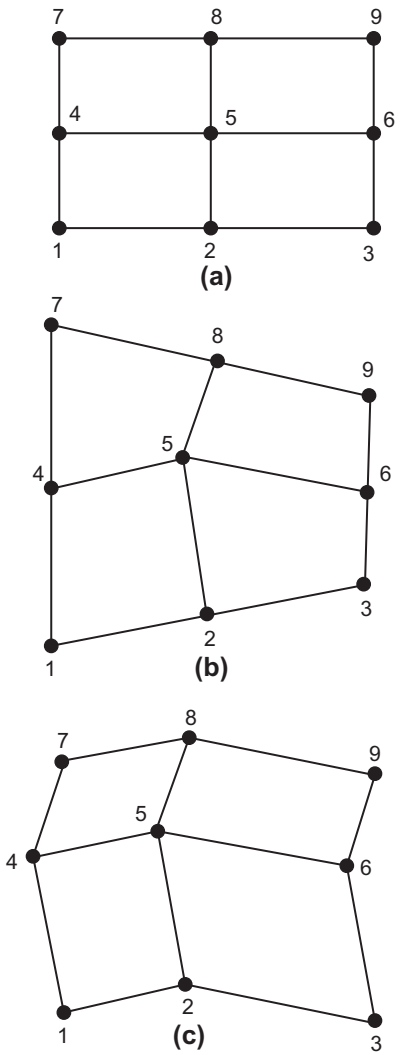



**FIGURE 8.12**
Patch test for an incompatible element form: (a) regular discretization, (b) irregular discretization about node 5, and (c) constant Jacobian discretization about node 5.

patch test. If we consider the patch shown in Fig. 8.12b, however, the patch test is not satisfied. The lack of satisfaction shows up in different ways for each form of the patch test. Test A produces nonzero $\mathbf{f}_2$ values when $\boldsymbol{\alpha}$ is set to zero and $\tilde{\mathbf{u}}$ according to the displacements considered. In form B the values of the nodal displacements $\tilde{\mathbf{u}}_5$ are in error and $\boldsymbol{\alpha}$ are nonzero, also leading to erroneous stresses in each element. In form C all unspecified displacements are in error as well as the stresses.

It is interesting to note that when a patch is constructed according to Fig. 8.12c in which all elements are parallelograms all three forms of the patch test are once again satisfied. Accordingly we can note that if any mesh is systematically refined by subdivision of each element into four elements whose sides are all along $\xi, \eta$ lines in the original element with values of $-1$, 0, or 1 (i.e., by bisections), the mesh converges to constant Jacobian approximations of the type shown in Fig. 8.12c. Thus, in this special case the incompatible mode element satisfies a weak patch test and will converge. In general, however, it may be necessary to use a very fine discretization to achieve sufficient accuracy, and hence the element probably has no practical (nor efficient) engineering use.

A simple artifice to ensure that an element passes the patch test is to replace the derivatives of the incompatible modes by

$$\begin{Bmatrix} \dfrac{\partial N_a^n}{\partial x} \\ \dfrac{\partial N_a^n}{\partial y} \end{Bmatrix} = \frac{J_0}{J(\xi,\eta)} \mathbf{J}_0^{-1} \begin{Bmatrix} \dfrac{\partial N_a^n}{\partial \xi} \\ \dfrac{\partial N_a^n}{\partial \eta} \end{Bmatrix} \tag{8.16}$$

where $J(\xi, \eta)$ is the determinant of the Jacobian $\mathbf{J}(\xi, \eta)$ and $\mathbf{J}_0^{-1}$ and $J_0$ are the values of the inverse Jacobian and Jacobian determinant evaluated at the element center ($\xi = \eta = 0$). This ensures satisfaction of the patch test for all element shapes, and with this alteration of the algorithm the incompatible element proves convergent and quite accurate [25].

An alternative approach which also passes the patch test constructs the derivatives used in the strains as

$$\begin{aligned} \frac{\partial N_a^n}{\partial x} &\Leftarrow \frac{\partial N_a^n}{\partial x} - \frac{1}{\Omega_e} \int_{\Omega_e} \frac{\partial N_a^n}{\partial x} \, \mathrm{d}\Omega \\ \frac{\partial N_a^n}{\partial y} &\Leftarrow \frac{\partial N_a^n}{\partial y} - \frac{1}{\Omega_e} \int_{\Omega_e} \frac{\partial N_a^n}{\partial y} \, \mathrm{d}\Omega \end{aligned} \tag{8.17}$$

where $\Omega_e$ is the volume of the element [26]. Indeed, this form may also be used to deduce terms in the strain matrix of enhanced strain forms (e.g., see Section 9.5.3) and the justification of the modification follows from the mixed approach used there. When the shape functions (8.13) are inserted into (8.17) and the Jacobian determinant is constant (as it will be for any parallelogram shape element) we immediately find that the integral term is zero. However, when the element has a nonconstant Jacobian which for the two-dimensional element has the form

$$J(\xi, \eta) = J_0 + J_\xi \xi + J_\eta \eta$$

where $J_0, J_\xi, J_\eta$ are constants depending on nodal coordinates of the element, the integral term is nonzero. Thus, the effectiveness of the modification is clearly evident in producing elements which pass the constant stress patch test for all element shapes.

## 8.9 Higher order patch test: Assessment of robustness

A higher order patch test may also be used to assess element "robustness." An element is termed robust if its performance is not sensitive to physical parameters of the differential equation. For example, the performance of many elements for solution of plane strain linear elasticity problems is sensitive to Poisson's ratio values near 0.5 (called "near incompressibility"). Indeed, for Poisson ratios near 0.5 the energy stored by a unit volumetric strain is many orders larger than the energy stored by a unit deviatoric strain. Accordingly finite elements which exhibit a strong coupling between volumetric and deviatoric strains often produce poor results in the nearly incompressible range, a problem discussed further in Chapter 10.

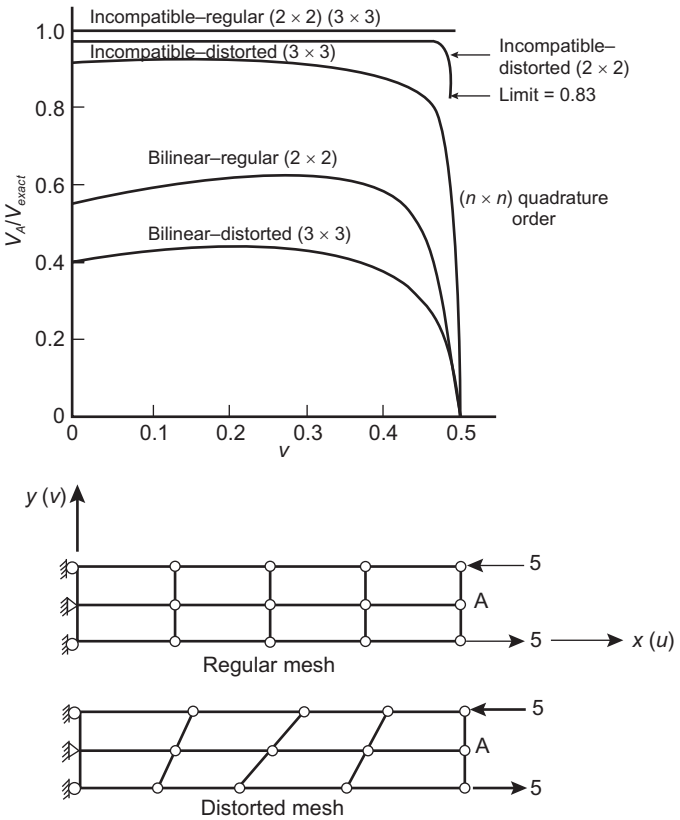



**FIGURE 8.13**
Plane strain four-node quadrilaterals with and without incompatible modes (higher order patch test for performance evaluation).

This may be observed using a four-node element to solve a problem with a quadratic displacement field (i.e., a higher order patch test). If we again consider a pure bending example and an eight-element mesh shown in Fig. 8.13, we can clearly observe the deterioration of results as Poisson's ratio approaches a value of one-half. Also shown in Fig. 8.13 are results for the incompatible modes described in Section 8.8. It is evident that the response is considerably improved by adding these modes, especially if $2 \times 2$ quadrature is used.

If we consider the regular mesh and four-node elements and further keep the domain constant and successively refine the problem using meshes of 8, 32, 128, and 512 elements, we observe that the answers do converge as guaranteed by the patch test. However, as shown in Fig. 8.14, the rate of convergence in energy for Poisson ratio values of 0.25 and 0.4999 is quite different. For 0.25 the rate of convergence is nearly a straight line for all meshes, whereas for 0.4999 the rate starts out quite low and never reaches the asymptotic value of 2 for the meshes considered. For $\nu$ near 0.25 the element is called robust, whereas for $\nu$ near 0.5 it is not. If we use selective reduced integration in which deviatoric strain terms are evaluated by standard order quadrature while volumetric terms use reduced quadrature (which for the plane strain case passes strong patch tests) and repeat the experiment, both values of $\nu$ produce a similar response and thus the element becomes robust for all values of Poisson's ratio less than 0.5.

The use of higher order patch tests can thus be very important to separate robust elements from nonrobust elements. For methods which seek to automatically refine a mesh adaptively in regions with high errors, as discussed in Chapter 16, it is extremely important to use robust elements.

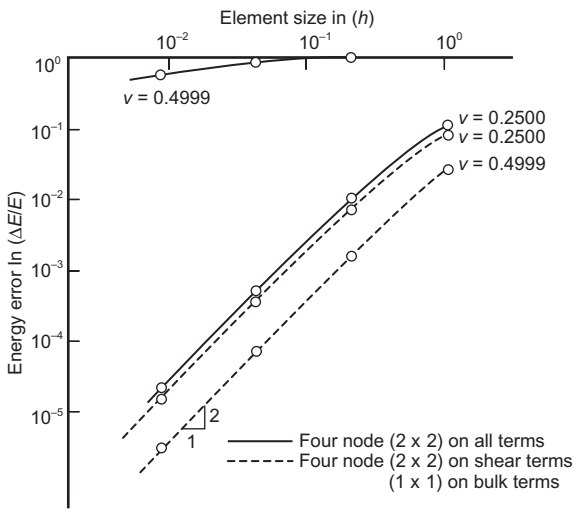



**FIGURE 8.14**

Higher order patch test on element robustness (see Fig. 8.13): convergence test under subdivision of elements.

## 8.10 Concluding remarks

In the preceding sections we have described the patch test and its use in practice by considering several example problems. The patch test described has two essential parts: (a) *a consistency evaluation* and (b) *a stability check*. In the consistency test a set of linearly independent essential polynomials (i.e., all independent terms up to the order needed to describe the finite element model) is used as a solution to the differential equations and boundary conditions, and in the limit as the size of a patch tends to zero the finite element model must exactly satisfy each solution. We presented three forms to perform this portion of the test which we call forms A, B, and C.

The use of form C, where all boundary conditions are the natural ones (e.g., tractions for elasticity) except for the minimum number of essential conditions needed to ensure a unique solution to the problem (e.g., rigid body modes for elasticity), is recommended to test consistency and stability simultaneously. Both one-element and more than one-element tests are necessary to ensure that the patch test is satisfied. With these conditions and assuming that the solution procedure used can detect any possible rank deficiencies the stability of the solution is also tested. If no such condition is included in the program a stability test must be conducted independently. This can be performed by computing the number of zero eigenvalues in the coefficient matrix for methods that use a solution of linear equations to compute the finite element parameters, $\tilde{\mathbf{u}}$. Alternatively, the loading used for the patch solution may be perturbed at one point by a small value (say square root of the round-off limit, e.g., by $10^{-8}$ for round-off of order $10^{-15}$) and the solution tested to ensure that it does not change by a large amount.

Once an element has been shown to pass all of the essential patch tests for both consistency and stability, convergence is assured as the size of elements tends to zero. However, in some situations (e.g., the nearly incompressible elastic problem) convergence may be very slow until a very large number of elements is used. Accordingly, we recommend that higher order patch tests be used to establish element robustness. Higher order patch tests involve the use of polynomial solutions of the differential equation and boundary conditions with the order of terms larger than the basic polynomials used in a patch test. Indeed, the order of polynomials used should be increased until the patch test is satisfied only in a weak sense (i.e., as $h$ tends to zero). The advantage of using a higher order patch test, as opposed to other boundary value problems, is that the exact solution may be easily computed everywhere in the model.

In some of the examples we have tested the use of incompatible function and inexact numerical integration procedures (reduced and selective integration). Some of these violations of the rules previously stipulated have proved justified not only by yielding improved performance but also by providing methods for which convergence is guaranteed. We shall discuss in Chapter 10 some of the reasons for such improved performance.

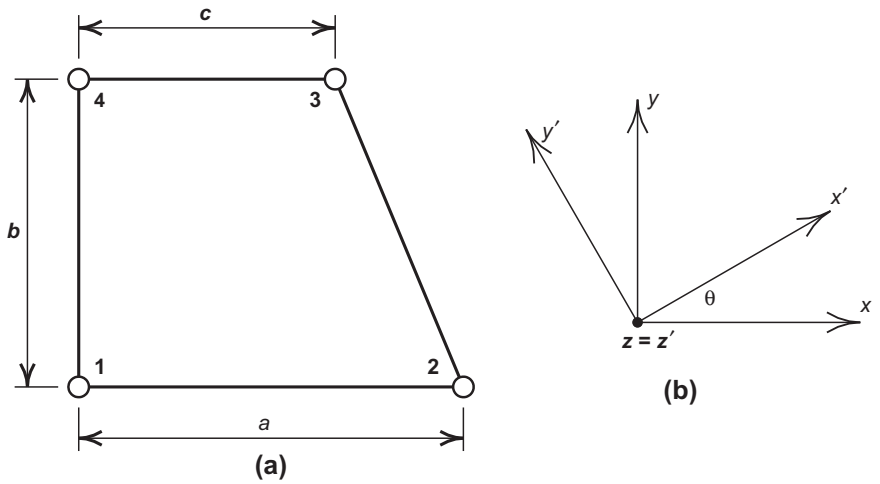



**FIGURE 8.15**
One-element patch test for four-node quadrilateral for Problems 8.1–8.4: (a) element geometry and (b) orthotropic axes.

## 8.11 Problems

**8.1** A patch test of form C for a plane strain problem is to be performed using the single element shown in Fig. 8.15a.
Assume an element has the dimensions $a = 15, b = 12, c = 10$ with elastic properties $E = 200, \nu = 0.25$. Nodes 1 and 2 are placed on the $x$-axis and $u_1 = v_1 = v_2 = 0$ are applied as boundary restraints:

- **(a)** Compute all nodal forces necessary to compute the test for a stress state $\sigma_x = 8$ with all other stresses zero.
- **(b)** Compute the displacements $u(x, y)$ and $v(x, y)$ for the solution.
- **(c)** Use *FEAPpv* (or any other available program) to perform the test. Is it passed?

**8.2** Solve Problem 8.1 for an axisymmetric geometry[3] with node 1 satisfying $v_1 = 0$ and all other nodes free to displace. For $\sigma_r = 8$ consider the following cases:

- **(a)** Node 1 placed at $r = 0$
- **(b)** Node 1 placed at $r = 15$

[3] *Note: FEAPpv* computes all axisymmetric arrays on a 1-radian sector in the $\theta$-direction, thus avoiding the $2\pi$ factor in a complete ring sector.

**8.3** Solve Problem 8.1 for a plane stress problem with an orthotropic material given by

$$\begin{Bmatrix} \sigma_{x'} \\ \sigma_{y'} \\ \tau_{x'y'} \end{Bmatrix} = \begin{bmatrix} 200 & 50 & 0 \\ 50 & 100 & 0 \\ 0 & 0 & 75 \end{bmatrix} \begin{Bmatrix} \varepsilon_{x'} \\ \varepsilon_{y'} \\ \gamma_{x'y'} \end{Bmatrix}$$

Let nodes 1 and 2 lie on the $x$-axis with node 1 placed at the origin:

**(a)** Compute the nodal forces acting on all nodes when the orthotropic axes are aligned as shown in Fig. 8.15b with $\theta = 30°$ and a single stress $\sigma'_x = 5$ is applied.

**(b)** Compute the displacement field for the case $u_1 = v_1 = u_4 = 0$.

**(c)** Use *FEAPpv* (or any other available program) to perform the patch test. Is it passed?

**8.4** The example described in Problem 8.1 is used to perform a patch test on an element with incompatible modes. The element matrix is given by

$$\begin{bmatrix} \mathbf{K} & \mathbf{C} \\ \mathbf{C}^T & \mathbf{V} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}} \\ \boldsymbol{\alpha} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_u \\ \mathbf{f}_\alpha \end{Bmatrix}$$

where

$$\mathbf{K} = \int_{\Omega_e} \mathbf{B}_u^T \mathbf{D} \mathbf{B}_u \, \mathrm{d}\Omega$$

$$\mathbf{C} = \int_{\Omega_e} \mathbf{B}_u^T \mathbf{D} \mathbf{B}_\alpha \, \mathrm{d}\Omega$$

$$\mathbf{V} = \int_{\Omega_e} \mathbf{B}_\alpha^T \mathbf{D} \mathbf{B}_\alpha \, \mathrm{d}\Omega$$

The element passes the patch test for all constant stress states when $a = c$; however, it fails when $a \neq c$.

Suggest a correction which will ensure the patch test is satisfied.

**8.5** Perform a patch test for the eight-node element shown in Fig. 8.16a for the assumed displacements

$$u = 0.1\,x \quad \text{and} \quad v = 0$$

Let the origin be at the lower left corner of the element. The dimensions are $a = b = 3$ and $c = 3.3$. The material is linear isotropic elastic with $E = 200$ and $\nu = 0.3$ and plane strain conditions are assumed:

**(a)** Use $2 \times 2$ Gaussian quadrature to compute the element arrays and conduct patch tests using forms A, B, and C.

**(b)** Repeat the calculation using $3 \times 3$ quadrature.

**(c)** Consider a higher order displacement

$$u = 0.2\,x\,y \quad \text{and} \quad v = 0$$

and repeat (a) and (b).

**(d)** Discuss any differences noted.

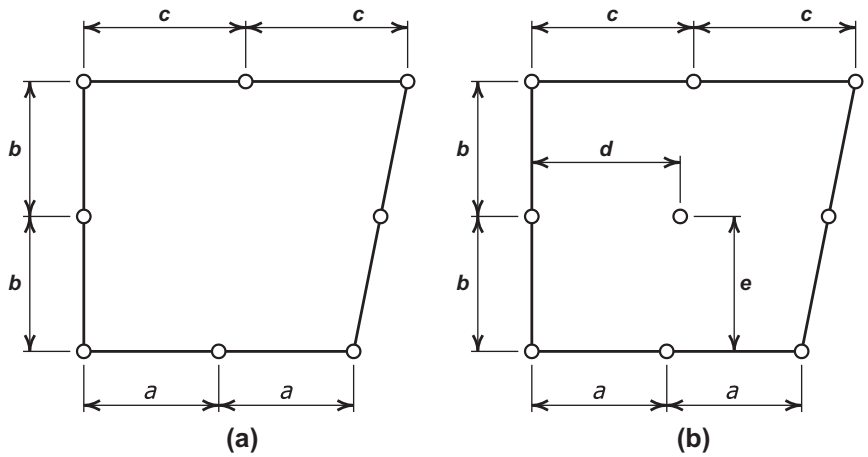



**FIGURE 8.16**
One-element patch test for eight- and nine-node quadrilateral for Problems 8.5–8.8.

**8.6** Perform a patch test for the nine-node element shown in Fig. 8.16b for the assumed displacements

$$u = 0.1\,x \quad \text{and} \quad v = 0$$

Let the origin be at the lower left corner of the element. The dimensions are $a = b = e = 3$, $c = 3.3$, and $d = 3.15$. The material is linear isotropic elastic with $E = 200$ and $\nu = 0.3$ and plane strain conditions are assumed:

**(a)** Use $2 \times 2$ Gaussian quadrature to compute the element arrays and conduct patch tests of forms A, B, and C.
**(b)** Repeat the calculation using $3 \times 3$ quadrature.
**(c)** Set $d = 3.4$ and $e = 2.9$ and repeat (a) and (b).
**(d)** Consider a higher order displacement

$$u = 0.2\,x\,y \quad \text{and} \quad v = 0$$

and repeat (a)–(c).
**(e)** Discuss any differences noted.

**8.7** Solve Problem 8.5 for an axisymmetric geometry (replace $x$, $y$ by $r$, $z$):

**(a)** Let the inner radius be located at $r = 0$.
**(b)** Let the inner radius be located at $r = 3$.

**8.8** Solve Problem 8.6 for an axisymmetric geometry (replace $x$, $y$ by $r$, $z$):

**(a)** Let the inner radius be located at $r = 0$.
**(b)** Let the inner radius be located at $r = 3$.

**8.9** For the four-element mesh configurations shown in Fig. 8.17 devise a set of patch tests for a plane strain problem in which individual constant stress

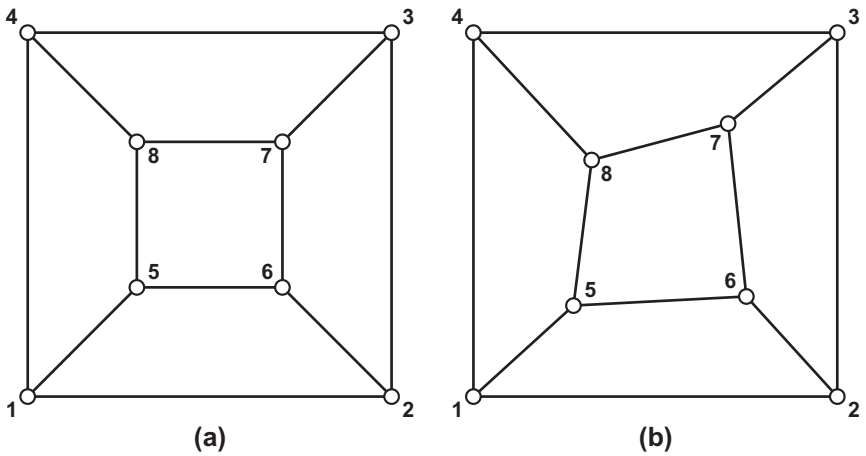



**FIGURE 8.17**
Multielement patch test. Problems 8.9–8.13: (a) regular and (b) distorted.

components are evaluated. Choose appropriate dimensions and isotropic elastic properties with $\nu \neq 0$.
Use *FEAPpv* (or any available program) to perform patch tests of forms A, B, and C for the arrays evaluated by (a) $1 \times 1$ quadrature and (b) $2 \times 2$ quadrature. Discuss your findings.

**8.10** Each quadrilateral subregion in Fig. 8.17 is to be represented by an eight-node isoparametric serendipity element. Each side of the region has a length of 10 units. A higher order patch test of a plane strain problem with isotropic material with $E = 200$ and $\nu = 0$ has the displacements

$$u = -0.1\,x\,y \quad \text{and} \quad v = 0.05\,x^2$$

- **(a)** Compute the state of stress for the given displacement field.
- **(b)** Select appropriate positions for nodes 5–8 for configurations (a) and (b). Specify appropriate nodal boundary conditions to prevent rigid body motion. (Hint: Place the origin of coordinates at the midpoint between nodes 1 and 4.)
- **(c)** Compute appropriate nodal forces and perform a patch test of form C for each configuration using $3 \times 3$ Gaussian quadrature to compute arrays.

Briefly discuss your findings.

**8.11** Solve Problem 8.10 using nine-node isoparametric Lagrangian elements.

**8.12** Replace $x$ by $r$ and $y$ by $z$ and solve Problem 8.10 using eight-node isoparametric serendipity elements on an axisymmetric geometry.

**8.13** Replace $x$ by $r$ and $y$ by $z$ and solve Problem 8.12 using nine-node isoparametric Lagrangian elements.

**8.14** Construct the generalization of the mesh configuration shown in Fig. 8.17 to a three-dimensional problem. For $E = 200$, $\nu = 0.25$, and equal side lengths of 10 units use eight-node isoparametric hexagonal elements to perform a patch test of form C for the single stress $\sigma_z = 5$. Use both regular and distorted positions for the internal nodes.
Hint: Check that there are no negative Jacobian determinants at the nodes of each element.

**8.15** Select dimensions and use *FEAPpv* (or any available program) to verify the results shown in Fig. 8.8.

**8.16** Select dimensions and use *FEAPpv* (or any available program) to verify the results shown in Fig. 8.9a.

**8.17** Select dimensions and use *FEAPpv* (or any available program) to verify the results shown in Fig. 8.9b and c.

## References

[1] B.M. Irons, Numerical integration applied to finite element methods, in: Proceedings of the Conference on Use of Digital Computers in Structural Engineering, University of Newcastle, 1966.

[2] G.P. Bazeley, Y.K. Cheung, B.M. Irons, O.C. Zienkiewicz, Triangular elements in bending—conforming and non-conforming solutions, in: Proceedings of the First Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-66-80, Wright Patterson Air Force Base, Ohio, October 1966, pp. 547–576.

[3] B.M. Irons, A. Razzaque, Experience with the patch test for convergence of finite elements, in: A.K. Aziz (Ed.), The Mathematics of Finite Elements with Application to Partial Differential Equations, Academic Press, New York, 1972, pp. 557–587.

[4] B. Fraeijs de Veubeke, Variational principles and the patch test, Int. J. Numer. Methods Eng. 8 (1974) 783–801.

[5] G. Sander, Bournes supérieures et inérieures dans l'analyse matricielle des plates en flexion-torsion, Bull. Soc. Royale Sci. Liège 33 (1974) 456–494.

[6] E.R. de Arantes Oliveira, The patch test and the general convergence criteria of the finite element method, Int. J. Solids Struct. 13 (1977) 159–178.

[7] G. Strang, Variational crimes and the finite element method, in: A.K. Aziz (Ed.), Proceedings of Foundations of the Finite Element Method, Academic Press, New York, 1972, pp. 689–710.

[8] G. Strang, G.J. Fix, An Analysis of the Finite Element Method, Prentice-Hall, Englewood Cliffs, NJ, 1973.

[9] F. Stummel, The limitations of the patch test, Int. J. Numer. Methods Eng. 15 (1980) 177–188.

[10] J. Robinson et al., Correspondence on patch test, Fin. Elem. News 1 (1982) 30–34.

[11] R.L. Taylor, O.C. Zienkiewicz, J.C. Simo, A.H.C. Chan, The patch test—a condition for assessing FEM convergence, Int. J. Numer. Methods Eng. 22 (1986) 39–62.
[12] R.E. Griffiths, A.R. Mitchell, Non-conforming elements, in: Mathematical Basis of Finite Element Methods, Institute of Mathematics and Its Applications Conference Series, Clarendon Press, Oxford, 1984, pp. 41–69.
[13] O.C. Zienkiewicz, R.L. Taylor, The finite element patch test revisited: a computer test for convergence, validation and error estimates, Comput. Methods Appl. Mech. Eng. 149 (1997) 523–544.
[14] O.C. Zienkiewicz, S. Qu, R.L. Taylor, S. Nakazawa, The patch test for mixed formulations, Int. J. Numer. Methods Eng. 23 (1986) 1873–1883.
[15] W.X. Zhong, FEM patch test and its convergence, Technical Report 97-3001, Research Institute Engineering Mechanics, Dalian University of Technology, 1997 (in Chinese).
[16] I. Babuška, T. Strouboulis, C.S. Upadhyay, A model study of the quality of a posteriori error estimators for linear elliptic problems. Error estimation in the interior of patchwise uniform grids of triangles, Comput. Methods Appl. Mech. Eng. 114 (1994) 307–378.
[17] B. Boroomand, O.C. Zienkiewicz, An improved REP recovery and the effectivity robustness test, Int. J. Numer. Methods Eng. 40 (1997) 3247–3277.
[18] A. Ralston, A First Course in Numerical Analysis, McGraw-Hill, New York, 1965.
[19] B.M. Irons, S. Ahmad, Techniques of Finite Elements, Horwood, Chichester, 1980.
[20] D. Kosloff, G.A. Frasier, Treatment of hour glass patterns in low order finite element codes, Int. J. Numer. Anal. Methods Geomech. 2 (1978) 57–72.
[21] T. Belytschko, W.E. Bachrach, The efficient implementation of quadrilaterals with high coarse mesh accuracy, Comput. Methods Appl. Mech. Eng. 54 (1986) 276–301.
[22] N. Bićanić, E. Hinton, Spurious modes in two dimensional isoparametric elements, Int. J. Numer. Methods Eng. 14 (1979) 1545–1557.
[23] E.L. Wachspress, High order curved finite elements, Int. J. Numer. Methods Eng. 17 (1981) 735–745.
[24] E.L. Wilson, R.L. Taylor, W.P. Doherty, J. Ghaboussi, Incompatible displacement models, in: S.T. Fenves et al. (Eds.), Numerical and Computer Methods in Structural Mechanics, Academic Press, New York, 1973, pp. 43–57.
[25] R.L. Taylor, P.J. Beresford, E.L. Wilson, A non-conforming element for stress analysis, Int. J. Numer. Methods Eng. 10 (1976) 1211–1219.
[26] A. Ibrahimbegovic, E.L. Wilson, A modified method of incompatible modes, Commun. Numer. Methods Eng. 7 (1991) 187–194.

CHAPTER

# 9 Mixed Formulation and Constraints: Complete Field Methods

## 9.1 Introduction

The set of differential equations from which we start the discretization process will determine whether we refer to the formulation as *mixed* or *irreducible*. Thus if we consider an equation system with several dependent variables $\mathbf{u}$ written as

$$\mathcal{A}(\mathbf{u}) = \mathbf{0} \quad \text{in domain } \Omega \tag{9.1a}$$

and

$$\mathcal{B}(\mathbf{u}) = \mathbf{0} \quad \text{on boundary } \Gamma \tag{9.1b}$$

in which none of the components of $\mathbf{u}$ can be eliminated still leaving a well-defined problem, then the formulation will be termed *irreducible*. If this is not the case the formulation will be called *mixed*. These definitions were given in earlier chapters.

This definition is not the only one possible [1] but appears to the authors to be widely applicable [2,3] if in the elimination process referred to we are allowed to introduce penalty functions. Further, for any given physical situation we shall find that more than one irreducible form is usually possible.

As an example we shall consider the simple two-dimensional problem of heat conduction (or the quasi-harmonic equation) to which we have referred in Chapters 2, 3, and 5. In this we start with a physical constitutive relation defining the flux [see Eq. (2.83)] in terms of the potential (temperature) gradients, i.e.,

$$\mathbf{q} = -\mathbf{k}\,\nabla\phi, \quad \mathbf{q} = \begin{Bmatrix} q_x \\ q_y \end{Bmatrix} \tag{9.2}$$

The continuity equation can be written as [see Eq. (2.78)]

$$\nabla^{\mathrm{T}}\mathbf{q} \equiv \frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} = -Q \tag{9.3}$$

If the above equations are satisfied in $\Omega$ and the boundary conditions

$$\phi = \bar{\phi} \quad \text{on } \Gamma_\phi \qquad \text{or} \qquad q_n = \bar{q}_n \quad \text{on } \Gamma_q \tag{9.4}$$

are obeyed then the problem is solved.

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00009-5

Clearly elimination of the vector $\mathbf{q}$ is possible and simple substitution of Eq. (9.2) into Eq. (9.3) leads to

$$-\nabla^{\mathrm{T}}(\mathbf{k}\nabla\phi) + Q = 0 \quad \text{in } \Omega \tag{9.5}$$

with appropriate boundary conditions expressed in terms of $\phi$ or its gradient.

In Chapter 5 we showed discretized solutions starting from this point and clearly, as no further elimination of variables is possible, the formulation is *irreducible*.

On the other hand, if we start the discretization from Eqs. (9.2)–(9.4) the formulation would be *mixed*.

An alternative irreducible form is also possible in terms of the variables $\mathbf{q}$. Here we have to introduce a penalty form and write in place of Eq. (9.3)

$$\nabla^{\mathrm{T}}\mathbf{q} + Q = \frac{\phi}{\alpha} \tag{9.6}$$

where $\alpha$ is a penalty number which tends to infinity. Clearly in the limit both equations are the same and in general if $\alpha$ is very large but finite the solutions should be approximately the same.

Now substitution into Eq. (9.2) gives the single governing equation

$$\nabla(\nabla^{\mathrm{T}}\mathbf{q}) + \frac{1}{\alpha}\mathbf{k}^{-1}\mathbf{q} + \nabla Q = \mathbf{0} \tag{9.7}$$

which again could be used for the start of a discretization process as a possible irreducible form [4].

The reader should observe that, by the definition given, the formulations used so far in this book are *irreducible*. In subsequent sections we will show how elasticity problems can be dealt with in *mixed* form and indeed will show how such formulations are essential in certain problems typified by the incompressible elasticity example to which we have referred in Chapters 2 and 7.

Before proceeding to a discussion of mixed type discretization (which will reveal the advantages and disadvantages of mixed methods) it is important to observe that if the operator specifying the mixed form is *symmetric* or *self-adjoint* (see Section 4.3) the formulation can proceed from the basis of a *variational principle* which can be directly obtained for linear problems. We invite the reader to prove by using the methods of Chapter 4 that stationarity of the *variational principle* given below is equivalent to the differential equations (9.2) and (9.3) together with the boundary conditions (9.4):

$$\Pi = \frac{1}{2}\int_{\Omega}\mathbf{q}^{\mathrm{T}}\mathbf{k}^{-1}\mathbf{q}\,\mathrm{d}\Omega + \int_{\Omega}\mathbf{q}^{\mathrm{T}}\nabla\phi\,\mathrm{d}\Omega - \int_{\Omega}\phi\,Q\,\mathrm{d}\Omega - \int_{\Gamma_q}\phi\bar{q}_n\,\mathrm{d}\Gamma \tag{9.8}$$

for

$$\phi = \bar{\phi} \quad \text{on } \Gamma_{\phi}$$

The establishment of such variational principles is a worthy academic pursuit and had led to many famous forms given in the classical work of Washizu [5]. However, we also know (see Section 3.7.2) that if symmetry of weighted residual matrices is obtained in a linear problem then a variational principle exists and can be determined. As such symmetry can be established by inspection we shall, in what follows, proceed with such weighting directly and thus avoid some unwarranted complexity.

## 9.2 Mixed form discretization: General remarks

We shall demonstrate the discretization process on the basis of the mixed form of the heat conduction equations (9.2) and (9.3). Here we start by assuming that each of the unknowns is approximated in the usual manner by appropriate shape functions and corresponding unknown parameters. Thus,

$$\mathbf{q} \approx \hat{\mathbf{q}} = \mathbf{N}_q \tilde{\mathbf{q}} \qquad \text{and} \qquad \phi \approx \hat{\phi} = \mathbf{N}_\phi \tilde{\boldsymbol{\phi}} \tag{9.9}$$

where $\tilde{\mathbf{q}}$ and $\tilde{\boldsymbol{\phi}}$ are the nodal (or element) parameters that have to be determined. Similarly the weighting functions are given by

$$\delta\mathbf{q} \approx \delta\hat{\mathbf{q}} = \mathbf{W}_q \delta\tilde{\mathbf{q}} \qquad \text{and} \qquad \delta\phi \approx \delta\hat{\phi} = \mathbf{W}_\phi \delta\tilde{\boldsymbol{\phi}} \tag{9.10}$$

where $\delta\tilde{\mathbf{q}}$ and $\delta\tilde{\boldsymbol{\phi}}$ are arbitrary parameters.

Assuming that the boundary conditions for $\phi = \bar{\phi}$ are satisfied by the choice of the expansion, the weighted statement of the problem is, for Eq. (9.2) after elimination of the arbitrary parameters,

$$\int_\Omega \mathbf{W}_q^{\mathrm{T}} (\mathbf{k}^{-1}\hat{\mathbf{q}} + \nabla\hat{\phi}) \mathrm{d}\Omega = \mathbf{0} \tag{9.11}$$

and, for Eq. (9.3) and the "natural" boundary conditions,

$$-\int_\Omega \mathbf{W}_\phi^{\mathrm{T}} (\nabla^{\mathrm{T}}\hat{\mathbf{q}} + Q) \mathrm{d}\Omega + \int_{\Gamma_q} \mathbf{W}_\phi^{\mathrm{T}} (\hat{q}_n - \bar{q}_n) \mathrm{d}\Gamma = \mathbf{0} \tag{9.12}$$

The reason we have premultiplied Eq. (9.2) by $\mathbf{k}^{-1}$ is now evident as the choice

$$\mathbf{W}_q = \mathbf{N}_q, \qquad \mathbf{W}_\phi = \mathbf{N}_\phi \tag{9.13}$$

will yield symmetric equations [using Green's theorem to perform integration by parts on the gradient term in Eq. (9.12)] of the form

$$\begin{bmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C}^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{q}} \\ \tilde{\boldsymbol{\phi}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{Bmatrix} \tag{9.14}$$

with

$$\begin{aligned} \mathbf{A} &= \int_\Omega \mathbf{N}_q^{\mathrm{T}} \mathbf{k}^{-1} \mathbf{N}_q \, \mathrm{d}\Omega, & \mathbf{C} &= \int_\Omega \mathbf{N}_q^{\mathrm{T}} \nabla \mathbf{N}_\phi \, \mathrm{d}\Omega \\ \mathbf{f}_1 &= \mathbf{0}, & \mathbf{f}_2 &= \int_\Omega \mathbf{N}_\phi^{\mathrm{T}} Q \, \mathrm{d}\Omega + \int_{\Gamma_q} \mathbf{N}_\phi^{\mathrm{T}} \bar{q}_n \, \mathrm{d}\Gamma \end{aligned} \tag{9.15}$$

This problem, which we shall consider as typifying a large number of mixed approximations, illustrates the main features of the mixed formulation, including its advantages and disadvantages. We note that:

1. The continuity requirements on the shape functions chosen are different. It is easily seen that those given for $\mathbf{N}_\phi$ can be $C_0$ continuous while those for $\mathbf{N}_q$ can be discontinuous in or between elements ($C_{-1}$ continuity) as no derivatives of this are present. Alternatively, this discontinuity can be transferred to $\mathbf{N}_\phi$ (using Green's theorem on the integral in $\mathbf{C}$) while maintaining $C_0$ continuity for $\mathbf{N}_q$.
   This relaxation of continuity is of particular importance in plate and shell bending problems (see Chapters 13 and 14) and indeed many important early uses of mixed forms have been made in that context [6–9].
2. If interest is focused on the variable $\mathbf{q}$ rather than $\phi$, use of an improved approximation for this may result in higher accuracy than possible with the irreducible form previously discussed. *However, we must note that if the approximation function for* $\mathbf{q}$ *is capable of reproducing precisely the same type of variation as that determinable from the irreducible form then no additional accuracy will result and, indeed, the two approximations will yield identical answers.*
   Thus, for instance, if we consider the mixed approximation to the field problems discussed using a linear triangle to determine $\mathbf{N}_\phi$ and piecewise constant $\mathbf{N}_q$, as shown in Fig. 9.1, we will obtain precisely the same results as those obtained by the irreducible formulation with the same $\mathbf{N}_\phi$ applied directly to Eq. (9.5), *provided* $\mathbf{k}$ *is constant within each element*. This is evident since using the above approximations the second of Eq. (9.14) is precisely the weighted continuity statement used in deriving the irreducible formulation in which the first of the equations also is identically satisfied.
   Indeed, should we choose to use a linear but discontinuous approximation form of $\mathbf{N}_q$ in the interior of such a triangle, we would still obtain precisely the same answers, with the additional coefficients becoming zero. This discovery was made by Fraeijs de Veubeke [10] and is called the *limitation principle*, showing that under some circumstances no additional accuracy is to be expected from a mixed formulation. In a more general case where $\mathbf{k}$ is, for instance, discontinuous and variable within an element, the results of the mixed approximation will be different and on occasion superior [2]. Note that a $C_0$-continuous approximation for $\mathbf{q}$ does not fall into this category as it is not capable of reproducing the discontinuous ones.
3. The equations resulting from mixed formulations frequently have zero diagonal terms as indeed in the case of Eq. (9.14).
   We noted in Chapter 4 that this is a characteristic of problems constrained by a Lagrange multiplier variable. Indeed, this is the origin of the problem, which adds some difficulty to a standard Gaussian elimination process used in equation solving. As the form of Eq. (9.14) is typical of many two-field problems we shall refer to the first variable (here $\tilde{\mathbf{q}}$) as the *primary variable* and the second (here $\tilde{\boldsymbol{\phi}}$) as the *constraint variable*.

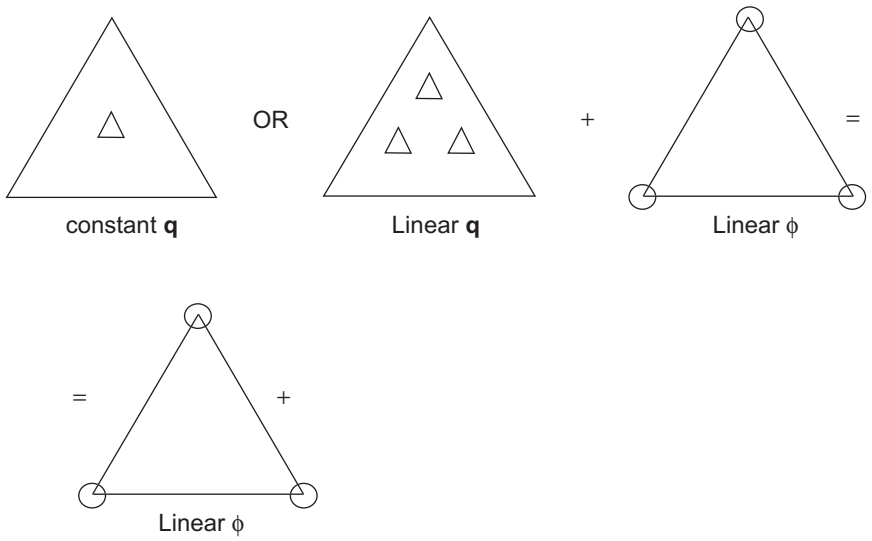



**FIGURE 9.1**
A mixed approximation to the heat conduction problem yielding identical results as the corresponding irreducible form (the constant **k** is assumed in each element).

4. The added number of variables means that generally larger size algebraic problems have to be dealt with.

The characteristics so far discussed did not mention one vital point which we elaborate in the next section.

## 9.3 Stability of mixed approximation: The patch test

### 9.3.1 Solvability requirement

Despite the relaxation of shape function continuity requirements in the mixed approximation, for certain choices of the individual shape functions the mixed approximation will not yield meaningful results. This limitation is indeed much more severe than in an *irreducible* formulation where a very simple "constant gradient" (or constant strain) condition sufficed to ensure a convergent form once continuity requirements were satisfied.

The mathematical reasons for this difficulty are discussed by Babuška [11,12] and Brezzi [13], who formulated a mathematical criterion associated with their names. However, some sources of the difficulties (and hence ways of avoiding them) follow from quite simple reasoning.

If we consider the equation system (9.14) to be typical of many mixed systems in which $\tilde{\mathbf{q}}$ is the *primary variable* and $\tilde{\boldsymbol{\phi}}$ is the *constraint variable* (equivalent to a Lagrangian multiplier), we note that the solution can proceed by eliminating $\tilde{\mathbf{q}}$ from the first equation and by substituting into the second to obtain

$$(\mathbf{C}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{C})\tilde{\boldsymbol{\phi}} = -\mathbf{f}_2 + \mathbf{C}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{f}_1 \tag{9.16}$$

which requires the matrix $\mathbf{A}$ to be nonsingular (or $\mathbf{A}\tilde{\mathbf{q}} \neq \mathbf{0}$ for all $\tilde{\mathbf{q}} \neq \mathbf{0}$). To calculate $\tilde{\boldsymbol{\phi}}$ it is necessary to ensure that the bracketed matrix, i.e.,

$$\mathbf{H} = \mathbf{C}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{C} \tag{9.17}$$

is nonsingular.

Singularity of the $\mathbf{H}$ matrix will always occur if the number of unknowns in the vector $\tilde{\mathbf{q}}$, which we call $n_q$, is less than the number of unknowns $n_\phi$ in the vector $\tilde{\boldsymbol{\phi}}$. Thus for avoidance of singularity

$$n_q \geq n_\phi \tag{9.18}$$

is *necessary* though not *sufficient* as we shall find later.

The reason for this is evident as the rank of the matrix (9.17), which needs to be $n_\phi$, cannot be greater than $n_q$, i.e., the rank of $\mathbf{A}^{-1}$.

In some problems the matrix $\mathbf{A}$ may well be singular. It can normally be made nonsingular by addition of a multiple of the second equation, thus changing the first equation to

$$\bar{\mathbf{A}} = \mathbf{A} + \gamma\mathbf{C}\mathbf{C}^{\mathrm{T}}$$
$$\bar{\mathbf{f}}_1 = \mathbf{f}_1 + \gamma\mathbf{C}\mathbf{f}_2$$

where $\gamma$ is an arbitrary number. We note that the solution to (9.14) is not changed by this modification.

Although both the matrices $\mathbf{A}$ and $\mathbf{C}\mathbf{C}^{\mathrm{T}}$ are singular their combination $\bar{\mathbf{A}}$ should not be, provided we ensure that for all vectors $\tilde{\mathbf{q}} \neq \mathbf{0}$ either

$$\mathbf{A}\tilde{\mathbf{q}} \neq \mathbf{0} \quad \text{or} \quad \mathbf{C}^{\mathrm{T}}\tilde{\mathbf{q}} \neq \mathbf{0}$$

In mathematical terminology this means that $\mathbf{A}$ is nonsingular in the null space of $\mathbf{C}\mathbf{C}^{\mathrm{T}}$.

The requirement of Eq. (9.18) is a necessary but not sufficient condition for nonsingularity of the matrix $\mathbf{H}$. An additional requirement evident from Eq. (9.16) is

$$\mathbf{C}\tilde{\boldsymbol{\phi}} \neq \mathbf{0} \quad \text{for all } \tilde{\boldsymbol{\phi}} \neq \mathbf{0}$$

If this is not the case the solution would not be unique.

The above requirements are inherent in the Babuška-Brezzi (BB) condition previously mentioned, but can always be verified algebraically.

### 9.3.2 Locking

Condition (9.18) ensures that nonzero answers for the variables $\tilde{\mathbf{q}}$ are possible. If it is violated *locking* or nonconvergent results will occur in the formulation, giving near-zero answers for $\tilde{\mathbf{q}}$.

To show this, we shall replace Eq. (9.14) by its penalized form:

$$\begin{bmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C}^{\mathrm{T}} & -\frac{1}{\alpha}\mathbf{I} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{q}} \\ \tilde{\boldsymbol{\phi}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{Bmatrix} \begin{array}{l} \text{with } \alpha \to \infty \\ \text{and } \mathbf{I} = \text{identity matrix} \end{array} \tag{9.19}$$

Elimination of $\tilde{\boldsymbol{\phi}}$ leads to

$$(\mathbf{A} + \alpha \mathbf{C}\mathbf{C}^{\mathrm{T}})\tilde{\mathbf{q}} = \mathbf{f}_1 + \alpha \mathbf{C}\mathbf{f}_2 \tag{9.20}$$

As $\alpha \to \infty$ the above becomes simply

$$(\mathbf{C}\mathbf{C}^{\mathrm{T}})\tilde{\mathbf{q}} = \mathbf{C}\mathbf{f}_2 \tag{9.21}$$

Nonzero answers for $\tilde{\mathbf{q}}$ should exist even when $\mathbf{f}_2$ is zero and hence the matrix $\mathbf{C}\mathbf{C}^{\mathrm{T}}$ *must be singular*. This singularity will always exist if $n_q > n_\phi$, but can exist also when $n_q = n_\phi$ if the rank of $\mathbf{C}$ is less than $n_q$.

The stability conditions derived on the particular example of Eq. (9.14) are generally valid for any problem exhibiting the standard Lagrange multiplier form. In particular, the necessary count condition will in many cases suffice to determine element acceptability; however, final conclusions for successful elements which pass all count conditions and the full test to ensure consistency must be evaluated by rank tests on the full matrix.

In the example just quoted $\tilde{\mathbf{q}}$ denotes flux and $\tilde{\boldsymbol{\phi}}$ temperature and perhaps the concept of locking was not clearly demonstrated. It is much more definite where the first primary variable is a displacement and the second constraining one is a stress or a pressure. There locking is more evident physically and simply means an occurrence of zero displacements throughout as the solution approaches a numerical instability limit. This unfortunately will happen on occasion.

### 9.3.3 The mixed patch test

The patch test for mixed elements can be carried out in exactly the way we have described in the previous chapter for irreducible elements. As *consistency* is easily assured by taking a polynomial approximation for each of the variables, only *stability* needs generally to be investigated. *Most answers* to this can be obtained by simply ensuring that *count condition* (9.18) is satisfied for any isolated patch on the boundaries of which we constrain the *maximum* number of primary variables and the *minimum* number of constraint variables [14].

**Example 9.1.** A single-element test

In Fig. 9.2 we illustrate a single-element test for two possible formulations with $C_0$ continuous $N_\phi$ (quadratic) and discontinuous $N_q$, assumed to be either constant or linear within an element of triangular form. As no values of $\tilde{\mathbf{q}}$ can here be specified on

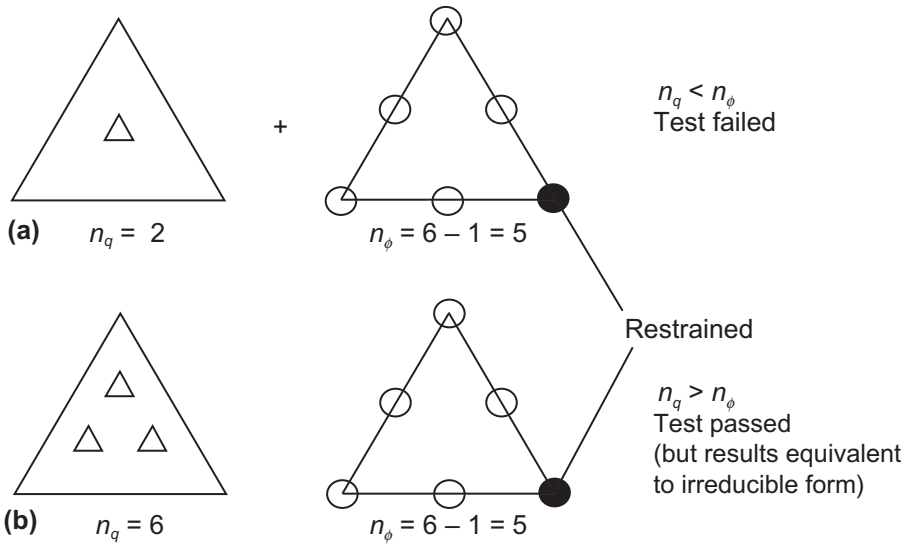



**FIGURE 9.2**

Single-element patch test for mixed approximations to the heat conduction problem with discontinuous flux $\mathbf{q}$ assumed: (a) quadratic $C_0$, $\phi$; constant $\mathbf{q}$; (b) quadratic $C_0$, $\phi$; linear $\mathbf{q}$.

the boundaries, on the patch (which is here simply that of a single element) we shall fix a single value of $\tilde{\boldsymbol{\phi}}$ only, as is necessary to ensure uniqueness. A count shows that only one of the formulations, i.e., that with linear flux variation, satisfies condition (9.18) and therefore may be acceptable (but will always determine elements which fail!).

**Example 9.2.** A single-element test with $C_0$ $\tilde{\mathbf{q}}$ and $\phi$

In Fig. 9.3 we illustrate a similar patch test on the same element but with identical $C_0$ continuous shape functions specified for both $\tilde{\mathbf{q}}$ and $\tilde{\boldsymbol{\phi}}$ variables. This example shows satisfaction of the basic condition of Eq. (9.18) and therefore is apparently a permissible formulation. The permissible formulation must always be subjected to a numerical rank test.

Clearly condition (9.18) will need to be satisfied and many useful conclusions can be drawn from such counts. These eliminate elements which will not function and on many occasions will give guidance to elements which will.

Even if the patch test is satisfied occasional difficulties can arise, and these are indicated mathematically by the Babuška-Brezzi (BB) condition already referred to [15]. These difficulties can be due to *excessive continuity* imposed on the problem by requiring, for instance, the flux condition to be of $C_0$ continuity class. In Fig. 9.4 we illustrate some cases in which the imposition of such continuity is *physically incorrect* and therefore can be expected to produce erroneous (and usually highly

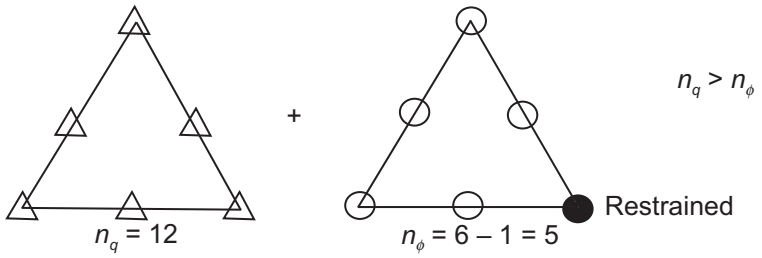



**FIGURE 9.3**
As Fig. 9.2 but with $C_0$ continuous $\mathbf{q}$.

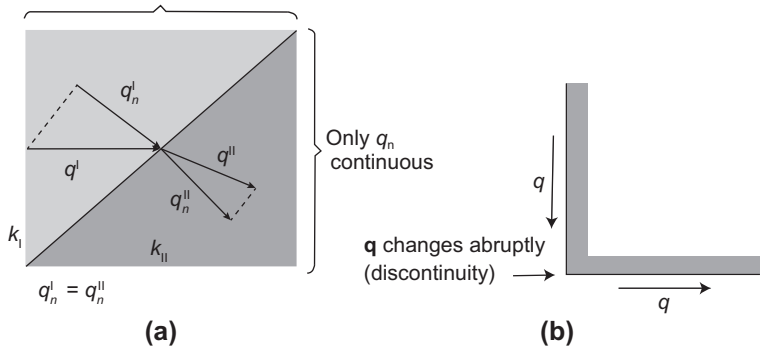



**FIGURE 9.4**
Some situations for which $C_0$ continuity of flux $\mathbf{q}$ is inappropriate: (a) discontinuous change of material properties and (b) singularity.

oscillating) results. In all such problems we recommend that *the continuity be relaxed on all surfaces where a physical discontinuity can occur*.

We shall discuss this problem further in Section 9.4.3.

## 9.4 Two-field mixed formulation in elasticity

### 9.4.1 General

In all the previous formulations of elasticity problems in this book we have used an irreducible formulation, using the displacement $\mathbf{u}$ as the primary variable. In earlier chapters, the virtual work principle was used to establish the equilibrium conditions and was written for static problems as

$$\int_\Omega \delta\boldsymbol{\epsilon}^{\mathrm{T}}\boldsymbol{\sigma}\,\mathrm{d}\Omega - \int_\Omega \delta\mathbf{u}^{\mathrm{T}}\mathbf{b}\,\mathrm{d}\Omega - \int_{\Gamma_t} \delta\mathbf{u}^{\mathrm{T}}\bar{\mathbf{t}}\,\mathrm{d}\Gamma = 0 \tag{9.22}$$

where $\bar{\mathbf{t}}$ are the tractions prescribed on $\Gamma_t$ and with

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\epsilon} \tag{9.23}$$

as the constitutive relation (omitting here initial strains and stresses for simplicity).

We recall that statements such as Eq. (9.22) are equivalent to weighted residual forms (see Chapter 3) and in what follows we shall use these frequently. In the above the strains are related to displacement by the matrix operator $\boldsymbol{\mathcal{S}}$ introduced in Chapter 2, giving

$$\begin{aligned} \boldsymbol{\epsilon} &= \boldsymbol{\mathcal{S}}\mathbf{u} \\ \delta\boldsymbol{\epsilon} &= \boldsymbol{\mathcal{S}}\,\delta\mathbf{u} \end{aligned} \tag{9.24}$$

with the displacement expansions constrained to satisfy the prescribed displacements on $\Gamma_u$.

With the displacement $\mathbf{u}$ approximated as

$$\mathbf{u} \approx \hat{\mathbf{u}} = \mathbf{N}_u\tilde{\mathbf{u}} \tag{9.25}$$

the required stiffness equations were obtained in terms of the unknown displacement vector $\tilde{\mathbf{u}}$ and the solution obtained.

It is possible to use mixed forms in which either $\boldsymbol{\sigma}$ or $\boldsymbol{\epsilon}$, or, indeed, both these variables, are approximated independently. We shall discuss such formulations below.

### 9.4.2 The u-$\sigma$ mixed form

In this we shall assume that Eq. (9.22) is valid but that we approximate $\boldsymbol{\sigma}$ independently as

$$\boldsymbol{\sigma} \approx \hat{\boldsymbol{\sigma}} = \mathbf{N}_\sigma\tilde{\boldsymbol{\sigma}} \tag{9.26}$$

and approximately satisfy the constitutive relation

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\mathcal{S}}\mathbf{u} \tag{9.27}$$

which replaces (9.23) and (9.24). The approximate integral form is written as

$$\int_\Omega \delta\boldsymbol{\sigma}^{\mathrm{T}}(\boldsymbol{\mathcal{S}}\mathbf{u} - \mathbf{D}^{-1}\boldsymbol{\sigma})\mathrm{d}\Omega = 0 \tag{9.28}$$

where the expression in the brackets is simply Eq. (9.27) premultiplied by $\mathbf{D}^{-1}$ to establish symmetry and $\delta\boldsymbol{\sigma}$ is introduced as a weighting variable.

Indeed, Eqs. (9.22) and (9.28) which now define the problem are equivalent to the stationarity of the functional

$$\Pi_{\mathrm{HR}} = \int_\Omega \boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\mathcal{S}}\mathbf{u}\,\mathrm{d}\Omega - \frac{1}{2}\int_\Omega \boldsymbol{\sigma}^{\mathrm{T}}\mathbf{D}^{-1}\boldsymbol{\sigma}\,\mathrm{d}\Omega - \int_\Omega \mathbf{u}^{\mathrm{T}}\mathbf{b}\,\mathrm{d}\Omega - \int_{\Gamma_t} \mathbf{u}^{\mathrm{T}}\bar{\mathbf{t}}\,\mathrm{d}\Gamma \tag{9.29}$$

where the boundary displacement

$$\mathbf{u} = \bar{\mathbf{u}}$$

is enforced on $\Gamma_u$, as the reader can readily verify. This is the well-known Hellinger-Reissner [16,17] variational principle, but, as we have remarked earlier, it is unnecessary in deriving approximate equations. Using

$$\begin{aligned} \mathbf{N}_u \delta\tilde{\mathbf{u}} &\quad \text{in place of } \delta\mathbf{u} \\ \mathbf{B}\delta\tilde{\mathbf{u}} \equiv \mathcal{S}\mathbf{N}_u\delta\tilde{\mathbf{u}} &\quad \text{in place of } \delta\boldsymbol{\epsilon} \\ \mathbf{N}_\sigma \delta\tilde{\boldsymbol{\sigma}} &\quad \text{in place of } \delta\boldsymbol{\sigma} \end{aligned}$$

we write the approximate Eqs. (9.28) and (9.22) in the standard form [see Eq. (9.14)]

$$\begin{bmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C}^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \tilde{\boldsymbol{\sigma}} \\ \tilde{\mathbf{u}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{Bmatrix} \tag{9.30}$$

with

$$\begin{aligned} &\mathbf{A} = -\int_\Omega \mathbf{N}_\sigma^{\mathrm{T}} \mathbf{D}^{-1} \mathbf{N}_\sigma \, \mathrm{d}\Omega, \quad \mathbf{C} = \int_\Omega \mathbf{N}_\sigma^{\mathrm{T}} \mathbf{B} \, \mathrm{d}\Omega \\ &\mathbf{f}_1 = \mathbf{0}, \qquad\qquad \mathbf{f}_2 = \int_\Omega \mathbf{N}_u^{\mathrm{T}} \mathbf{b} \, \mathrm{d}\Omega + \int_{\Gamma_t} \mathbf{N}_u^{\mathrm{T}} \bar{\mathbf{t}} \, \mathrm{d}\Gamma \end{aligned} \tag{9.31}$$

In the form given above the $\mathbf{N}_u$ shape functions still have to be of $C_0$ continuity, though $\mathbf{N}_\sigma$ can be discontinuous. However, integration by parts of the expression for $\mathbf{C}$ allows a reduction of such continuity and indeed this form has been used by Herrmann [6,18,19] for problems of plates and shells.

### 9.4.3 Stability of two-field approximation in elasticity (u-$\sigma$)

Before attempting to formulate practical mixed approach approximations in detail, identical stability problems to those discussed in Section 9.3 have to be considered.

For the $\mathbf{u}$-$\boldsymbol{\sigma}$ forms it is clear that $\boldsymbol{\sigma}$ is the *primary variable* and $\mathbf{u}$ the *constraint variable* (see Section 9.2), and for the total problem as well as for element patches we must have as a necessary, though not sufficient condition

$$n_\sigma \geq n_u \tag{9.32}$$

where $n_\sigma$ and $n_u$ stand for numbers of degrees of freedom in appropriate variables.

In Fig. 9.5 we consider a two-dimensional plane problem and show a series of elements in which $\mathbf{N}_\sigma$ is discontinuous while $\mathbf{N}_u$ has $C_0$ continuity. We note again, by invoking the Veubeke "limitation principle," that all the elements that pass the single-element test here will in fact yield identical results to those obtained by using the equivalent irreducible form, provided the $\mathbf{D}$ matrix and the determinant of the Jacobian matrix are constant within each element. They are therefore of little interest. However, we note in passing that the Q 4/8, which fails in a single-element test, passes that patch test for assemblies of two or more elements, and performs well in many circumstances. We shall see later that this is equivalent to using four-point Gauss, *reduced* integration and as we have mentioned in Chapters 7 and 8 such elements will not always be robust.

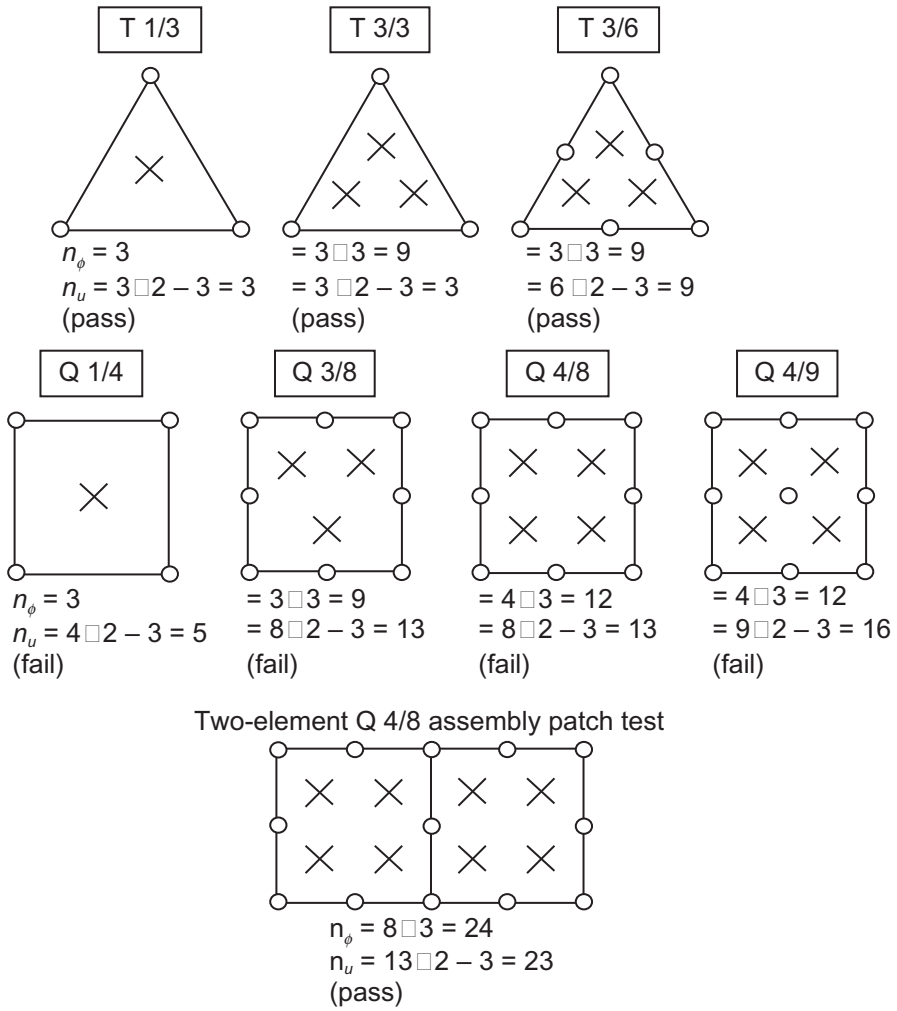



**FIGURE 9.5**

Elasticity by the mixed $\boldsymbol{\sigma}$-$\mathbf{u}$ formulation. Discontinuous stress approximation. Single-element patch test. No restraint on $\tilde{\boldsymbol{\sigma}}$ variables but three $\mathbf{u}$ degrees-of-freedom restrained on patch. Test condition $n_\sigma \geq n_u$ [$\times$ denotes $\tilde{\boldsymbol{\sigma}}$ (3 DOF) and $\circ$ the $\mathbf{u}$ (2 DOF) variables].

It is of interest to note that if a higher order of interpolation is used for $\boldsymbol{\sigma}$ than for $\mathbf{u}$ the patch test is still satisfied, but in general the results will not be improved because of the limitation principle.

We do not show the similar patch test for the $C_0$ continuous $\mathbf{N}_\sigma$ assumption but state simply that, similarly to the example of Fig. 9.3, identical interpolation of $\mathbf{N}_\sigma$ and $\mathbf{N}_u$ is acceptable from the point of view of stability. However, as in Fig. 9.4, restriction of *excessive continuity* for stresses has to be avoided at singularities and

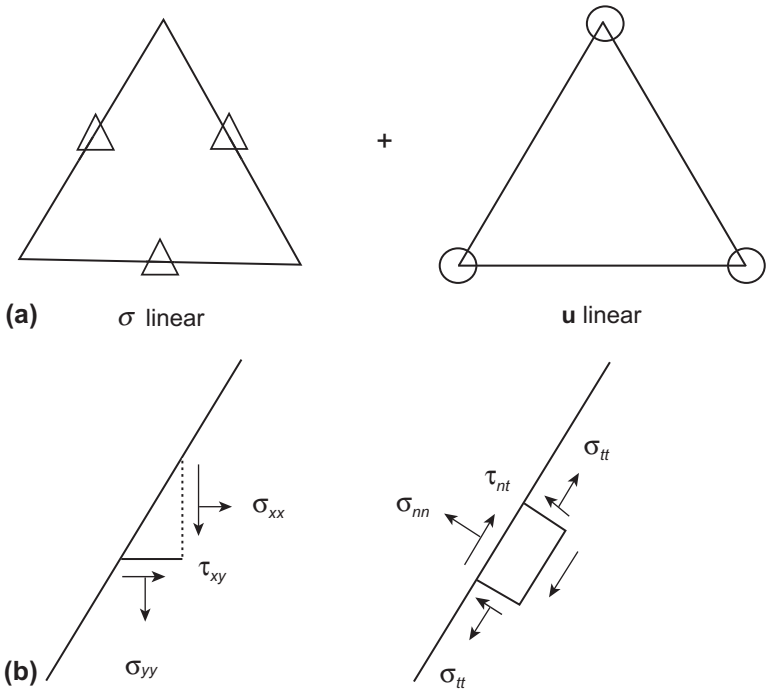



**FIGURE 9.6**

Elasticity by the mixed $\boldsymbol{\sigma}$-$\mathbf{u}$ formulation. Partially continuous $\boldsymbol{\sigma}$ (continuity at nodes only): (a) $\boldsymbol{\sigma}$ linear, $\mathbf{u}$ linear and (b) possible transformation of interface stresses with $\sigma_{tt}$ disconnected.

at abrupt material property change interfaces, where only the normal and tangential tractions are continuous.

The disconnection of stress variables at corner nodes can only be accomplished for all the stress variables. For this reason an alternative set of elements with continuous stress nodes at element interfaces can be introduced (see Fig. 9.6) [20].

In such elements excessive continuity can easily be avoided by disconnecting only the direct stress components parallel to an interface at which material changes occur. It should be noted that even in the case when all stress components are connected at a mid-side node such elements do not ensure stress continuity along the whole interface. Indeed, the amount of such discontinuity can be useful as an error measure. However, we observe that for the linear element (Fig. 9.6a) the interelement stresses are continuous *in the mean*.

It is, of course, possible to derive elements that exhibit complete continuity of the appropriate components along interfaces and indeed this was achieved by Raviart and Thomas [21] in the case of the heat conduction problem discussed previously. Extension to the full stress problem is difficult [22] and as yet such elements have not been successfully noted.

**Example 9.3.** Pian-Sumihara rectangle

Today very few two-field elements based on interpolation of the full stress and displacement fields are used. One, however, deserves to be mentioned. We begin by first considering a rectangular element where interpolations may be given directly in terms of Cartesian coordinates. A four-node plane rectangular element with side lengths $2a$ in the $x$-direction and $2b$ in the $y$-direction, shown in Fig. 9.7, has displacement interpolation given by

$$\mathbf{u} = \sum_{a=1}^{4} N_a(x, y)\tilde{\mathbf{u}}_a$$

The shape functions are given by

$$N_1(x, y) = \frac{1}{4}\left(1 - \frac{x'}{a}\right)\left(1 - \frac{y'}{b}\right); \quad N_2(x, y) = \frac{1}{4}\left(1 + \frac{x'}{a}\right)\left(1 - \frac{y'}{b}\right)$$
$$N_3(x, y) = \frac{1}{4}\left(1 + \frac{x'}{a}\right)\left(1 + \frac{y'}{b}\right); \quad N_4(x, y) = \frac{1}{4}\left(1 - \frac{x'}{a}\right)\left(1 + \frac{y'}{b}\right)$$

in which $x' = x - x_0$, $y' = y - y_0$, and $x_0$ and $y_0$ are the Cartesian coordinates at the element center. The strains generated from this interpolation will be such that

$$\epsilon_x = \beta_1 + \beta_2 y', \quad \epsilon_y = \beta_3 + \beta_4 x', \quad \gamma_{xy} = \beta_5 + \beta_6 x' + \beta_7 y'$$

where $\beta_j$ are expressed in terms of $\tilde{\mathbf{u}}$. For isotropic linear elasticity problems these strains will lead to stresses which have a complete linear polynomial variation in each element (except for the special case when $\nu = 0$).

Here the stress interpolation is restricted to each element individually and, thus, can be discontinuous between adjacent elements. The limitation principle restricts

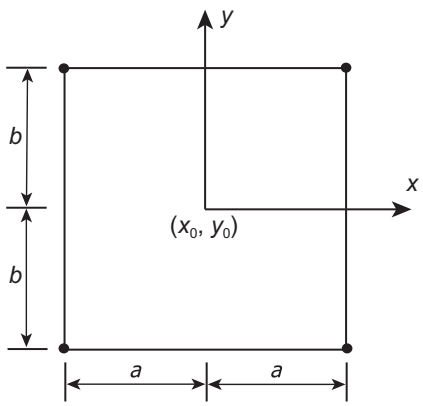



**FIGURE 9.7**
Geometry of rectangular $\boldsymbol{\sigma}$-$\mathbf{u}$ element.

the possible choices which lead to different results from the standard displacement solution. Namely, the approximation must be less than a complete linear polynomial. To satisfy the stability condition given by Eq. (9.32) we need at least five stress parameters in each element. A viable choice for a five-term approximation is one which has the same variation in each element as the normal strains given above but only a constant shear stress. Accordingly,

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 & y' & 0 \\ 0 & 1 & 0 & 0 & x' \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \end{Bmatrix}$$

Indeed, this approximation satisfies Eq. (9.32) and leads to excellent results for a rectangular element.

**Example 9.4.** Pian-Sumihara quadrilateral

We now rewrite the formulation given in Example 9.3 to permit a general quadrilateral shape to be used. The element coordinate and displacement field are given by a standard bilinear isoparametric expansion

$$\mathbf{x} = \sum_{a=1}^{4} N_a(\xi, \eta)\tilde{\mathbf{x}}_a \quad \text{and} \quad \hat{\mathbf{u}} = \sum_{a=1}^{4} N_a(\xi, \eta)\, \tilde{\mathbf{u}}_a$$

where now

$$N_a(\xi, \eta) = \frac{1}{4}(1 + \xi_a \xi)(1 + \eta_a \eta)$$

in which $\xi_a$ and $\eta_a$ are the values of the parent coordinates at node $a$.

The problem remains to deduce an approximation for stresses for the general quadrilateral element. Here this is accomplished by first assuming stresses on the parent element (for convenience in performing the coordinate transformation the tensor form is used, see Appendix B) in an analogous manner as the rectangle above:

$$\boldsymbol{\Sigma}(\xi, \eta) = \begin{bmatrix} \Sigma_{\xi\xi} & \Sigma_{\xi\eta} \\ \Sigma_{\eta\xi} & \Sigma_{\eta\eta} \end{bmatrix} = \begin{bmatrix} \alpha_1 + \alpha_4 \eta & \alpha_3 \\ \alpha_3 & \alpha_2 + \alpha_5 \xi \end{bmatrix}$$

In the above the parent normal stresses again produce constant and bending terms while shear stress is only constant. These stresses are then transformed to Cartesian space using

$$\boldsymbol{\sigma} = \mathbf{T}^{\mathrm{T}} \boldsymbol{\Sigma}(\xi, \eta) \mathbf{T}$$

It remains now only to select an appropriate form for $\mathbf{T}$. The transformation must:

1. Produce stresses in Cartesian space which satisfy the patch test (i.e., can produce constant stresses and be stable)
2. Be independent of the orientation of the initially chosen element coordinate system and numbering of element nodes (invariance requirement)

Pian and Sumihara [23] use a constant array (to preserve constant stresses) deduced from the Jacobian matrix at the center of the element. Accordingly, with

$$\mathbf{J}_0 = \begin{bmatrix} J_{0,11} & J_{0,12} \\ J_{0,21} & J_{0,22} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix}_{\xi,\eta=0}$$

the elements of the Jacobian matrix at the center are given by

$$J_{0,11} = \frac{1}{4} x_a \, \xi_a, \quad J_{0,12} = \frac{1}{4} x_a \, \eta_a$$
$$J_{0,21} = \frac{1}{4} y_a \, \xi_a, \quad J_{0,22} = \frac{1}{4} y_a \, \eta_a$$

Using $\mathbf{T} = \mathbf{J}_0$ gives the stresses (in matrix Voigt form)

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = \begin{Bmatrix} \bar{\alpha}_1 \\ \bar{\alpha}_2 \\ \bar{\alpha}_3 \end{Bmatrix} + \begin{bmatrix} J_{0,11}^2 \eta & J_{0,12}^2 \xi \\ J_{0,21}^2 \eta & J_{0,22}^2 \xi \\ J_{0,12} J_{0,21} \eta & J_{0,12} J_{0,22} \xi \end{bmatrix} \begin{Bmatrix} \alpha_4 \\ \alpha_5 \end{Bmatrix}$$

where the parameters $\bar{\alpha}_i$, $i = 1, 2, 3$, replace the transformed quantities for the constant part of the stresses. This approximation satisfies the constant stress condition (Condition 1) and can also be shown to satisfy the invariance condition (Condition 2). The development is now complete and the arrays indicated in Eq. (9.31) may be computed. We note that the integrals are computed exactly for all quadrilateral elements (with constant $\mathbf{D}$) using $2 \times 2$ Gaussian quadrature.

An alternative to the above definition for $\mathbf{T}$ is to use the transpose of the Jacobian inverse at the center of the element (i.e., $\mathbf{T} = \mathbf{J}_0^{-\mathrm{T}}$). This has also been suggested recently by several authors as an invariant transformation. However, as shown in Fig. 9.8, the sensitivity to element distortion is much greater for this form than the original one given by Pian and Sumihara for the above two-field approximation. The other two options (e.g., $\mathbf{T} = \mathbf{J}_0^{\mathrm{T}}$ and $\mathbf{T} = \mathbf{J}_0^{-1}$) do not satisfy the frame invariance requirement, thus giving elements which depend on the orientation of the element with respect to the global coordinates.

## 9.5 Three-field mixed formulations in elasticity

### 9.5.1 The u-$\sigma$-$\epsilon$ mixed form

It is, of course, possible to use an independent approximation to all the essential variables entering the elasticity problem. We can then write the three Eqs. (9.22)–(9.24)

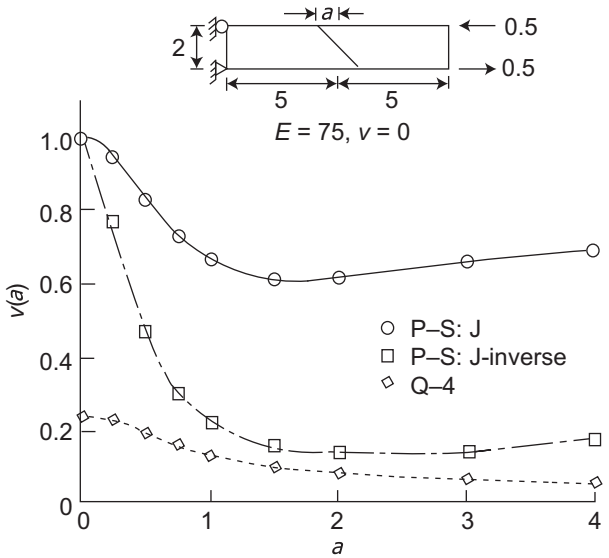



**FIGURE 9.8**

Pian-Sumihara quadrilateral (P-S) compared with displacement quadrilateral (Q-4). Effect of element distortion (Exact = 1.0).

in their weak form as

$$\begin{aligned} \int_\Omega \delta\boldsymbol{\epsilon}^{\mathrm{T}}(\mathbf{D}\boldsymbol{\epsilon} - \boldsymbol{\sigma})\mathrm{d}\Omega &= 0 \\ \int_\Omega \delta\boldsymbol{\sigma}^{\mathrm{T}}(\boldsymbol{\mathcal{S}}\mathbf{u} - \boldsymbol{\epsilon})\mathrm{d}\Omega &= 0 \\ \int_\Omega \delta(\boldsymbol{\mathcal{S}}\mathbf{u})^{\mathrm{T}}\boldsymbol{\sigma}\,\mathrm{d}\Omega - \int_\Omega \delta\mathbf{u}^{\mathrm{T}}\mathbf{b}\,\mathrm{d}\Omega - \int_{\Gamma_t} \delta\mathbf{u}^{\mathrm{T}}\bar{\mathbf{t}}\,\mathrm{d}\Gamma &= 0 \end{aligned} \tag{9.33}$$

where $\mathbf{u} \equiv \bar{\mathbf{u}}$ on $\Gamma_u$ is enforced.[1] The variational principle equivalent to Eq. (9.33) is known by the name of Hu-Washizu [5] (see Problem 9.1).

Introducing the approximations

$$\mathbf{u} \approx \hat{\mathbf{u}} = \mathbf{N}_u\tilde{\mathbf{u}}, \quad \boldsymbol{\sigma} \approx \hat{\boldsymbol{\sigma}} = \mathbf{N}_\sigma\tilde{\boldsymbol{\sigma}} \quad \text{and} \quad \boldsymbol{\epsilon} \approx \hat{\boldsymbol{\epsilon}} = \mathbf{N}_\epsilon\tilde{\boldsymbol{\epsilon}} \tag{9.34}$$

with corresponding "variations" (i.e., the Galerkin form $\mathbf{W}_u = \mathbf{N}_u$, etc.) into Eq. (9.33), and writing the approximating equations in a similar fashion as we have in the

[1] It is possible to include the displacement boundary conditions in Eq. (9.33) as a natural rather than imposed constraint; however, most finite element applications of the principle are in the form shown.

previous section yields an equation system of the following form:

$$\begin{bmatrix} \mathbf{A} & \mathbf{C} & \mathbf{0} \\ \mathbf{C}^{\mathrm{T}} & \mathbf{0} & \mathbf{E} \\ \mathbf{0} & \mathbf{E}^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \tilde{\boldsymbol{\epsilon}} \\ \tilde{\boldsymbol{\sigma}} \\ \tilde{\mathbf{u}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \end{Bmatrix} \tag{9.35}$$

where

$$\begin{aligned} &\mathbf{A} = \int_\Omega \mathbf{N}_\epsilon^{\mathrm{T}} \mathbf{D} \mathbf{N}_\epsilon \, \mathrm{d}\Omega, \quad \mathbf{E} = \int_\Omega \mathbf{N}_\sigma^{\mathrm{T}} \mathbf{B} \, \mathrm{d}\Omega, \quad \mathbf{C} = -\int_\Omega \mathbf{N}_\epsilon^{\mathrm{T}} \mathbf{N}_\sigma \, \mathrm{d}\Omega \\ &\mathbf{f}_1 = \mathbf{f}_2 = 0, \quad \mathbf{f}_3 = \int_\Omega \mathbf{N}_u^{\mathrm{T}} \mathbf{b} \, \mathrm{d}\Omega + \int_{\Gamma_t} \mathbf{N}_u^{\mathrm{T}} \bar{\mathbf{t}} \, \mathrm{d}\Gamma \end{aligned} \tag{9.36}$$

The reader will observe that in this section we have developed all the approximations directly without using a variational principle. In Problem 9.2 we suggest that the reader show the equivalence of a development from the variational principle which one made directly using a weak form.

### 9.5.2 Stability condition of three-field approximation (u-$\sigma$-$\epsilon$)

The stability condition derived in Section 9.3 [Eq. (9.18)] for two-field problems, which we later used in Eq. (9.32) for the simple mixed elasticity form, needs to be modified when three-field approximations of the form given in Eq. (9.35) are considered.

Many other problems fall into a similar category (for instance, plate bending) and hence the conditions of stability are generally useful. The requirement now is that

$$\begin{aligned} n_\epsilon + n_u &\geq n_\sigma \\ n_\sigma &\geq n_u \end{aligned} \tag{9.37}$$

This was first stated in Ref. [24] and follows directly from the two-field criterion as shown below.

The system of Eq. (9.35) can be "regularized" by adding $\gamma \mathbf{E}$ times the third equation to the second, with $\gamma$ being an arbitrary constant. We now have

$$\begin{bmatrix} \mathbf{A} & \mathbf{C} & \mathbf{0} \\ \mathbf{C}^{\mathrm{T}} & \gamma \mathbf{E}\mathbf{E}^{\mathrm{T}} & \mathbf{E} \\ \mathbf{0} & \mathbf{E}^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \tilde{\boldsymbol{\epsilon}} \\ \tilde{\boldsymbol{\sigma}} \\ \tilde{\mathbf{u}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 + \gamma \mathbf{E} \mathbf{f}_3 \\ \mathbf{f}_3 \end{Bmatrix}$$

On elimination of $\boldsymbol{\epsilon}$ using the first of the above we have

$$\begin{bmatrix} (\gamma \mathbf{E}\mathbf{E}^{\mathrm{T}} - \mathbf{C}^{\mathrm{T}} \mathbf{A}^{-1} \mathbf{C}) & \mathbf{E} \\ \mathbf{E}^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \tilde{\boldsymbol{\sigma}} \\ \tilde{\mathbf{u}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_2 + \gamma \mathbf{E} \mathbf{f}_3 - \mathbf{C}^{\mathrm{T}} \mathbf{A}^{-1} \mathbf{f}_1 \\ \mathbf{f}_3 \end{Bmatrix}$$

From the two-field requirement [Eq. (9.32)] it follows that we require

$$n_\sigma \geq n_u \tag{9.38}$$

for the equation system to have a solution.

To establish the second condition we rearrange Eq. (9.35) as

$$\begin{bmatrix} \mathbf{A} & \mathbf{0} & \mathbf{C} \\ \mathbf{0} & \mathbf{0} & \mathbf{E}^{\mathrm{T}} \\ \mathbf{C}^{\mathrm{T}} & \mathbf{E} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \tilde{\boldsymbol{\epsilon}} \\ \tilde{\mathbf{u}} \\ \tilde{\boldsymbol{\sigma}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{f}_3 \\ \mathbf{f}_2 \end{Bmatrix}$$

This again can be regularized by adding multiples $\gamma\mathbf{C}$ and $\gamma\mathbf{E}^{\mathrm{T}}$ of the third of the above equations to the first and second, respectively, obtaining

$$\begin{bmatrix} \mathbf{A}+\gamma\mathbf{C}\mathbf{C}^{\mathrm{T}} & \gamma\mathbf{C}\mathbf{E} & \mathbf{C} \\ \gamma\mathbf{E}^{\mathrm{T}}\mathbf{C}^{\mathrm{T}} & \gamma\mathbf{E}^{\mathrm{T}}\mathbf{E} & \mathbf{E}^{\mathrm{T}} \\ \mathbf{C}^{\mathrm{T}} & \mathbf{E} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \tilde{\boldsymbol{\epsilon}} \\ \tilde{\mathbf{u}} \\ \tilde{\boldsymbol{\sigma}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1+\gamma\mathbf{C}\mathbf{f}_2 \\ \mathbf{f}_3+\gamma\mathbf{E}^{\mathrm{T}}\mathbf{f}_2 \\ \mathbf{f}_2 \end{Bmatrix}$$

By partitioning as above it is evident that we require

$$n_\epsilon + n_u \geq n_\sigma \tag{9.39}$$

We shall not discuss in detail any of the possible approximations to the $\boldsymbol{\epsilon}$-$\boldsymbol{\sigma}$-$\mathbf{u}$ formulation or their corresponding patch tests as the arguments are similar to those of two-field problems.

In some practical applications of the three-field form the approximation of the second and third equations in (9.33) is used directly to eliminate all but the displacement terms. This leads to a special form of the displacement method which has been called a $\bar{\mathbf{B}}$ ($\mathbf{B}$-bar) form [25,26]. In the $\bar{\mathbf{B}}$ form the shape function derivatives are replaced by approximations resulting from the mixed form. We shall illustrate this concept with an example of a *nearly incompressible* material in Section 10.4.

### 9.5.3 The u-$\sigma$-$\epsilon_{\mathrm{en}}$ form: Enhanced strain formulation

In the previous two sections the general form and stability conditions of the three-field formulation for elasticity problems are given in Eqs. (9.32) and (9.37). Here we consider a special case of this form from which several useful elements may be deduced.

In the special form considered the strain approximation is split into two parts: one the usual displacement-gradient term and, second, an added or *enhanced strain* part. Accordingly, we write

$$\boldsymbol{\epsilon} = \boldsymbol{\mathcal{S}}\mathbf{u} + \boldsymbol{\epsilon}_{\mathrm{en}}, \quad \delta\boldsymbol{\epsilon} = \delta(\boldsymbol{\mathcal{S}}\mathbf{u}) + \delta\boldsymbol{\epsilon}_{\mathrm{en}} \tag{9.40}$$

Substitution into Eq. (9.33) yields the weak forms as

$$\begin{aligned} \int_\Omega \delta\left(\boldsymbol{\mathcal{S}}\mathbf{u}\right)^{\mathrm{T}}\left(\mathbf{D}\left(\boldsymbol{\mathcal{S}}\mathbf{u}+\boldsymbol{\epsilon}_{\mathrm{en}}\right)-\boldsymbol{\sigma}\right)\mathrm{d}\Omega &= 0 \\ \int_\Omega \delta\boldsymbol{\epsilon}_{\mathrm{en}}^{\mathrm{T}}\left(\mathbf{D}\left(\boldsymbol{\mathcal{S}}\mathbf{u}+\boldsymbol{\epsilon}_{\mathrm{en}}\right)-\boldsymbol{\sigma}\right)\mathrm{d}\Omega &= 0 \\ \int_\Omega \delta\boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\epsilon}_{\mathrm{en}}\,\mathrm{d}\Omega &= 0 \\ \int_\Omega \delta\left(\boldsymbol{\mathcal{S}}\mathbf{u}\right)^{\mathrm{T}}\boldsymbol{\sigma}\,\mathrm{d}\Omega - \int_\Omega \delta\mathbf{u}^{\mathrm{T}}\mathbf{b}\,\mathrm{d}\Omega - \int_{\Gamma_t}\delta\mathbf{u}^{\mathrm{T}}\bar{\mathbf{t}}\,\mathrm{d}\Gamma &= 0 \end{aligned} \tag{9.41}$$

where, as before, $\mathbf{u} = \bar{\mathbf{u}}$ is enforced on $\Gamma_u$.

We can directly discretize Eq. (9.41) by taking the approximations

$$\mathbf{u} \approx \hat{\mathbf{u}} = \mathbf{N}_u \tilde{\mathbf{u}}, \quad \boldsymbol{\sigma} \approx \hat{\boldsymbol{\sigma}} = \mathbf{N}_\sigma \tilde{\boldsymbol{\sigma}}, \quad \boldsymbol{\epsilon}_{\text{en}} \approx \hat{\boldsymbol{\epsilon}}_{\text{en}} = \mathbf{N}_{\text{en}} \tilde{\boldsymbol{\epsilon}}_{\text{en}} \tag{9.42}$$

with corresponding expressions for variations. Substituting the approximations into Eq. (9.41) yields the discrete equation system

$$\begin{bmatrix} \mathbf{A} & \mathbf{C} & \mathbf{G} \\ \mathbf{C}^{\mathrm{T}} & \mathbf{0} & \mathbf{0} \\ \mathbf{G}^{\mathrm{T}} & \mathbf{0} & \mathbf{K} \end{bmatrix} \begin{Bmatrix} \tilde{\boldsymbol{\epsilon}}_{\text{en}} \\ \tilde{\boldsymbol{\sigma}} \\ \tilde{\mathbf{u}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \end{Bmatrix} \tag{9.43}$$

where

$$\begin{aligned} \mathbf{A} &= \int_\Omega \mathbf{N}_{\text{en}}^{\mathrm{T}} \mathbf{D}\, \mathbf{N}_{\text{en}}\, \mathrm{d}\Omega, \quad \mathbf{K} = \int_\Omega \mathbf{B}^{\mathrm{T}} \mathbf{D}\, \mathbf{B}\, \mathrm{d}\Omega \\ \mathbf{C} &= -\int_\Omega \mathbf{N}_{\text{en}}^{\mathrm{T}} \mathbf{N}_\sigma\, \mathrm{d}\Omega, \quad \mathbf{f}_1 = \mathbf{f}_2 = \mathbf{0} \\ \mathbf{G} &= \int_\Omega \mathbf{N}_{\text{en}}^{\mathrm{T}} \mathbf{D}\, \mathbf{B}\, \mathrm{d}\Omega, \quad \mathbf{f}_3 = \int_\Omega \mathbf{N}_u^{\mathrm{T}} \mathbf{b}\, \mathrm{d}\Omega + \int_{\Gamma_t} \mathbf{N}_u^{\mathrm{T}} \bar{\mathbf{t}}\, \mathrm{d}\Gamma \end{aligned} \tag{9.44}$$

In this form there is only one zero diagonal term and the stability condition reduces to the single condition

$$n_u + n_{\text{en}} \geq n_\sigma \tag{9.45}$$

Further, the use of the strains deduced from the displacement interpolation leads to a matrix which is identical to that from the irreducible form and we have thus included this in Eq. (9.44) as $\mathbf{K}$.

**Example 9.5.** Simo-Rifai quadrilateral

An enhanced strain formulation for application to problems in plain elasticity was introduced by Simo and Rifai [27]. The element has four nodes and employs isoparametric interpolation for the displacement field. The derivatives of the shape functions yield a form

$$\begin{Bmatrix} \frac{\partial N_a}{\partial x} \\ \frac{\partial N_a}{\partial y} \end{Bmatrix} = \frac{1}{J(\xi, \eta)} \begin{Bmatrix} a_{x,a}(y_b) + b_{x,a}(y_b)\xi + c_{x,a}(y_b)\eta \\ a_{y,a}(x_b) + b_{y,a}(x_b)\xi + c_{y,a}(x_b)\eta \end{Bmatrix}$$

where $a_a$, $b_a$, and $c_a$ depend on the nodal coordinates, and the Jacobian determinant for the four-node quadrilateral is given by[2]

$$\det \mathbf{J} = J(\xi, \eta) = J_0 + J_\xi \xi + J_\eta \eta$$

The enhanced strains are first assumed in the parent coordinate frame and transformed to the Cartesian frame using a transformation similar to that used in developing the

[2] In general, the determinant of the Jacobian for the two-dimensional Lagrange family of elements will not contain the term with the product of the highest order polynomial, e.g., $\xi\eta$ for the four-node element, $\xi^2\eta^2$ for the nine-node element, etc.

Pian-Sumihara quadrilateral in Example 9.4. Due to the presence of the Jacobian determinant in the strains computed from the displacements (as well as the requirement to later pass the patch test for constant stress states), the enhanced strains are computed from

$$\boldsymbol{\epsilon}_{\text{en}} = \frac{1}{J(\xi,\eta)}\,\mathbf{T}^{\mathrm{T}}\mathbf{E}(\xi,\eta)\mathbf{T}$$

In matrix form this may be written as

$$\left\{\begin{array}{c}\epsilon_x\\ \epsilon_y\\ \gamma_{xy}\end{array}\right\}_{\text{en}} = \frac{1}{J(\xi,\eta)}\left[\begin{array}{ccc} T_{11}^2 & T_{21}^2 & T_{11}T_{21}\\ T_{12}^2 & T_{22}^2 & T_{12}T_{22}\\ 2T_{11}T_{12} & 2T_{21}T_{22}^2 & T_{11}T_{22}+T_{12}T_{21}\end{array}\right]\left\{\begin{array}{c}E_{\xi\xi}\\ E_{\eta\eta}\\ 2E_{\xi\eta}\end{array}\right\}$$

The parent strains (strains with components in the parent element frame) are assumed as

$$\left\{\begin{array}{c}E_{\xi\xi}\\ E_{\eta\eta}\\ 2E_{\xi\eta}\end{array}\right\} = \left[\begin{array}{cccc}\xi & 0 & 0 & 0\\ 0 & \eta & 0 & 0\\ 0 & 0 & \xi & \eta\end{array}\right]\left\{\begin{array}{c}\beta_1\\ \beta_2\\ \beta_3\\ \beta_4\end{array}\right\}$$

The above is motivated by the fact that the derivatives of the shape functions with respect to parent coordinates yield

$$\frac{\partial N_a}{\partial \xi} = a_\xi + b_\xi \eta, \quad \frac{\partial N_a}{\partial \eta} = a_\eta + b_\eta \xi$$

and these may be combined to form strains in the usual manner, but in the parent frame. Thus, by design, the above enhanced strains are specified to generate complete polynomials in the parent coordinates for each strain component. Andelfinger and Ramm [28] and Bischoff et al. [29] discuss the relationship between the design of assumed stress elements using the two-field form and the selection of enhanced strain modes so as to produce the same result.

### *Remarks*

1. The above enhanced strains are defined so that the **C** array is identically zero for constant assumed stresses in each element.
2. Parent normal strains have linearly independent terms added. However, the assumed parent shear strains are linearly dependent. Due to this linear dependence the final shearing strain will usually be nearly constant in each element. Accordingly, to be more explicit, normal strains are *enhanced* while shearing strain is *de-enhanced*.

Since the **C** array vanishes, the equation set to be solved becomes

$$\left[\begin{array}{cc}\mathbf{A} & \mathbf{G}\\ \mathbf{G}^{\mathrm{T}} & \mathbf{K}\end{array}\right]\left\{\begin{array}{c}\tilde{\boldsymbol{\epsilon}}_{\text{en}}\\ \tilde{\mathbf{u}}\end{array}\right\} = \left\{\begin{array}{c}\mathbf{f}_1\\ \mathbf{f}_3\end{array}\right\}$$

and in this form no additional count conditions are apparently needed. The solution may be accomplished partly at the element level by eliminating the equation associated with the enhanced strain parameters. Accordingly,

$$\mathbf{K}^{\star}\tilde{\mathbf{u}} = \mathbf{f}_3^{\star}$$

where

$$\mathbf{K}^{\star} = \mathbf{K} - \mathbf{G}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{G} \quad \text{and} \quad \mathbf{f}_3^{\star} = \mathbf{f}_3 - \mathbf{G}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{f}_1$$

The sensitivity of the enhanced strain element to geometric distortion is evaluated using the problem shown in Fig. 9.8. The transformation from the parent to the global frame is assessed using $\mathbf{T} = \mathbf{J}_0$ and $\mathbf{T} = \mathbf{J}_0^{-\mathrm{T}}$. These are the only options which maintain frame invariance for the element. As observed in Fig. 9.9 the results are now better using the inverse transpose. Since the stress and strain are conjugates in an energy sense, this result could be anticipated from the equivalence relationship

$$E = \frac{1}{2}\int_{\Omega} \boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\epsilon}\, \mathrm{d}\Omega \equiv \frac{1}{2}\int_{\square} \boldsymbol{\Sigma}^{\mathrm{T}}\mathbf{E}\, \mathrm{d}\square$$

where $E$ is energy and $\square$ denotes the domain of the element in the parent coordinate system (i.e., the bi-unit square for a quadrilateral element).

The performance of the enhanced element is compared to the Pian-Sumihara element for a shear loading on the mesh shown in Fig. 9.10. In Fig. 9.11 the convergence

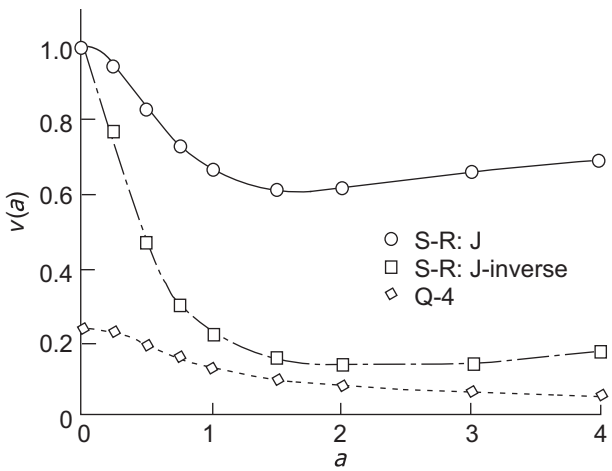



**FIGURE 9.9**

Simo-Rifai enhanced strain quadrilateral (S-R) compared with displacement quadrilateral (Q-4). Effect of element distortion (Exact = 1.0).

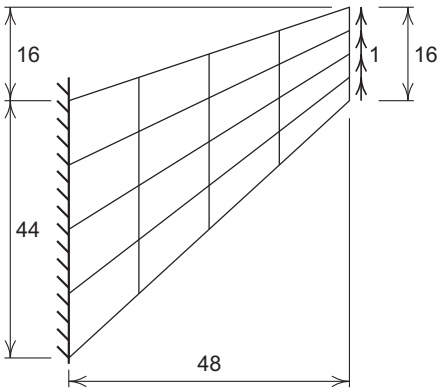



**FIGURE 9.10**
Mesh with 4 × 4 elements for unit end shear load.

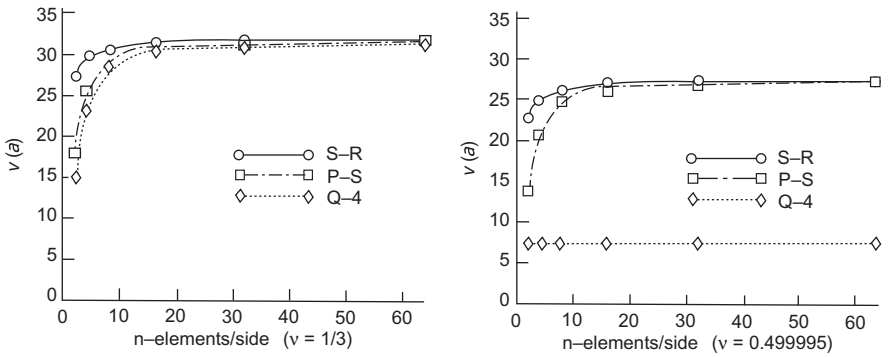



**FIGURE 9.11**
Convergence behavior for (a) $\nu = 1/3$ and (b) $\nu = 0.499995$.

results for various order meshes are shown for linear elastic, plane strain conditions with (a) $E = 70$ and $\nu = 1/3$ and (b) for $E = 70$ and $\nu = 0.499995$. The results shown in Fig. 9.11 clearly show the strong dependence of the displacement formulation on Poisson's ratio—namely the tendency for the element to *lock* for values which approach the incompressibility limit of $\nu = 1/2$. On the other hand, the performance of both the enhanced strain and the Pian-Sumihara element is nearly insensitive to the value of Poisson's ratio selected, with somewhat better performance of the enhanced element on coarse meshing.

## 9.6 Complementary forms with direct constraint

### 9.6.1 General forms

In the introduction to this chapter we defined the irreducible and mixed forms and indicated that on occasion it is possible to obtain more than one "irreducible" form. To illustrate this in the problem of heat transfer given by Eqs. (9.2) and (9.3) we introduced a penalty function $\alpha$ in Eq. (9.6) and derived a corresponding single governing equation (9.7) given in terms of $\mathbf{q}$. This penalty function here has no obvious physical meaning and served simply as a device to obtain a *close enough* approximation to the satisfaction of the continuity of flow equations.

On occasion it is possible to solve the problem as an irreducible one assuming *a priori* that the choice of the variable satisfies one of the equations. We call such forms *directly constrained* and obviously the choice of the shape function becomes difficult.

We shall consider two examples.

#### 9.6.1.1 The complementary heat transfer problem

In this we assume *a priori* that the choice of $\mathbf{q}$ is such that it satisfies Eq. (9.3) and the natural boundary conditions

$$\nabla^{\mathrm{T}}\mathbf{q} = -Q \quad \text{in } \Omega \qquad \text{and} \qquad q_n = \mathbf{q}^{\mathrm{T}}\mathbf{n} = \bar{q}_n \quad \text{on } \Gamma_q \tag{9.46}$$

where $\mathbf{n}$ is the unit normal to the boundary. Thus we only have to satisfy the constitutive relation (9.2), i.e.,

$$\mathbf{k}^{-1}\mathbf{q} + \nabla\phi = \mathbf{0} \quad \text{in } \Omega \quad \text{with } \phi = \bar{\phi} \quad \text{on } \Gamma_\phi \tag{9.47}$$

A weak statement of the above is

$$\int_\Omega \delta\mathbf{q}^{\mathrm{T}}(\mathbf{k}^{-1}\mathbf{q} + \nabla\phi)\mathrm{d}\Omega - \int_{\Gamma_\phi} \delta q_n(\phi - \bar{\phi})\mathrm{d}\Gamma = 0 \tag{9.48}$$

in which $\delta q_n = \delta\mathbf{q}^{\mathrm{T}}\mathbf{n}$ represents the variation of normal flux on the boundary.

Use of Green's theorem transforms the above into

$$\int_\Omega \delta\mathbf{q}^{\mathrm{T}}\mathbf{k}^{-1}\mathbf{q}\,\mathrm{d}\Omega - \int_\Omega \nabla^{\mathrm{T}}\delta\mathbf{q}\phi\,\mathrm{d}\Omega + \int_{\Gamma_\phi} \delta q_n\bar{\phi}\,\mathrm{d}\Gamma + \int_{\Gamma_q} \delta q_n\phi\,\mathrm{d}\Gamma = 0 \tag{9.49}$$

If we further assume that $\nabla^{\mathrm{T}}\delta\mathbf{q} \equiv 0$ in $\Omega$ and $\delta q_n = 0$ on $\Gamma_q$, i.e., that the weighting functions are simply the variations of $\mathbf{q}$, the equation reduces to

$$\int_\Omega \delta\mathbf{q}^{\mathrm{T}}\mathbf{k}^{-1}\mathbf{q}\,\mathrm{d}\Omega + \int_{\Gamma_\phi} \delta q_n\bar{\phi}\,\mathrm{d}\Gamma = 0 \tag{9.50}$$

This is in fact the variation of a complementary flux principle

$$\Pi = \int_\Omega \frac{1}{2}\mathbf{q}^{\mathrm{T}}\mathbf{k}^{-1}\mathbf{q}\,\mathrm{d}\Omega + \int_{\Gamma_\phi} q_n\bar{\phi}\,\mathrm{d}\Gamma \tag{9.51}$$

Numerical solutions can obviously be started from either of the above equations but the difficulty is the choice of the trial function satisfying the constraints. We shall return to this problem in Section 9.6.2.

#### 9.6.1.2 The complementary elastic energy principle

In the elasticity problem specified in Section 9.4 we can proceed similarly, assuming stress fields which satisfy the equilibrium conditions both on the boundary $\Gamma_t$ and in the domain $\Omega$.

Thus in an analogous manner to that of the previous example we impose on the permissible stress field the constraints which we assume to be satisfied by the approximation identically, i.e.,

$$\mathcal{S}^{\mathrm{T}}\boldsymbol{\sigma} + \mathbf{b} = \mathbf{0} \quad \text{in } \Omega \quad \text{and} \quad \mathbf{t} = \bar{\mathbf{t}} \quad \text{on } \Gamma_t \tag{9.52}$$

Thus only the constitutive relations and displacement boundary conditions remain to be satisfied, i.e.,

$$\mathbf{D}^{-1}\boldsymbol{\sigma} - \mathcal{S}\mathbf{u} = \mathbf{0} \quad \text{in } \Omega \quad \text{and} \quad \mathbf{u} = \bar{\mathbf{u}} \quad \text{on } \Gamma_u \tag{9.53}$$

The weak statement of the above can be written as

$$\int_\Omega \delta\boldsymbol{\sigma}^{\mathrm{T}}(\mathbf{D}^{-1}\boldsymbol{\sigma} - \mathcal{S}\mathbf{u})\mathrm{d}\Omega + \int_{\Gamma_u} \delta\mathbf{t}^{\mathrm{T}}(\mathbf{u} - \bar{\mathbf{u}})\mathrm{d}\Gamma = 0 \tag{9.54}$$

which on integration by Green's theorem gives

$$\int_\Omega \delta\boldsymbol{\sigma}^{\mathrm{T}}\mathbf{D}^{-1}\boldsymbol{\sigma}\,\mathrm{d}\Omega + \int_\Omega (\mathcal{S}^{\mathrm{T}}\delta\boldsymbol{\sigma})^{\mathrm{T}}\mathbf{u}\,\mathrm{d}\Omega - \int_{\Gamma_u} \delta\mathbf{t}^{\mathrm{T}}\bar{\mathbf{u}}\,\mathrm{d}\Gamma - \int_{\Gamma_t} \delta\mathbf{t}^{\mathrm{T}}\mathbf{u}\,\mathrm{d}\Gamma = 0 \tag{9.55}$$

Again assuming that the test functions are complete variations satisfying the homogeneous equilibrium equation, i.e.,

$$\mathcal{S}^{\mathrm{T}}\delta\boldsymbol{\sigma} = \mathbf{0} \quad \text{in } \Omega \quad \text{and} \quad \delta\mathbf{t} = \mathbf{0} \quad \text{on } \Gamma_t \tag{9.56}$$

we have as the weak statement

$$\int_\Omega \delta\boldsymbol{\sigma}^{\mathrm{T}}\mathbf{D}^{-1}\boldsymbol{\sigma}\,\mathrm{d}\Omega - \int_{\Gamma_u} \delta\mathbf{t}^{\mathrm{T}}\bar{\mathbf{u}}\,\mathrm{d}\Gamma = 0 \tag{9.57}$$

The corresponding complementary energy variational principle is

$$\Pi = \frac{1}{2}\int_\Omega \boldsymbol{\sigma}^{\mathrm{T}}\mathbf{D}^{-1}\boldsymbol{\sigma}\,\mathrm{d}\Omega - \int_{\Gamma_u} \mathbf{t}^{\mathrm{T}}\bar{\mathbf{u}}\,\mathrm{d}\Gamma \tag{9.58}$$

Once again in practical use the difficulties connected with the choice of the approximating function arise but on occasion a direct choice is possible [30].

### 9.6.2 Solution using auxiliary functions

Both the complementary forms can be solved using auxiliary functions to ensure the satisfaction of the constraints.

**Example 9.6.** Heat transfer solution by potential function

In the *heat transfer problem* it is easy to verify that the homogeneous equation

$$\boldsymbol{\nabla}^{\mathrm{T}}\mathbf{q} \equiv \frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} = 0 \tag{9.59}$$

is automatically satisfied by defining a function $\psi$ such that

$$q_x = \frac{\partial \psi}{\partial y}, \quad q_y = -\frac{\partial \psi}{\partial x} \tag{9.60}$$

Thus we define

$$\mathbf{q} = \mathcal{L}\psi + \mathbf{q}_0 \quad \text{and} \quad \delta\mathbf{q} = \mathcal{L}\delta\psi \tag{9.61}$$

where $\mathbf{q}_0$ is any flux chosen so that

$$\boldsymbol{\nabla}^{\mathrm{T}}\mathbf{q}_0 = -Q \tag{9.62}$$

and

$$\mathcal{L} = \left[\frac{\partial}{\partial y} - \frac{\partial}{\partial x}\right]^{\mathrm{T}} \tag{9.63}$$

The formulations of Eqs. (9.50) and (9.51) can be used without any constraints and, for instance, the stationarity

$$\Pi = \int_{\Omega} \frac{1}{2}(\mathcal{L}\psi + \mathbf{q}_0)^{\mathrm{T}}\mathbf{k}^{-1}(\mathcal{L}\psi + \mathbf{q}_0)\mathrm{d}\Omega - \int_{\Gamma_\phi} \left(\frac{\partial \psi}{\partial s}\right)\bar{\phi}\,\mathrm{d}\Gamma \tag{9.64}$$

will suffice to so formulate the problem (here $s$ is the tangential direction to the boundary).

The above form will require shape functions for $\psi$ satisfying $\mathrm{C}_0$ continuity. The reader should observe the formulation is identical to the stress function used to solve the torsion problem (viz. Section 5.3.1).

**Example 9.7.** Elasticity solution by Airy stress function

In the elasticity problem a two-dimensional form can be obtained by the use of the so-called Airy stress function $\psi$ [31].

Now the equilibrium equations

$$\boldsymbol{\mathcal{S}}^{\mathrm{T}}\boldsymbol{\sigma} + \mathbf{b} \equiv \begin{Bmatrix} \dfrac{\partial \sigma_x}{\partial x} + \dfrac{\partial \tau_{xy}}{\partial y} + b_x \\ \dfrac{\partial \tau_{xy}}{\partial x} + \dfrac{\partial \sigma_y}{\partial y} + b_y \end{Bmatrix} = \mathbf{0} \tag{9.65}$$

are identically solved by choosing

$$\boldsymbol{\sigma} = \mathcal{L}\psi + \boldsymbol{\sigma}_0 \tag{9.66}$$

where

$$\mathcal{L} = \left[ \frac{\partial^2}{\partial y^2} \quad \frac{\partial^2}{\partial x^2} \quad - \quad \frac{\partial^2}{\partial x \, \partial y} \right]^{\mathrm{T}} \tag{9.67}$$

and $\boldsymbol{\sigma}_0$ is an arbitrary stress chosen so that

$$\boldsymbol{\mathcal{S}}^{\mathrm{T}} \boldsymbol{\sigma}_0 + \mathbf{b} = \mathbf{0} \tag{9.68}$$

Again the substitution of (9.66) into the weak statement (9.57) or the complementary variational problem (9.58) will yield a direct formulation to which no additional constraints need be applied. However, use of the above forms does lead to further complexity in multiply connected regions where further conditions are needed. The reader will note that in Chapter 5 we encountered this in a similar problem in torsion and suggested a very simple procedure of avoidance (see Section 5.3.1).

The use of this stress function formulation in the two-dimensional context was first made by Fraeijs de Veubeke and Zienkiewicz [32] and Elias [33], but the reader should note that now with second-order operators present, $C_1$ continuity of shape functions is needed in a similar manner to the problems which we will consider in plate bending (see Chapter 13).

Incidentally, analogies with plate bending go further here and indeed it can be shown that some of these can be usefully employed for other problems [34].

## 9.7 Concluding remarks: Mixed formulation or a test of element "robustness"

The mixed form of finite element formulation outlined in this chapter opens a new range of possibilities, many with potentially higher accuracy and robustness than those offered by irreducible forms. However, an additional advantage arises even in situations where, by the *limitation principle*, the irreducible and mixed forms yield identical results. Here the study of the behavior of the mixed form can frequently reveal weaknesses or lack of "robustness" in the irreducible form which otherwise would be difficult to determine.

The mixed approximation, if properly understood, expands the potential of the finite element method and presents almost limitless possibilities of detailed improvement. Some of these will be discussed further in the subsequent chapters, and others in Ref. [35].

## 9.8 Problems

**9.1** Show that the stationarity of the variational principle given by

$$\Pi_{\mathrm{HW}} = \int_\Omega \frac{1}{2} \boldsymbol{\epsilon}^{\mathrm{T}} \mathbf{D} \boldsymbol{\epsilon} \, \mathrm{d}\Omega - \int_\Omega \boldsymbol{\sigma}^{\mathrm{T}} (\boldsymbol{\epsilon} - \boldsymbol{\mathcal{S}} \mathbf{u}) \mathrm{d}\Omega - \int_\Omega \mathbf{u}^{\mathrm{T}} \mathbf{b} \, \mathrm{d}\Omega - \int_{\Gamma_t} \mathbf{u}^{\mathrm{T}} \bar{\mathbf{t}} \, \mathrm{d}\Gamma$$

where $\mathbf{u} \equiv \bar{\mathbf{u}}$ on $\Gamma_u$ is equivalent to Eq. (9.33).

**9.2** Using the variational principle of Problem 9.1 with the approximations (9.34) show that the stationarity conditions give (9.35) and (9.36).

**9.3** Show that the variational principle given by stationarity of

$$\Pi_{\text{en}} = \int_\Omega \frac{1}{2} \left(\mathcal{S}\mathbf{u} + \boldsymbol{\epsilon}_{\text{en}}\right)^{\text{T}} \mathbf{D} \left(\mathcal{S}\mathbf{u} + \boldsymbol{\epsilon}_{\text{en}}\right) \text{d}\Omega + \int_\Omega \boldsymbol{\sigma}^{\text{T}} \boldsymbol{\epsilon}_{\text{en}} \, \text{d}\Omega - \int_\Omega \mathbf{u}^{\text{T}} \mathbf{b} \, \text{d}\Omega - \int_{\Gamma_t} \mathbf{u}^{\text{T}} \bar{\mathbf{t}} \, \text{d}\Gamma$$

with $\mathbf{u} = \bar{\mathbf{u}}$ enforced on $\Gamma_u$ is equivalent to Eq. (9.41).

**9.4** For the rectangular element shown in Fig. 9.7 develop the expressions for $\beta_i$ for the Pian-Sumihara element described in Example 9.3. For an isotropic elastic material and a plane stress problem compute the expressions for the stresses which result from the strains (these are those of the displacement model described in Chapter 7). How do these differ from those assumed for the mixed element?

**9.5** For the enhanced strain formulation described in Section 9.5.3 use the constant stress patch test for a plane strain problem to show that $\boldsymbol{\epsilon}_{\text{en}} = \mathbf{0}$.
Show that a necessary condition to satisfy this requirement is

$$\int_{\Omega_e} N_{\text{en}} \, \text{d}\Omega = 0$$

**9.6** Generalize the Simo-Rifai quadrilateral given as Example 9.5 in Section 9.5.3 for a three-dimensional solid modeled by eight-node hexahedral elements.

**9.7** Generalize the Simo-Rifai quadrilateral given as Example 9.5 in Section 9.5.3 for an axisymmetric geometry.

**9.8** A plane stress problem has the geometry shown in Fig. 9.10 and is loaded by a uniformly distributed shear traction (i.e., $t_y = \text{const.}$). Use *FEAPpv* to solve the problem using a series of three-node triangular meshes. The first mesh should be as shown with each quadrilateral divided into two triangles. Consider two values for the elastic properties: (a) $E = 70$, $\nu = 1/3$ and (b) $E = 70$, $\nu = 0.499995$. Let the thickness of the slab be one unit.
Next, perform the solution using four-node quadrilaterals based on (a) the displacement solution described in Chapter 7; (b) the Simo-Rifai enhanced element described in Section 9.5.3. Plot the displacement convergence for the top and bottom points at the loaded end. Plot contours for displacement and principal stresses.
Repeat the calculations assuming plane strain conditions.
Briefly discuss your findings.

## References

[1] S.N. Atluri, R.H. Gallagher, O.C. Zienkiewicz (Eds.), Hybrid and Mixed Finite Element Methods, John Wiley & Sons, New York, 1983.

[2] O.C. Zienkiewicz, R.L. Taylor, J.A.W. Baynham, Mixed and irreducible formulations in finite element analysis, in: S.N. Atluri, R.H. Gallagher, O.C. Zienkiewicz (Eds.), Hybrid and Mixed Finite Element Methods, John Wiley & Sons, 1983, pp. 405–431.
[3] I. Babuška, J.E. Osborn, Generalized finite element methods and their relations to mixed problems, SIAM J. Numer. Anal. 30 (1983) 510–536.
[4] R.L. Taylor, O.C. Zienkiewicz, Complementary energy with penalty function in finite element analysis, in: R. Glowinski, E.Y. Rodin, O.C. Zienkiewicz (Eds.), Energy Methods in Finite Element Analysis, John Wiley & Sons, Chichester, 1979 (Chapter 8).
[5] K. Washizu, Variational Methods in Elasticity and Plasticity, third ed., Pergamon Press, New York, 1982.
[6] L.R. Herrmann, Finite element bending analysis of plates, in: Proceedings of the First Conference on Matrix Methods in Structural Mechanics, AFFDL-TR-66-80, Wright-Patterson Air Force Base, Ohio, 1965, pp. 577–602.
[7] K. Hellan, Analysis of elastic plates in flexure by a simplified finite element method, Technical Report Civ. Eng. Series 46, Acta Polytechnica Scandinavia, Trondheim, 1967.
[8] R.S. Dunham, K.S. Pister, A finite element application of the Hellinger-Reissner variational theorem, in: Proceedings of the First Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-66-80, Wright Patterson Air Force Base, Ohio, October 1966.
[9] R.L. Taylor, O.C. Zienkiewicz, Mixed finite element solution of fluid flow problems, in: R.H. Gallagher, G.F. Carey, J.T. Oden, O.C. Zienkiewicz (Eds.), Finite Elements in Fluids, vol. 1, John Wiley & Sons, 1982, pp. 1–20 (Chapter 4).
[10] B. Fraeijs de Veubeke, Displacement and equilibrium models in finite element method, in: O.C. Zienkiewicz, G.S. Holister (Eds.), Stress Analysis, John Wiley & Sons, Chichester, 1965, pp. 145–197 (Chapter 9).
[11] I. Babuška, Error bounds for finite element methods, Numer. Math. 16 (1971) 322–333.
[12] I. Babuška, The finite element method with Lagrangian multipliers, Numer. Math. 20 (1973) 179–192.
[13] F. Brezzi, On the existence, uniqueness and approximation of saddle-point problems arising from Lagrange multipliers, R.A.I.R.O. 8 (R-2) (1974) 129–151.
[14] O.C. Zienkiewicz, S. Qu, R.L. Taylor, S. Nakazawa, The patch test for mixed formulations, Int. J. Numer. Methods Eng. 23 (1986) 1873–1883.
[15] J.T. Oden, N. Kikuchi, Finite element methods for constrained problems in elasticity, Int. J. Numer. Methods Eng. 18 (1982) 701–725.
[16] E. Hellinger, Die allgemeine Aussetze der Mechanik der Kontinua, in: F. Klein, C. Muller (Eds.), Encyclopedia der Mathematishen Wissnschaften, vol. 4, Tebner, Leipzig, 1914.
[17] E. Reissner, On a variational theorem in elasticity, J. Math. Phys. 29 (2) (1950) 90–95.

[18] L.R. Herrmann, Finite element bending analysis of plates, J. Eng. Mech., ASCE 94 (EM5) (1968) 13–25.
[19] L.R. Herrmann, D.M. Campbell, Finite element analysis for thin shells, J. AIAA 6 (1968) 1842–1847.
[20] O.C. Zienkiewicz, D. Lefebvre, Mixed methods for FEM and the patch test. Some recent developments, in: F. Murat, O. Pirenneau (Eds.), Analyse Mathematique of Application, Gauthier Villars, Paris, 1988.
[21] P.-A. Raviart, J.M. Thomas, A mixed finite element method for second order elliptic problems, in: Lecture Notes in Mathematics, No. 606, Springer-Verlag, Berlin, 1977, pp. 292–315.
[22] D.N. Arnold, F. Brezzi, J. Douglas, PEERS, a new mixed finite element for plane elasticity, Jpn. J. Appl. Math. 1 (1984) 347–367.
[23] T.H.H. Pian, K. Sumihara, Rational approach for assumed stress finite elements, Int. J. Numer. Methods Eng. 20 (1985) 1685–1695.
[24] O.C. Zienkiewicz, D. Lefebvre, Three field mixed approximation and the plate bending problem, Commun. Appl. Numer. Methods 3 (1987) 301–309.
[25] T.J.R. Hughes, Generalization of selective integration procedures to anisotropic and non-linear media, Int. J. Numer. Methods Eng. 15 (1980) 1413–1418.
[26] J.C. Simo, R.L. Taylor, K.S. Pister, Variational and projection methods for the volume constraint in finite deformation plasticity, Comput. Methods Appl. Mech. Eng. 51 (1985) 177–208.
[27] J.C. Simo, M.S. Rifai, A class of mixed assumed strain methods and the method of incompatible modes, Int. J. Numer. Methods Eng. 29 (1990) 1595–1638.
[28] U. Andelfinger, E. Ramm, EAS-elements for two-dimensional, three-dimensional, plate and shell structures and their equivalence to HR-elements, Int. J. Numer. Methods Eng. 36 (1993) 1311–1337.
[29] M. Bischoff, E. Ramm, D. Braess, A class of equivalent enhanced assumed strain and hybrid stress finite elements, Comput. Mech. 22 (1999) 443–449.
[30] C. Loubignac, G. Cantin, C. Touzot, Continuous stress fields in finite element analysis, J. AIAA 15 (1978) 1645–1647.
[31] S.P. Timoshenko, J.N. Goodier, Theory of Elasticity, third ed., McGraw-Hill, New York, 1969.
[32] B. Fraeijs de Veubeke, O.C. Zienkiewicz, Strain energy bounds in finite element analysis by slab analogy, J. Strain Anal. 2 (1967) 265–271.
[33] Z.M. Elias, Duality in finite element methods, Proc. Am. Soc. Civ. Eng. 94 (EM4) (1968) 931–946.
[34] R.V. Southwell, On the analogues relating flexure and displacement of flat plates, Quart. J. Mech. Appl. Math. 3 (1950) 257–270.
[35] O.C. Zienkiewicz, R.L. Taylor, P. Nithiarasu, The Finite Element Method for Fluid Dynamics, seventh ed., Elsevier, Oxford, 2013.

CHAPTER

# 10 Incompressible Problems, Mixed Methods, and Other Procedures of Solution

## 10.1 Introduction

We noted earlier that the standard displacement formulation of elastic problems fails when the Poisson's ratio $\nu$ becomes 0.5 or when the material becomes incompressible. Indeed, problems can arise even when the material is nearly incompressible with $\nu > 0.4$.

The application of a mixed formulation for such problems can avoid the difficulties and is of great practical interest as *nearly* incompressible behavior is encountered in a variety of real engineering problems ranging from soil mechanics to aerospace engineering. Identical problems also arise when the flow of incompressible fluids is encountered.

In this chapter we shall discuss mixed approaches to incompressible problems, generally using a two-field manner where the displacement $\mathbf{u}$ and the pressure $p$ are the variables.[1] Such formulation will allow us to deal with full incompressibility as well as near incompressibility as it occurs. However, what we will find is that the interpolations used will be very much limited by the stability conditions of the mixed patch test. For this reason much interest has been focused on the development of so-called *stabilized* procedures in which the violation of the mixed patch test (or Babuška-Brezzi conditions) is artificially compensated. A part of this chapter will be devoted to such stabilized methods.

## 10.2 Deviatoric stress and strain, pressure, and volume change

The main problem in the application of a "standard" displacement formulation to incompressible or nearly incompressible problems lies in the determination of the mean stress or pressure which is related to the volumetric part of the strain (for isotropic materials). For this reason it is convenient to separate this from the total stress field and treat it as an independent variable. Using the Voigt notation of stress,

[1] Similar mixed approaches apply also to fluid problems where $\mathbf{u}$ is velocity [1].

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00010-1

recall from Section 2.2.8.2 that the mean stress or pressure is given by

$$p = \frac{1}{3}\left(\sigma_x + \sigma_y + \sigma_z\right) = \frac{1}{3}\mathbf{m}^{\mathrm{T}}\boldsymbol{\sigma} \tag{10.1}$$

where $\mathbf{m}$ for the general three-dimensional state of stress is given by

$$\mathbf{m} = \left[1,\ 1,\ 1,\ 0,\ 0,\ 0\right]^{\mathrm{T}}$$

For isotropic behavior the "pressure" is related to the volumetric strain, $\varepsilon_v$, by the bulk modulus of the material, $K$. Thus,

$$\varepsilon_v = \varepsilon_x + \varepsilon_y + \varepsilon_z = \mathbf{m}^{\mathrm{T}}\boldsymbol{\epsilon} = \frac{p}{K} \tag{10.2}$$

For an incompressible material $K = \infty$ ($\nu \equiv 0.5$) and the volumetric strain is simply zero.

The deviatoric strain $\boldsymbol{\epsilon}^d$ is defined by

$$\boldsymbol{\epsilon}^d = \boldsymbol{\epsilon} - \tfrac{1}{3}\mathbf{m}\varepsilon_v \equiv \left(\mathbf{I} - \tfrac{1}{3}\mathbf{m}\mathbf{m}^{\mathrm{T}}\right)\boldsymbol{\epsilon} = \mathbf{I}_d\boldsymbol{\epsilon} \tag{10.3}$$

where $\mathbf{I}_d$ is a deviatoric projection matrix which also proves useful in problems with more general constitutive relations [2]. In isotropic elasticity the deviatoric strain is related to the deviatoric stress by the shear modulus $G$ as

$$\boldsymbol{\sigma}^d = \boldsymbol{\sigma} - \mathbf{B}mp = \mathbf{B}I_d\boldsymbol{\sigma} = 2G\mathbf{I}_0\boldsymbol{\epsilon}^d = 2G\left(\mathbf{I}_0 - \tfrac{1}{3}\mathbf{m}\mathbf{m}^{\mathrm{T}}\right)\boldsymbol{\epsilon} \tag{10.4}$$

where the diagonal matrix [viz. Eq. (2.58a)]

$$\mathbf{I}_0 = \frac{1}{2}\begin{bmatrix} 2 & & & & & \\ & 2 & & & & \\ & & 2 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \end{bmatrix}$$

is introduced because of the Voigt notation. A deviatoric form for the elastic moduli of an isotropic material is written as

$$\mathbf{D}_d = 2G\left(\mathbf{I}_0 - \tfrac{1}{3}\mathbf{m}\mathbf{m}^{\mathrm{T}}\right) \tag{10.5}$$

for convenience in writing subsequent equations.

The above relationships are but an alternate way of determining the stress strain relations shown in Chapters 2 and 7, with the material parameters related through

$$\begin{aligned} G &= \frac{E}{2\left(1+\nu\right)} \\ K &= \frac{E}{3\left(1-2\nu\right)} \end{aligned} \tag{10.6}$$

and indeed Eqs. (10.4) and (10.2) can be used to define the standard $\mathbf{D}$ matrix in an alternative manner.

## 10.3 Two-field incompressible elasticity (u-*p* form)

In the mixed form considered next we shall use as variables the displacement $\mathbf{u}$ and the pressure $p$.

Now the equilibrium equation (9.22) is rewritten using (10.4), treating $p$ as an independent variable, as

$$\int_\Omega \delta\boldsymbol{\epsilon}^{\mathrm{T}}\mathbf{D}_d\boldsymbol{\epsilon}\,\mathrm{d}\Omega + \int_\Omega \delta\boldsymbol{\epsilon}^{\mathrm{T}}\mathbf{m}\,p\,\mathrm{d}\Omega - \int_\Omega \delta\mathbf{u}^{\mathrm{T}}\mathbf{b}\,\mathrm{d}\Omega - \int_{\Gamma_t} \delta\mathbf{u}^{\mathrm{T}}\bar{\mathbf{t}}\,\mathrm{d}\Gamma = 0 \tag{10.7}$$

and in addition we shall impose a weak form of Eq. (10.2), i.e.,

$$\int_\Omega \delta p\left[\mathbf{m}^{\mathrm{T}}\boldsymbol{\epsilon} - \frac{p}{K}\right]\mathrm{d}\Omega = 0 \tag{10.8}$$

with $\boldsymbol{\epsilon} = \mathcal{S}\mathbf{u}$. Independent approximation of $\mathbf{u}$ and $p$ as

$$\mathbf{u} \approx \hat{\mathbf{u}} = \mathbf{N}_u\tilde{\mathbf{u}} \quad \text{and} \quad p \approx \hat{p} = \mathbf{N}_p\tilde{\mathbf{p}} \tag{10.9}$$

immediately gives the mixed approximation in the form

$$\begin{bmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C}^{\mathrm{T}} & -\mathbf{V} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}} \\ \tilde{\mathbf{p}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{Bmatrix} \tag{10.10}$$

where

$$\begin{aligned} \mathbf{A} &= \int_\Omega \mathbf{B}^{\mathrm{T}}\mathbf{D}_d\,\mathbf{B}\,\mathrm{d}\Omega, \quad \mathbf{C} = \int_\Omega \mathbf{B}^{\mathrm{T}}\mathbf{m}\mathbf{N}_p\,\mathrm{d}\Omega \\ \mathbf{V} &= \int_\Omega \mathbf{N}_p^{\mathrm{T}}\frac{1}{K}\mathbf{N}_p\,\mathrm{d}\Omega, \quad \mathbf{f}_1 = \int_\Omega \mathbf{N}_u^{\mathrm{T}}\mathbf{b}\,\mathrm{d}\Omega + \int_{\Gamma_t} \mathbf{N}_u^{\mathrm{T}}\bar{\mathbf{t}}\,\mathrm{d}\Gamma, \quad \mathbf{f}_2 = \mathbf{0} \end{aligned} \tag{10.11}$$

We note that for incompressible situations the equations are of the "standard" form, see Eq. (9.14) with $\mathbf{V} = \mathbf{0}$ (as $K = \infty$), but the formulation is useful in practice when $K$ has a high value (or $\nu \to 0.5$).

A formulation similar to that above and using a corresponding variational theorem was first proposed by Herrmann [3] and later generalized by Key [4] for anisotropic elasticity. The arguments concerning stability (or singularity) of the matrices which we presented in Section 9.3 are again of great importance in this problem.

Clearly the mixed patch condition about the number of degrees of freedom now yields [see Eq. (9.18)]

$$n_u \geq n_p \tag{10.12}$$

and *is necessary* for prevention of locking (or instability) with the pressure $p$ acting now as the constraint variable of the Lagrangian multiplier enforcing zero volumetric strain.

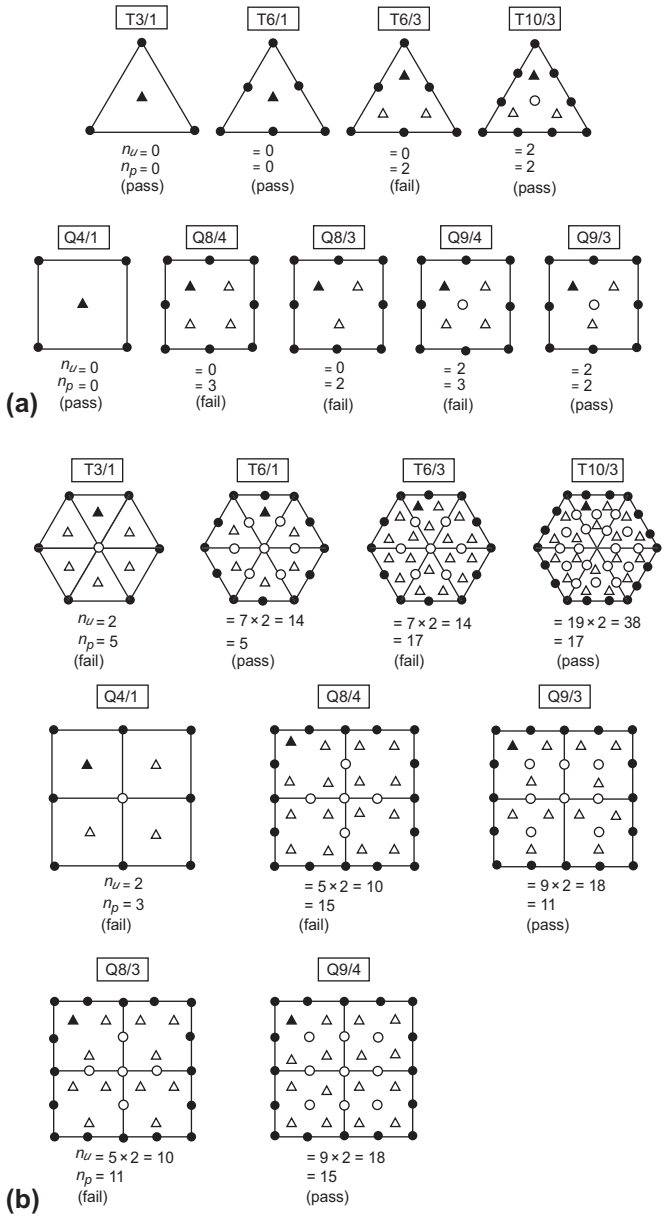



**FIGURE 10.1**

Incompressible elasticity $\mathbf{u}$-$p$ formulation. Discontinuous pressure approximation. (a) Single-element patch tests. (b) Multiple-element patch tests.

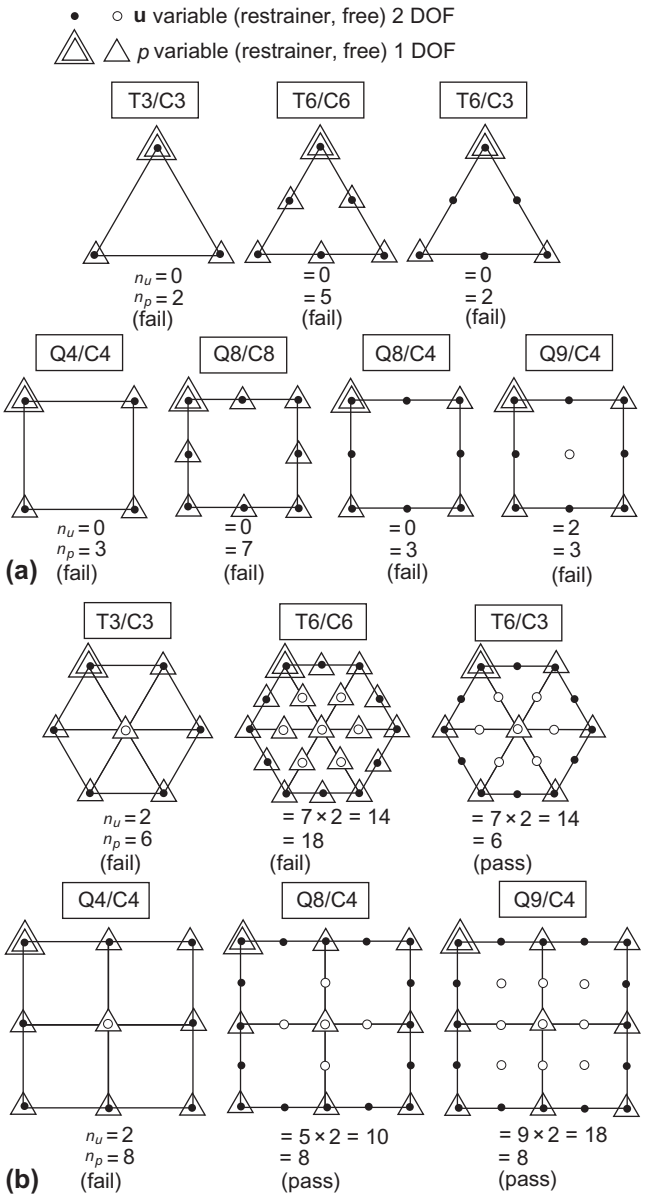



**FIGURE 10.2**

Incompressible elasticity **u**-$p$ formulation. Continuous ($C_0$) pressure approximation. (a) Single-element patch tests. (b) Multiple-element patch tests.

In the form of a patch test this condition is most critical and we show in Figs. 10.1 and 10.2 a series of such patch tests on elements with $C_0$ continuous interpolation of $\mathbf{u}$ and either discontinuous or continuous interpolation of $p$. For each we have included all combinations of constant, linear, and quadratic functions.

In the test we prescribe *all* the displacements on the boundaries of the patch and one pressure variable as it is well known that in fully incompressible situations pressure will be indeterminate by a constant for the problem with all boundary displacements prescribed.[2]

The single-element test is very stringent and eliminates most continuous pressure approximations whose performance is known to be acceptable in many situations. For this reason we attach more importance to the assembly test and it would appear that the following elements could be permissible according to the criteria of Eq. (10.12):

*Triangles:* T6/1, T10/3, T6/C3
*Quadrilaterals:* Q9/3, Q8/C4, Q9/C4

We note, however, that in practical applications quite adequate answers have been reported with Q4/1, Q8/3, and Q9/4 quadrilaterals, although severe oscillations of $p$ may occur. If full robustness is sought the choice of the elements is limited [5].

It is unfortunate that in the present "acceptable" list, the linear triangle and linear quadrilateral are missing. This appreciably restricts the use of these simplest elements. A possible and indeed effective procedure here is to not apply the pressure constraint at the level of a single element but on an assembly. This was done by Herrmann in his original presentation [3] where four elements were chosen for such a constraint as shown in Fig. 10.3a. This composite "element" passes the single-element (and multiple-element) patch tests but apparently so do several others fitting into this category. In Fig. 10.3b we show how a single triangle can be internally subdivided into three parts by the introduction of a central node. This coupled with constant pressure on the assembly allows the necessary count condition to be satisfied and a standard element procedure applies to the original triangle treating the central node as an internal variable. Indeed, the same effect could be achieved by the introduction of any other internal element function which gives zero value on the main triangle perimeter. Such a *bubble function* can simply be written in terms of the area coordinates (see Chapter 6) as $L_1L_2L_3$. However, as we have stated before, the degree of freedom count is a necessary but not sufficient condition for stability and a direct rank test is always required. In particular it can be verified by algebra that the conditions stated in Section 9.3 are not fulfilled for this triple subdivision of a linear triangle (or the case with the bubble function) and thus

$$\mathbf{Cp} = \mathbf{0} \text{ for some nonzero values of } \mathbf{p}$$

indicating instability.

[2] Alternatively, it is possible to omit all boundary conditions on pressure if one displacement with a component normal to the boundary is allowed to exist.

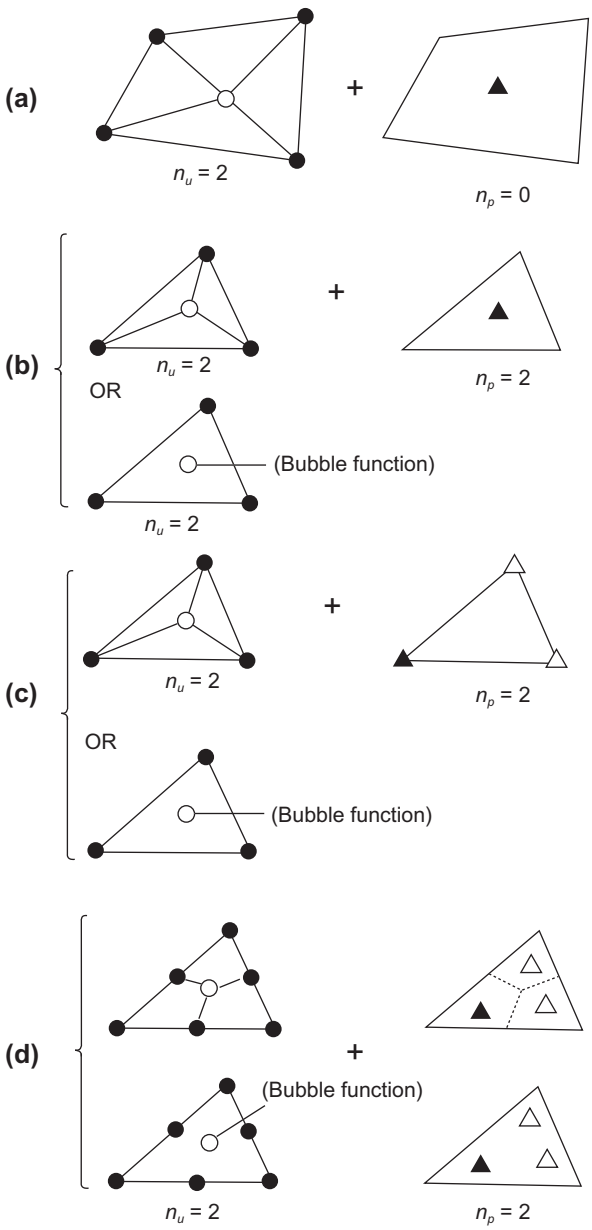



**FIGURE 10.3**

Some simple combinations of linear triangles and quadrilaterals that pass the necessary patch test counts. Combinations (a), (c), and (d) are successful but (b) is still singular and not usable.

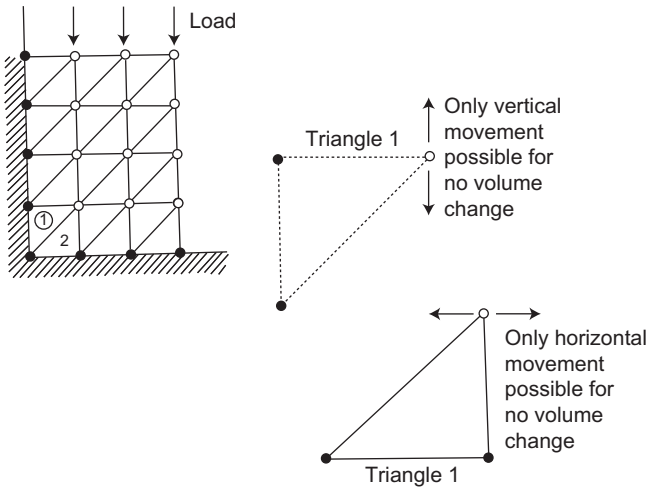



**FIGURE 10.4**
Locking (zero displacements) of a simple assembly of linear triangles for which incompressibility is fully required ($n_p = n_u = 24$).

In Fig. 10.3c we show, however, that the same concept can be used with good effect for $C_0$ continuous $p$ [6]. Similar internal subdivision into quadrilaterals or the introduction of bubble functions in quadratic triangles can be used, as shown in Fig. 10.3d, with success.

The performance of all the elements mentioned above has been extensively discussed [7–12] but detailed comparative assessment of merit is difficult. As we have observed, it is essential to have $n_u \geq n_p$ but if near equality is only obtained in a large problem no meaningful answers will result for $\mathbf{u}$ as we observe, for example, in Fig. 10.4 in which linear triangles for $\mathbf{u}$ are used with the element constant $p$. Here the only permissible answer is of course $\mathbf{u} = \mathbf{0}$ as the triangles have to preserve constant volumes.

The ratio $n_u/n_p$ which occurs as the field of elements is enlarged gives some indication of the relative performance, and we show this in Fig. 10.5. This approximates to the behavior of a very large element assembly, but of course for any practical problem such a ratio will depend on the boundary conditions imposed.

We see that for the discontinuous pressure approximation this ratio for "good" elements is 2–3 while for $C_0$ continuous pressure it is 6–8. All the elements shown in Fig. 10.5 perform very well, though two (Q4/1 and Q9/4) can on occasion lock when most boundary conditions are on $\mathbf{u}$.

**Example 10.1.** Simple triangle with bubble—MINI element

In Fig. 10.3c we indicate that the simple triangle with $C_0$ linear interpolation and an added bubble for the displacements $\mathbf{u}$ together with continuous $C_0$ linear interpolation for the pressure $p$ satisfied the count test part of the mixed patch test and, verifying the

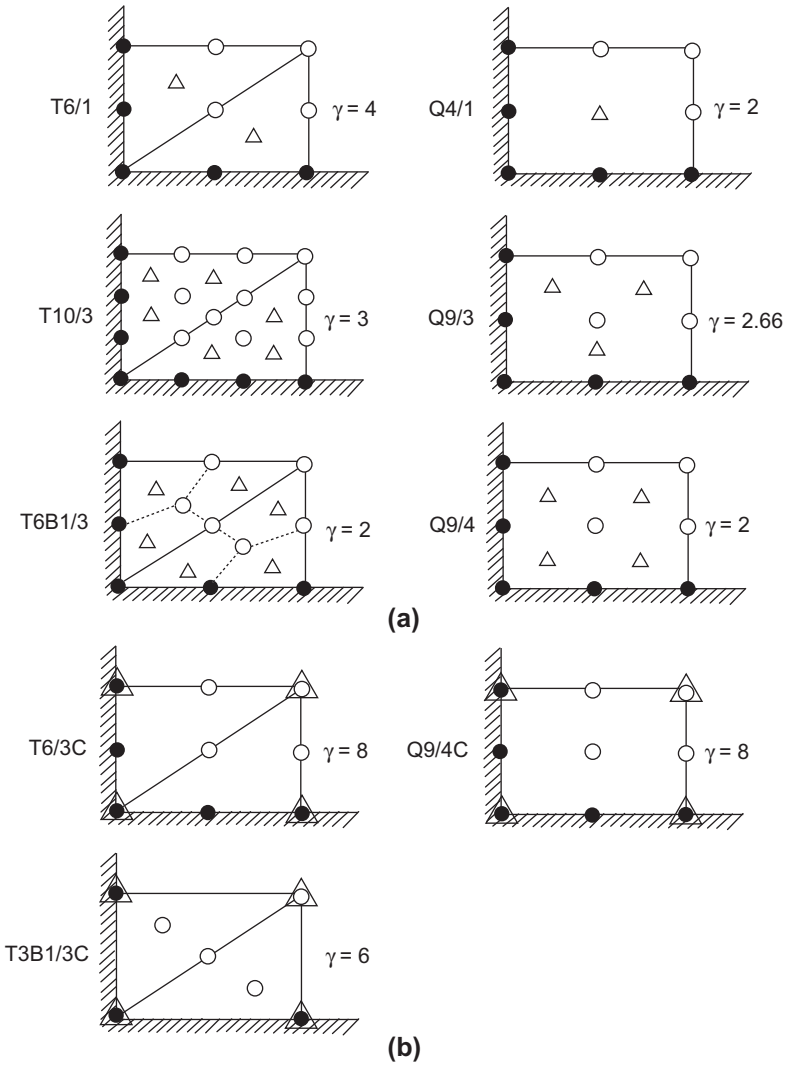



**FIGURE 10.5**

The *freedom index* or *infinite patch ratio* for various $\mathbf{u}$-$p$ elements for incompressible elasticity ($\gamma = n_u/n_p$): (a) discontinuous pressure and (b) continuous pressure.

consistency condition, can be used with success [6]. Here we consider this element further to develop some understanding about its performance at the incompressible limit.

The displacement field with the bubble is written in hierarchical form as

$$\mathbf{u} \approx \hat{\mathbf{u}} = \sum_a N_a \tilde{\mathbf{u}}_a + N_h \tilde{\mathbf{u}}_h \tag{10.13}$$

where here

$$N_h = L_1 L_2 L_3 \tag{10.14}$$

$\tilde{\mathbf{u}}_a$ are nodal parameters of displacement and $\tilde{\mathbf{u}}_h$ are parameters of the hierarchical bubble function. The pressures are similarly given by

$$p \approx \hat{p} = \sum_a N_a \tilde{p}_a \tag{10.15}$$

where $\tilde{p}_a$ are nodal parameters of the pressure. In the above the shape functions are given by (e.g., see Section 6.2.1)

$$N_a = L_a = \frac{1}{2\Delta}\left(a_a + b_a x + c_a y\right) \tag{10.16}$$

where

$$a_a = x_b y_c - x_c y_b, \quad b_a = y_b - y_c, \quad c_a = x_c - y_b$$

$b, c$ are cyclic permutations of $a$, and

$$2\Delta = \det \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} = a_1 + a_2 + a_3$$

The derivatives of the shape functions are thus given by

$$\frac{\partial N_a}{\partial x} = \frac{b_a}{2\Delta} \quad \text{and} \quad \frac{\partial N_a}{\partial y} = \frac{c_a}{2\Delta}$$

Similarly the derivatives of the bubble are given by

$$\begin{aligned} \frac{\partial N_h}{\partial x} &= \frac{1}{2\Delta}\left(b_1 L_2 L_3 + b_2 L_3 L_1 + b_3 L_1 L_2\right) \\ \frac{\partial N_h}{\partial y} &= \frac{1}{2\Delta}\left(c_1 L_2 L_3 + c_2 L_3 L_1 + c_3 L_1 L_2\right) \end{aligned}$$

The strains may be expressed in terms of the above and the nodal parameters as[3]

$$\boldsymbol{\epsilon} = \sum_a \frac{1}{2\Delta} \begin{bmatrix} b_a & 0 \\ 0 & c_a \\ c_a & b_a \end{bmatrix} \tilde{\mathbf{u}}_a + \sum_a \frac{L_b L_c}{2\Delta} \begin{bmatrix} b_a & 0 \\ 0 & c_a \\ c_a & b_a \end{bmatrix} \tilde{\mathbf{u}}_h \tag{10.17}$$

where again $b, c$ are cyclic permutations of $a$.

[3] At this point it is also possible to consider the term added to the derivatives to be *enhanced modes* and delete the bubble mode from displacement terms.

Substituting the above strains into Eq. (10.11) and evaluating the integrals give

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \mathbf{A}_{13} & \mathbf{0} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & \mathbf{A}_{23} & \mathbf{0} \\ \mathbf{A}_{31} & \mathbf{A}_{32} & \mathbf{A}_{33} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{A}_{hh} \end{bmatrix} \tag{10.18}$$

where

$$\mathbf{A}_{ab} = \frac{G}{6\Delta} \begin{bmatrix} (4b_a b_b + 3c_a c_b) & (3c_a b_b - 2b_a c_b) \\ (3b_a c_b - 2c_a b_b) & (3b_a b_b + 4c_a c_b) \end{bmatrix}$$

$$\mathbf{A}_{hh} = \frac{G}{2160\Delta} \begin{bmatrix} (4\mathbf{b}^{\mathrm{T}}\mathbf{b} + 3\mathbf{c}^{\mathrm{T}}\mathbf{c}) & \mathbf{b}^{\mathrm{T}}\mathbf{c} \\ \mathbf{b}^{\mathrm{T}}\mathbf{c} & (3\mathbf{b}^{\mathrm{T}}\mathbf{b} + 4\mathbf{c}^{\mathrm{T}}\mathbf{c}) \end{bmatrix}$$

and

$$\mathbf{b} = \begin{bmatrix} b_1, & b_2, & b_3 \end{bmatrix}^{\mathrm{T}} \quad \text{and} \quad \mathbf{c} = \begin{bmatrix} c_1, & c_2, & c_3 \end{bmatrix}^{\mathrm{T}}$$

Note in the above that all terms except $\mathbf{A}_{hh}$ are standard displacement stiffnesses for the deviatoric part. Similarly,

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} & \mathbf{C}_{13} \\ \mathbf{C}_{21} & \mathbf{C}_{22} & \mathbf{C}_{23} \\ \mathbf{C}_{31} & \mathbf{C}_{32} & \mathbf{C}_{33} \\ \mathbf{C}_{h1} & \mathbf{C}_{h2} & \mathbf{C}_{h3} \end{bmatrix} \tag{10.19}$$

where

$$\mathbf{C}_{ab} = \frac{1}{6} \begin{bmatrix} b_b \\ c_b \end{bmatrix} \quad \text{and} \quad \mathbf{C}_{hb} = -\frac{1}{120} \begin{bmatrix} b_b \\ c_b \end{bmatrix}$$

In all the above arrays $a$ and $b$ have values from 1 to 3 and $h$ denotes the hierarchic bubble mode.

We note that the bubble mode is decoupled from the other entries in the $\mathbf{A}$ array—it is precisely for this reason that the discontinuous constant pressure case shown in Fig. 10.3b cannot be improved by the addition of the internal parameters associated with $\tilde{\mathbf{u}}_h$. Also, the parameters $\tilde{\mathbf{u}}_h$ are defined separately for each element. Consequently, we may perform a partial solution at the element level [13] to obtain the set of equations in the form Eq. (10.10) where now

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \mathbf{A}_{13} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & \mathbf{A}_{23} \\ \mathbf{A}_{31} & \mathbf{A}_{32} & \mathbf{A}_{33} \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} & \mathbf{C}_{13} \\ \mathbf{C}_{21} & \mathbf{C}_{22} & \mathbf{C}_{23} \\ \mathbf{C}_{31} & \mathbf{C}_{32} & \mathbf{C}_{33} \end{bmatrix}, \quad \mathbf{V} = \begin{bmatrix} V_{11} & V_{12} & V_{13} \\ V_{21} & V_{22} & V_{23} \\ V_{31} & V_{32} & V_{33} \end{bmatrix}$$

with

$$V_{ab} = \begin{bmatrix} \dfrac{b_a}{2\Delta} & \dfrac{c_a}{2\Delta} \end{bmatrix} \begin{bmatrix} \tau_{11} & \tau_{12} \\ \tau_{21} & \tau_{22} \end{bmatrix} \begin{Bmatrix} \dfrac{b_b}{2\Delta} \\ \dfrac{c_b}{2\Delta} \end{Bmatrix} \Delta \tag{10.20a}$$

and

$$\boldsymbol{\tau} = \frac{3\Delta^2}{10Gd} \begin{bmatrix} (3\mathbf{b}^{\mathrm{T}}\mathbf{b} + 4\mathbf{c}^{\mathrm{T}}\mathbf{c}) & -\mathbf{b}^{\mathrm{T}}\mathbf{c} \\ -\mathbf{b}^{\mathrm{T}}\mathbf{c} & (4\mathbf{b}^{\mathrm{T}}\mathbf{b} + 3\mathbf{c}^{\mathrm{T}}\mathbf{c}) \end{bmatrix} \tag{10.20b}$$

in which

$$d = 12\left(\mathbf{b}^{\mathrm{T}}\mathbf{b}\right)^2 + 25\left(\mathbf{b}^{\mathrm{T}}\mathbf{b}\right)\left(\mathbf{c}^{\mathrm{T}}\mathbf{c}\right) + 12\left(\mathbf{c}^{\mathrm{T}}\mathbf{c}\right)^2 - \left(\mathbf{b}^{\mathrm{T}}\mathbf{c}\right)^2$$

Now all parameters are at the three nodes of the basic triangle. The reader may recognize the $\mathbf{V}$ array given above as that for the two-dimensional, steady heat equation with conductivity $\mathbf{k} = \boldsymbol{\tau}$ and discretized by linear triangular elements. The direct reduction of the bubble matrix $\mathbf{A}_{hh}$ as given above leads to a full matrix $\boldsymbol{\tau}$. Some numerical experiments including the above formulation are presented in Section 10.7.

## 10.4 Three-field nearly incompressible elasticity (u-$p$ -$\epsilon_v$ form)

A direct approximation of the three-field form leads to an important method in finite element solution procedures for nearly incompressible materials which has sometimes been called the **B**-bar method. The methodology can be illustrated for the nearly incompressible isotropic problem. For this problem the method often reduces to the same two-field form previously discussed. However, for more general anisotropic or inelastic materials and in finite deformation problems the method has distinct advantages as are discussed in Ref. [2]. The usual irreducible form (displacement method) has been shown to "lock" for the nearly incompressible problem. As shown in Section 10.3, the use of a two-field mixed method can avoid this locking phenomenon when properly implemented (e.g., using the Q9/3 two-field form). Below we present an alternative which leads to an efficient and accurate implementation in many situations. For the development shown we shall assume that the material is isotropic linear elastic but it may be extended easily to include anisotropic materials.

Assuming an independent approximation to $\varepsilon_v$ and $p$ we can formulate the problem by use of Eq. (10.7) and the weak statement of relation (10.2) written as

$$\begin{aligned}
&\int_\Omega (\boldsymbol{\mathcal{S}}\delta\mathbf{u})^{\mathrm{T}}\left[\mathbf{D}_d\,\boldsymbol{\mathcal{S}}\mathbf{u} + \mathbf{m}\,p\right]\mathrm{d}\Omega - \int_\Omega \delta\mathbf{u}^{\mathrm{T}}\mathbf{b}\,\mathrm{d}\Omega - \int_\Omega \delta\mathbf{u}^{\mathrm{T}}\bar{\mathbf{t}}\,\mathrm{d}\Gamma = 0\\
&\int_\Omega \delta p\left[\mathbf{m}^{\mathrm{T}}\boldsymbol{\mathcal{S}}\mathbf{u} - \varepsilon_v\right]\mathrm{d}\Omega = 0\\
&\int_\Omega \delta\varepsilon_v\left[K\varepsilon_v - p\right]\mathrm{d}\Omega = 0
\end{aligned} \tag{10.21}$$

If we approximate the $\mathbf{u}$ and $p$ fields by Eq. (10.9) and

$$\varepsilon_v \approx \hat{\varepsilon}_v = \mathbf{N}_v\tilde{\boldsymbol{\epsilon}}_v \tag{10.22}$$

we obtain a mixed approximation in the form of Section 9.5.3 but now only for $p$ and $\varepsilon_v$:

$$\begin{bmatrix} \mathbf{A} & \mathbf{C} & \mathbf{0} \\ \mathbf{C}^{\mathrm{T}} & \mathbf{0} & -\mathbf{E} \\ \mathbf{0} & -\mathbf{E}^{\mathrm{T}} & \mathbf{H} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}} \\ \tilde{\mathbf{p}} \\ \tilde{\boldsymbol{\epsilon}}_v \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \end{Bmatrix} \tag{10.23}$$

where $\mathbf{A}$, $\mathbf{C}$, $\mathbf{f}_1$, $\mathbf{f}_2$ are given by Eq. (10.11) and

$$\mathbf{H} = \int_\Omega \mathbf{N}_v^{\mathrm{T}} K \mathbf{N}_v \, \mathrm{d}\Omega, \quad \mathbf{E} = \int_\Omega \mathbf{N}_v^{\mathrm{T}} \mathbf{N}_p \, \mathrm{d}\Omega, \quad \mathbf{f}_3 = \mathbf{0} \tag{10.24}$$

For completeness we give the variational theorem whose first variation gives (10.21). First we define the strain deduced from the standard displacement approximation as

$$\boldsymbol{\epsilon}_u = \mathcal{S}\mathbf{u} \tag{10.25}$$

The variational theorem is then given as

$$\begin{aligned} \Pi = \frac{1}{2}\int_\Omega \left(\boldsymbol{\epsilon}_u^{\mathrm{T}} \mathbf{D}_d \boldsymbol{\epsilon}_u + \varepsilon_v K \varepsilon_v\right) \mathrm{d}\Omega + \int_\Omega p\left(\mathbf{m}^{\mathrm{T}}\boldsymbol{\epsilon}_u - \varepsilon_v\right) \mathrm{d}\Omega \\ - \int_\Omega \mathbf{u}^{\mathrm{T}}\mathbf{b}\, \mathrm{d}\Omega - \int_{\Gamma_t} \mathbf{u}^{\mathrm{T}}\bar{\mathbf{t}}\, \mathrm{d}\Gamma \end{aligned} \tag{10.26}$$

The reader can now verify that the first variation gives precisely Eqs. (10.21).

### 10.4.1 The B-bar method for nearly incompressible problems

The second row of the matrix Eq. (10.23) has the solution

$$\tilde{\boldsymbol{\epsilon}}_v = \mathbf{E}^{-1}\mathbf{C}^{\mathrm{T}}\tilde{\mathbf{u}} = \mathbf{W}\tilde{\mathbf{u}} \tag{10.27}$$

In the above we assume that $\mathbf{E}$ may be inverted, which implies that $\mathbf{N}_v$ and $\mathbf{N}_p$ have the same number of terms. Furthermore, the approximations for the volumetric strain and pressure are constructed for each element individually and are not continuous across element boundaries. Thus, the solution of Eq. (10.27) may be performed for each individual element. In practice $\mathbf{N}_v$ is normally assumed identical to $\mathbf{N}_p$ so that $\mathbf{E}$ is symmetric positive definite. The solution of the third row of the matrix Eq. (10.23) yields the pressure parameters in terms of the volumetric strain parameters and is given by

$$\tilde{\mathbf{p}} = \mathbf{E}^{-\mathrm{T}}\mathbf{H}^{\mathrm{T}}\tilde{\boldsymbol{\epsilon}}_v \tag{10.28}$$

Substitution of (10.27) and (10.28) into the first row of the matrix Eq. (10.23) gives a solution that is in terms of displacements only. Accordingly,

$$\bar{\mathbf{A}}\tilde{\mathbf{u}} = \mathbf{f}_1 \tag{10.29}$$

where for isotropy

$$\begin{aligned} \bar{\mathbf{A}} &= \int_\Omega \mathbf{B}^{\mathrm{T}}\mathbf{D}_d\mathbf{B}\, \mathrm{d}\Omega + \mathbf{W}^{\mathrm{T}}\mathbf{H}\mathbf{W} \\ &= \mathbf{A} + \mathbf{W}^{\mathrm{T}}\mathbf{H}\mathbf{W} \end{aligned} \tag{10.30}$$

The solution of (10.29) yields the nodal parameters for the displacements. Use of (10.27) and (10.28) then gives the approximations for the volumetric strain and pressure in each element.

The result given by (10.30) may be further modified to obtain a form that is similar to the standard displacement method. Accordingly, we write

$$\bar{\mathbf{A}} = \int_{\Omega} \bar{\mathbf{B}}^{\mathrm{T}} \mathbf{D} \bar{\mathbf{B}} \, \mathrm{d}\Omega \tag{10.31}$$

where the strain-displacement matrix is now

$$\bar{\mathbf{B}} = \mathbf{I}_d \mathbf{B} + \frac{1}{3} \mathbf{m} \mathbf{N}_v \mathbf{W} \tag{10.32}$$

For isotropy the modulus matrix is

$$\mathbf{D} = \mathbf{D}_d + K \mathbf{m} \mathbf{m}^{\mathrm{T}} \tag{10.33}$$

We note that the above form is identical to a standard displacement model except that $\mathbf{B}$ is replaced by $\bar{\mathbf{B}}$. The method has been discussed more extensively in Refs. [14–16].

The equivalence of (10.30) and (10.31) can be verified by simple matrix multiplication. Extension to treat general small strain formulations can be simply performed by replacing the isotropic $\mathbf{D}$ matrix by an appropriate form for the general material model. The formulation shown above has been implemented into an element included as part of the program available on the website. The elegance of the method is more fully utilized when considering nonlinear problems, such as plasticity and finite deformation elasticity (see Ref. [2]).

We note that elimination starting with the third equation could be accomplished leading to a $\mathbf{u}$-$p$ two-field form using $K$ as a penalty number. This is convenient for the case where $p$ is continuous but $\varepsilon_v$ remains discontinuous. Such an elimination, however, points out that precisely the same stability criteria operate here as in the two-field approximation discussed earlier.

## 10.5 Reduced and selective integration and its equivalence to penalized mixed problems

In Chapter 6 we mentioned the lowest order numerical integration rules that still preserve the required convergence order for various elements, but at the same time pointed out the possibility of a singularity in the resulting element matrices. In Chapter 8 we again referred to such low order integration rules, introducing the name "reduced integration" for those that did not evaluate the stiffness exactly for simple elements and pointed out some dangers of its indiscriminate use due to resulting instability. Nevertheless, such reduced integration and selective integration (where the low order integration is only applied to certain parts of the matrix) has proved its worth in practice, often yielding much more accurate results than the use of more precise integration rules. This was particularly noticeable in nearly incompressible elasticity (or Stokes fluid flow which is similar) [17–19] and in problems of plate and shell flexure dealt with as a case of a degenerate solid [20,21] (see Chapters 13 and 14 for more information on plate and shell problems).

The success of these procedures derived initially by heuristic arguments proved quite spectacular—though some consider it somewhat verging on immorality to obtain improved results while doing less work! Obviously fuller justification of such processes is required [22]. The main reason for success is associated with the fact that it provides the necessary singularity of the constraint part of the matrix [viz. Eqs. (9.19)–(9.21)] which avoids locking. Such singularity can be deduced from a count of integration points (viz. Section 6.8), but it is simpler to show that there is a complete equivalence between reduced (or selective) integration procedures and the mixed formulation already discussed in Section 10.3. This equivalence was first shown by Malkus and Hughes [23] and later in a general context by Zienkiewicz and Nakazawa [24].

We shall demonstrate this equivalence on the basis of the nearly incompressible elasticity problem for which the mixed weak Galerkin integral statement is given by Eqs. (10.7) and (10.8). It should be noted, however, that equivalence holds only for the discontinuous pressure approximation.

The corresponding irreducible form can be written by satisfying the second of Eq. (10.8) exactly, implying

$$p = K\mathbf{m}^{\mathrm{T}}\boldsymbol{\epsilon} \tag{10.34}$$

and substituting the above into (10.7) as

$$\begin{aligned}\int_{\Omega} \delta\boldsymbol{\epsilon}^{\mathrm{T}} 2G\left(\mathbf{I}_0 - \frac{1}{3}\mathbf{m}\mathbf{m}^{\mathrm{T}}\right)\boldsymbol{\epsilon}\,\mathrm{d}\Omega + \int_{\Omega} \delta\boldsymbol{\epsilon}^{\mathrm{T}}\mathbf{m}K\mathbf{m}^{\mathrm{T}}\boldsymbol{\epsilon}\,\mathrm{d}\Omega \\ - \int_{\Omega} \mathbf{u}^{\mathrm{T}}\mathbf{b}\,\mathrm{d}\Omega - \int_{\Gamma_t} \mathbf{u}^{\mathrm{T}}\bar{\mathbf{t}}\,\mathrm{d}\Gamma = 0\end{aligned} \tag{10.35}$$

On substituting

$$\mathbf{u} \approx \hat{\mathbf{u}} = \mathbf{N}_u\tilde{\mathbf{u}} \quad \text{and} \quad \boldsymbol{\epsilon} \approx \hat{\boldsymbol{\epsilon}} = \mathcal{S}\mathbf{N}_u\tilde{\mathbf{u}} = \mathbf{B}\tilde{\mathbf{u}} \tag{10.36}$$

we have

$$\left(\mathbf{A} + \bar{\mathbf{A}}\right)\tilde{\mathbf{u}} = \mathbf{f}_1 \tag{10.37}$$

where $\mathbf{A}$ and $\mathbf{f}_1$ are exactly as given in Eq. (10.11) and

$$\bar{\mathbf{A}} = \int_{\Omega} \mathbf{B}^{\mathrm{T}}\mathbf{m}K\mathbf{m}^{\mathrm{T}}\mathbf{B}\,\mathrm{d}\Omega \tag{10.38}$$

The solution of Eq. (10.37) for $\tilde{\mathbf{u}}$ allows the pressures to be determined at all points by Eq. (10.34). In particular, if we have used an integration scheme for evaluating (10.38) which samples at points $(\xi_k)$ we can write

$$p(\xi_k) = K\mathbf{m}^{\mathrm{T}}\boldsymbol{\epsilon}(\xi_k) = K\mathbf{m}^{\mathrm{T}}\mathbf{B}(\xi_k)\tilde{\mathbf{u}} = \sum_j N_{p_j}(\xi_k)\tilde{p}_j \tag{10.39}$$

Now if we turn our attention to the penalized mixed form of Eqs. (10.7)–(10.11) we note that the second of Eq. (10.10) is explicitly

$$\int_{\Omega} \mathbf{N}_p^{\mathrm{T}}\left(\mathbf{m}^{\mathrm{T}}\mathbf{B}\tilde{\mathbf{u}} - \frac{1}{K}\mathbf{N}_p\tilde{\mathbf{p}}\right)\mathrm{d}\Omega = \mathbf{0} \tag{10.40}$$

If a numerical integration is applied to the above sampling at the pressure nodes located at coordinate ($\xi_l$), previously defined in Eq. (10.39), we can write for each scalar component of $\mathbf{N}_p$

$$\sum_l N_{p_j}(\xi_l)\left(\mathbf{m}^{\mathrm{T}}\mathbf{B}(\xi_l)\tilde{\mathbf{u}} - \frac{1}{K}\mathbf{N}_p(\xi_l)\tilde{\mathbf{p}}\right)W_l = 0 \tag{10.41}$$

in which the summation is over all integration points ($\xi_l$) and $W_l$ are the appropriate weights and Jacobian determinants. Now as

$$N_{p_j}(\xi_k) = \delta_{jk}$$

if $\xi_l$ is at the pressure node $j$ and zero at other pressure nodes, Eq. (10.41) reduces simply to the requirement that at all pressure nodes

$$\mathbf{m}^{\mathrm{T}}\mathbf{B}(\xi_l)\tilde{\mathbf{u}} = \frac{1}{K}\mathbf{N}_p(\xi_l)\tilde{\mathbf{p}} \tag{10.42}$$

This is precisely the same condition as that given by Eq. (10.39) and the equivalence of the procedures is proved, *provided the integrating scheme used for evaluating* $\bar{\mathbf{A}}$ *gives an identical integral of the mixed form of* Eq. (10.40).

This is true in many cases and for these the reduced integration-mixed equivalence is exact. In all other cases this equivalence exists for a mixed problem in which an inexact rule of integration has been used in evaluating equations such as (10.40).

For curved isoparametric elements the equivalence is in fact inexact, and slightly different results will be obtained using reduced integration and mixed forms. This is illustrated in examples given in Ref. [25].

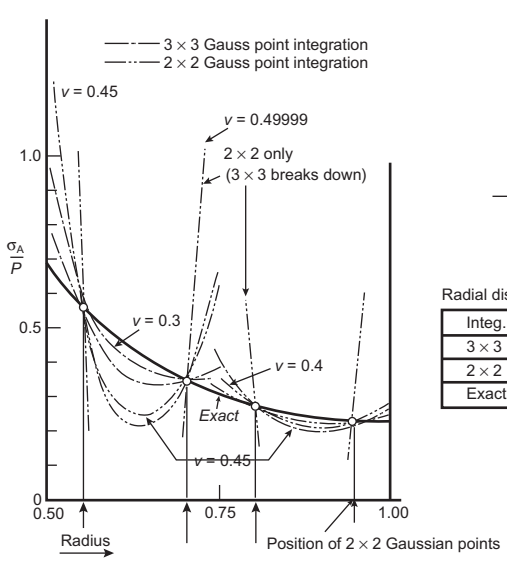
We can conclude without detailed proof that this type of equivalence is quite general and that with any problem of a similar type the application of numerical quadrature at $n_p$ points in evaluating the matrix $\bar{\mathbf{A}}$ within each element is equivalent to a mixed problem in which the variable $p$ is interpolated element-by-element using as $p$ nodal values the same integrating points.

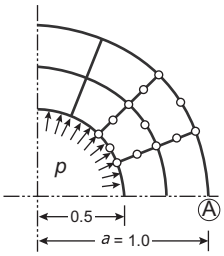
The equivalence is only complete for the selective integration process, i.e., application of reduced numerical quadrature only to the matrix $\bar{\mathbf{A}}$, and ensures that this matrix is singular, i.e., no locking occurs if we have satisfied the previously stated conditions ($n_u > n_p$).

The full use of reduced integration on the remainder of the matrix determining $\tilde{\mathbf{u}}$, i.e., $\mathbf{A}$, is only permissible if that remains nonsingular—the case which we have discussed previously for the Q8/4 element.

It can therefore be concluded that all the elements with discontinuous interpolation of $p$ which we have verified as applicable to the mixed problem (viz. Fig. 10.1, for instance) can be implemented for nearly incompressible situations by a penalized irreducible form using corresponding selective integration.[4]

[4]The Q9/3 element would involve three-point quadrature which is somewhat unnatural for quadrilaterals. It is therefore better to simply use the mixed form here—and, indeed, in any problem which has nonlinear behavior between $p$ and $\mathbf{u}$ (see Ref. [2]).




Radial displacements at Ⓐ *pa*/*E*

| Integ. | $\nu$=0.3 | $\nu$=0.4 | $\nu$=0.45 | $\nu$=0.49999 |
|---|---|---|---|---|
| 3 × 3 | 0.3779 | 0.3904 | 0.3950 | No result |
| 2 × 2 | 0.3776 | 0.3910 | 0.3977 | 0.4041 |
| Exact | 0.3809 | 0.3945 | 0.4013 | 0.4081 |

**FIGURE 10.6**

Sphere under internal pressure. Effect of numerical integration rules on results with different Poisson ratios.

In Fig. 10.6 we show an example which clearly indicates the improvement of displacements achieved by such reduced integration as the compressibility modulus $K$ increases (or the Poisson ratio tends to 0.5). We note also in this example the dramatically improved performance of such points for stress sampling.

For problems in which the $p$ (constraint) variable is continuously interpolated ($C_0$) the arguments given above fail as quantities such as $\mathbf{m}^{\mathrm{T}}\boldsymbol{\epsilon}$ are not interelement continuous in the irreducible form.

A very interesting corollary of the equivalence just proved for (nearly) incompressible behavior is observed if we note the rapid increase of order of integrating formulae with the number of quadrature points (viz. Chapters 3 and 6). *For high order elements the number of quadrature points equivalent to the p constraint permissible for stability rapidly reaches that required for exact integration and hence their performance in nearly incompressible situations is excellent, even if exact integration is used*. This was observed on many occasions [26–28] and Sloan and Randolf [29] have shown good performance with the quintic triangle. Unfortunately such high order elements pose other difficulties and are seldom used in practice.

A final remark concerns the use of "reduced" integration in particular and of penalized, mixed, methods in general. As we pointed out in Section 9.3.1 it is possible in such forms to obtain sensible results for the *primary variable* (**u** in the present example) even though the general stability conditions are violated, provided some of the *constraint equations* are linearly dependent. Now of course the *constraint variable* ($p$ in the present example) is not determinate in the limit.

This situation occurs with some elements that are occasionally used for the solution of incompressible problems but which do not pass our mixed patch test, such as Q8/4 and Q9/4 of Fig. 10.1. If we take the latter number to correspond to the integrating points these will yield acceptable **u** fields, though not good $p$ values.

Figure 10.7 illustrates the point on an application involving slow viscous flow through an orifice—a problem that obeys identical equations to those of incompressible elasticity. Here elements Q8/4, Q8/3, Q9/4, and Q9/3 are compared although only the last completely satisfies the stability requirements of the mixed patch test. All elements are found to give a reasonable velocity (**u**) field but pressures are acceptable only for the last one, with element Q8/4 failing to give results which can be plotted [5].

It is of passing interest to note that a similar situation develops if four triangles of the T3/1 type are assembled to form a quadrilateral in the manner of Fig. 10.8. Although the original element locks, as we have previously demonstrated, a linear dependence of the constraint equation allows the assembly to be used quite effectively in many incompressible situations, as shown in Ref. [30].

**Example 10.2.** A weak patch test—selective integration

In order to illustrate the performance of an element which only satisfies a weak patch test we consider an axisymmetric linear elastic problem modeled by four-node isoparametric elements. The material is assumed isotropic and the finite element stiffness and reaction force matrices are computed using a selective integration method where terms associated with the bulk modulus are evaluated by a single-point Gaussian quadrature, whereas all other terms are computed using a $2 \times 2$ (standard) Gaussian quadrature. It may be readily verified that the stiffness matrix is of proper rank and thus stability of solutions is not an issue. On the other hand, consistency must still be evaluated.

In order to assess the performance of a selective reduced quadrature formulation we consider the patch of elements shown in Fig. 10.9. The patch is not as generally shaped as desirable and is only used to illustrate performance of an element that satisfies a weak patch test. The polynomial solution considered is

$$\begin{aligned} u &= 2r \\ w &= 0 \end{aligned} \tag{10.43}$$

and material constants $E = 1$ and $\nu = 0$ are used in the analysis. The resulting stress field is given by

$$\sigma_r = \sigma_\theta = 2 \tag{10.44}$$

with other components identically zero. The exact solution for the nodal quantities of the mesh shown in Fig. 10.9 are summarized in Table 10.1. Patch tests have been performed for this problem using the selective reduced integration scheme described above and values of $h$ of 0.8, 0.4, 0.2, 0.1, and 0.05. The result for the radial displacement at nodes 2 and 5 (reported to six digits) is given in Table 10.2. All other

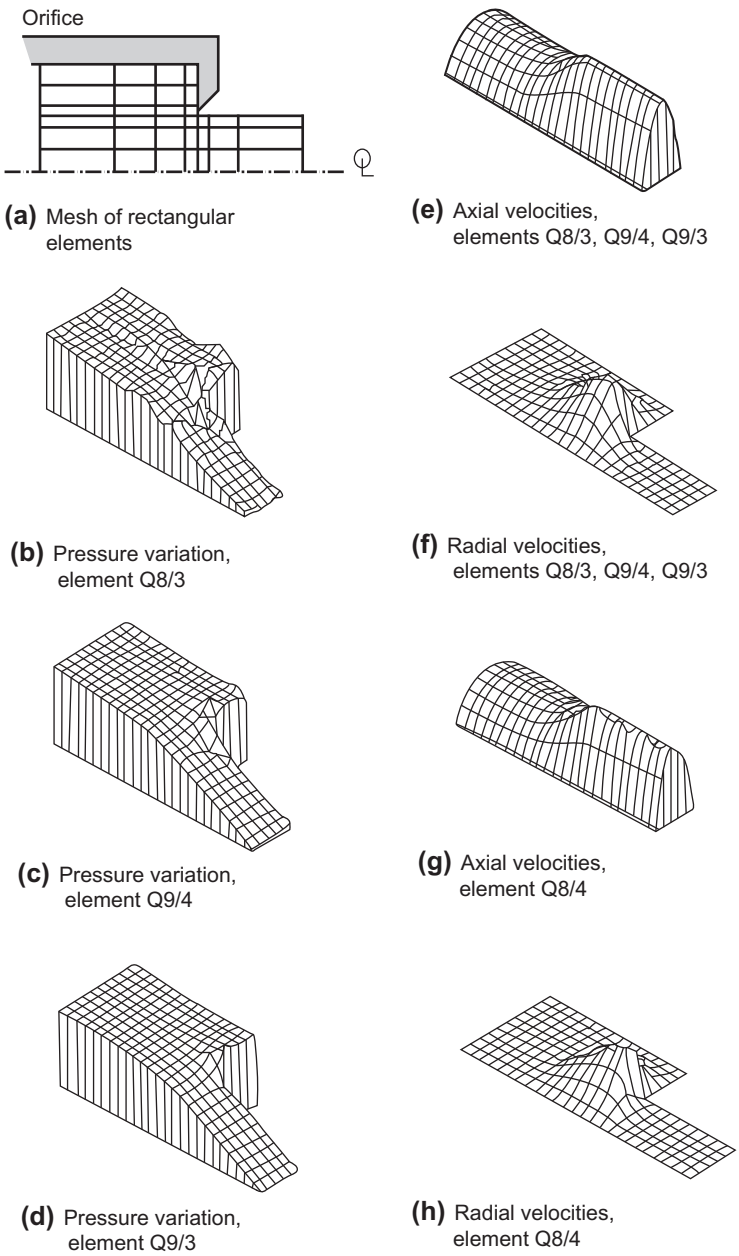



**FIGURE 10.7**

Steady-state, low Reynolds number flow through an orifice. Note that pressure variation for element Q8/4 is so large it cannot be plotted. Solution with $\mathbf{u}/p$ elements Q8/3, Q8/4, Q9/3, Q9/4.

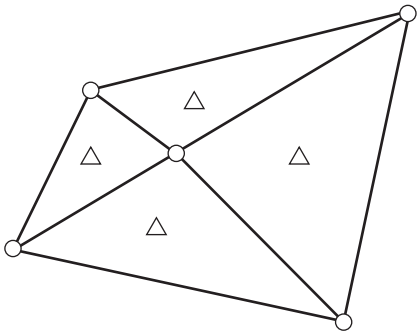


**FIGURE 10.8**
A quadrilateral with intersecting diagonals forming an assembly of four T3/1 elements. This allows displacements to be determined for nearly incompressible behavior but does not yield pressure results.

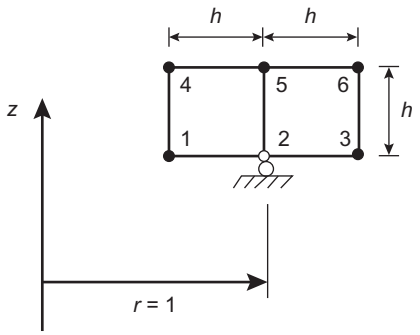



**FIGURE 10.9**
Patch for selective, reduced quadrature on axisymmetric four-node elements.

**Table 10.1** Exact Solution for Patch

| Node | Radius | Displacement | | Force | |
|---|---|---|---|---|---|
| $a$ | $r_a$ | $u_a$ | $w_a$ | $F_{ra}$ | $F_{za}$ |
| 1, 4 | $1-h$ | $2(1-h)$ | 0 | $-(1-h)h$ | 0 |
| 2, 5 | 1 | 2 | 0 | 0 | 0 |
| 3, 6 | $1+h$ | $2(1+h)$ | 0 | $(1+h)h$ | 0 |

quantities (displacements, strains, and stresses) have similar performance with convergence rates of at least $O(h)$ or more. Based on this assessment we conclude the element passes a weak patch test. A similar result will be found for elements which are not rectangular and thus the element produces convergent results.

**Table 10.2** Radial Displacement at Nodes 2 and 5

| *h* | *u* |
|---|---|
| 0.8 | 2.01114 |
| 0.4 | 2.00049 |
| 0.2 | 2.00003 |
| 0.1 | 2.00000 |
| 0.05 | 2.00000 |

## 10.6 A simple iterative solution process for mixed problems: Uzawa method

### 10.6.1 General

In the general remarks on the algebraic solution of mixed problems characterized by equations of the type [viz. Eq. (9.14)]

$$\begin{bmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C}^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \mathbf{x} \\ \mathbf{y} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{Bmatrix} \tag{10.45}$$

we have remarked on the difficulties posed by the zero diagonal and the increased number of unknowns ($n_x + n_y$) as compared with the irreducible form ($n_x$ or $n_y$).

A general iterative form of solution is possible, however, which substantially reduces the cost and avoids pivoting to solve the algebraic equations [31]. In this we solve successively

$$\mathbf{y}^{(k+1)} = \mathbf{y}^{(k)} + \boldsymbol{\rho}\mathbf{r}^{(k)} \tag{10.46}$$

where $\mathbf{r}^{(k)}$ is the residual of the second equation computed as

$$\mathbf{r}^{(k)} = \mathbf{C}^{\mathrm{T}}\mathbf{x}^{(k)} - \mathbf{f}_2 \tag{10.47}$$

and follow with solution of the first equation, i.e.,

$$\mathbf{x}^{(k+1)} = \mathbf{A}^{-1}(\mathbf{f}_1 - \mathbf{C}\mathbf{y}^{(k+1)}) \tag{10.48}$$

In the above $\boldsymbol{\rho}$ is a "convergence accelerator matrix" and is chosen to be efficient and simple to use.

The algorithm is similar to that described initially by Uzawa and co-workers [32] and has been widely applied in an optimization context [27,33–37].

Its relative simplicity can best be grasped when a particular example is considered.

### 10.6.2 Iterative solution for incompressible elasticity

In this case we start from Eq. (10.10) now written with $\mathbf{V} = \mathbf{0}$, i.e., complete incompressibility is assumed. The various matrices are defined in (10.11), resulting in the

form

$$\begin{bmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C}^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}} \\ \tilde{\mathbf{p}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{0} \end{Bmatrix} \tag{10.49}$$

Now, however, for three-dimensional problems the matrix $\mathbf{A}$ is singular (as volumetric changes are not restrained) and it is necessary to *augment* it to make it nonsingular. We can do this in the manner described in Section 9.3.1, or equivalently by the addition of a fictitious compressibility matrix, thus replacing $\mathbf{A}$ by

$$\bar{\mathbf{A}} = \mathbf{A} + \int_{\Omega} \mathbf{B}^{\mathrm{T}} (\lambda G \mathbf{m}\mathbf{m}^{\mathrm{T}}) \mathbf{B} \, \mathrm{d}\Omega \tag{10.50}$$

If the second matrix uses an integration consistent with the number of discontinuous pressure parameters assumed, then this is precisely equivalent to writing

$$\bar{\mathbf{A}} = \mathbf{A} + \lambda G \mathbf{C}\mathbf{C}^{\mathrm{T}} \tag{10.51}$$

and is simpler to evaluate. Clearly this addition does not change the equation system.

The iteration of the algorithm (10.46)–(10.48) is now conveniently taken with the "convergence accelerator" being simply defined as

$$\boldsymbol{\rho} = \lambda G \mathbf{I} \tag{10.52}$$

We now have the iterative system given as

$$\tilde{\mathbf{p}}^{(k+1)} = \tilde{\mathbf{p}}^{(k)} + \lambda G \mathbf{r}^{(k)} \tag{10.53}$$

where

$$\mathbf{r}^{(k)} = \mathbf{C}^{\mathrm{T}} \tilde{\mathbf{u}}^{(k)} \tag{10.54}$$

is the residual of the incompressibility constraint, and

$$\tilde{\mathbf{u}}^{(k+1)} = \bar{\mathbf{A}}^{-1} (\mathbf{f}_1 - \mathbf{C} \tilde{\mathbf{p}}^{(k+1)}) \tag{10.55}$$

In this $\bar{\mathbf{A}}$ can be interpreted as the stiffness matrix of a compressible material with bulk modulus $K = \lambda G$ and the process may be interpreted as the successive addition of volumetric "initial" strains designed to reduce the volumetric strain to zero. Indeed this simple approach led to the first realization of this algorithm [38–40]. Alternatively the process can be visualized as an amendment of the original equation (10.49) by subtracting the term $\mathbf{p}/(\lambda G)$ from each side of the second to give (this is often called an *augmented Lagrangian form*) [31,36,37]

$$\begin{bmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C}^{\mathrm{T}} & -\frac{1}{\lambda G}\mathbf{I} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}} \\ \tilde{\mathbf{p}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ -\frac{1}{\lambda G}\tilde{\mathbf{p}} \end{Bmatrix} \tag{10.56}$$

and adopting the iteration

$$\begin{bmatrix} \bar{\mathbf{A}} & \mathbf{C} \\ \mathbf{C}^{\mathrm{T}} & -\frac{1}{\lambda G}\mathbf{I} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}} \\ \tilde{\mathbf{p}} \end{Bmatrix}^{(k+1)} = \begin{Bmatrix} \mathbf{f}_1 \\ -\frac{1}{\lambda G}\tilde{\mathbf{p}}^{(k)} \end{Bmatrix} \tag{10.57}$$

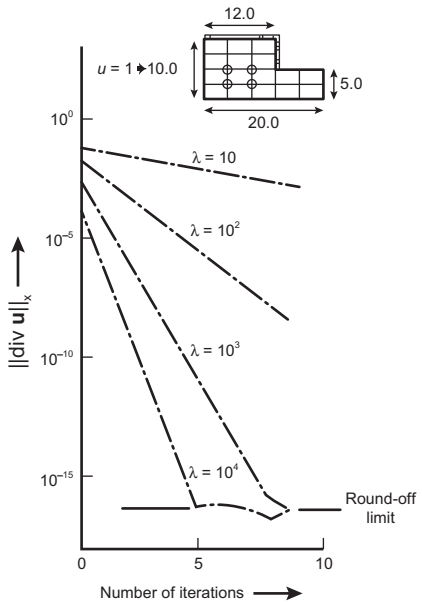



**FIGURE 10.10**
Convergence of iterations in an extrusion problem for different values of the parameter $\lambda$.

With this, on elimination, a sequence similar to Eqs. (10.53)–(10.55) will be obtained provided $\bar{\mathbf{A}}$ is defined by Eq. (10.51).

Starting the iteration from

$$\tilde{\mathbf{u}}^{(0)} = \mathbf{0} \quad \text{and} \quad \tilde{\mathbf{p}}^{(0)} = \mathbf{0}$$

in Fig. 10.10 we show the convergence of the maximum $\varepsilon_v = \operatorname{div} \mathbf{u}$ computed at any of the integrating points used. We note that this convergence becomes quite rapid for large values of $\lambda = (10^3\text{–}10^4)$.

With $\lambda = 10^4$, for instance, in five iterations the initial div $\mathbf{u}$ is reduced from the value $\sim 10^{-4}$ to $10^{-16}$, which is at the round-off limit of the particular computer used.

Finally, we remind the reader that the above iterative process solves the equations of a mixed problem. Accordingly, it is fully effective only when the element used satisfies the stability and consistency conditions of the mixed patch test.

## 10.7 Stabilized methods for some mixed elements failing the incompressibility patch test

It has been observed earlier in this chapter that many of the two-field $\mathbf{u}$-$p$ elements do not pass the stability conditions imposed by the mixed patch test at the incompressible limit (or the Babuška-Brezzi conditions). Here in particular we have such

methods in which the displacement and pressure are interpolated in an identical manner (for instance, linear triangles, linear quadrilaterals, quadratic triangles, etc.) and many attempts for *stabilization* of such elements have been introduced. Indeed one may view the bubble introduced in Example 10.1 as a stabilized method. However, several alternative categories to these exist. The first category is the introduction of nonzero diagonal terms of the constraint equation by adding a least-squares form to the Galerkin formulation. This was first suggested by Courant [41] as a means of improving accuracy in solutions. It appears that Brezzi and Pitkäranta [42] were the first to add terms to the Galerkin solution in an attempt to stabilize results. Numerous further suggestions have been proposed by Hughes et al. with the final form again a least squares approach called the *Galerkin least squares method* [43–45]. An alternative proposal of achieving similar answers has been proposed by Oñate [46] which gains the addition of diagonal terms by the introduction of so-called *finite increment calculus* to the formulation. A very simple stabilization also has been proposed by Dohrmann and Bochev [47] in which a stabilization involving the difference between the interpolated pressure and a direct projection of pressure is appended to the Galerkin equations in a least-squares form.

There is, however, an alternative possibility introduced by time integration of the full incompressible formulation. Here many of the algorithms will yield, when steady-state conditions are recovered, a stabilized form. A number of such algorithms have been discussed by Zienkiewicz and Wu [48] and a very efficient method has appeared as a by-product of a fluid mechanics algorithm named the *characteristic-based split* (CBS) procedure [49–53] (which is discussed at length in Ref. [1]).

In the latter algorithm there exists a free parameter. This parameter depends on the size of the time increment. In the other methods there is a weighting parameter applied to the additional terms introduced. We shall discuss each of these algorithms in the following subsections and compare the numerical results obtainable.

One may question, perhaps, that resort to stabilization procedures is not worthwhile in view of the relative simplicity of the full mixed form. But this is a matter practice will decide and is clearly in the hands of the analyst applying the necessary solutions.

### 10.7.1 Laplacian pressure stabilization

In the first part of this chapter we separated the stress into the deviatoric and pressure components as

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^d + \mathbf{m}p$$

Using the tensor form described in Appendix B this may be written in index form as

$$\sigma_{ij} = \sigma_{ij}^d + \delta_{ij}\, p$$

The deviatoric stresses are related to the deviatoric strains through the relation

$$\sigma_{ij}^d = 2G\varepsilon_{ij}^d = G\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial u_k}{\partial x_k}\right) \tag{10.58}$$

The equilibrium equations (in the absence of inertial forces) are

$$\frac{\partial \sigma_{ij}^{d}}{\partial x_i} + \frac{\partial p}{\partial x_j} + b_j = 0$$

Substituting the constitutive equations for the deviatoric part yields the equilibrium form (assuming $G$ is constant)

$$G\left[\frac{\partial^2 u_j}{\partial x_i\, \partial x_i} + \frac{1}{3}\frac{\partial^2 u_i}{\partial x_i\, \partial x_j}\right] + \frac{\partial p}{\partial x_j} + b_j = 0 \tag{10.59}$$

In vector form this is given as

$$G[\nabla^2 \mathbf{u} + \frac{1}{3}\nabla(\operatorname{div}\mathbf{u})] + \nabla p + \mathbf{b} = \mathbf{0}$$

where $\nabla^2$ is the Laplacian operator and $\nabla$ the gradient operator. The constitutive equation (10.2) is expressed in terms of the displacement as

$$\varepsilon_v = \frac{\partial u_i}{\partial x_i} = \operatorname{div}\mathbf{u} = \frac{1}{K}p \tag{10.60}$$

where $\operatorname{div}(\cdot)$ is the divergence of the quantity. A single equation for pressure may be deduced from the divergence of the equilibrium equation. Accordingly, from Eq. (10.59) we obtain

$$\frac{4G}{3}\nabla^2\left(\operatorname{div}\mathbf{u}\right) + \nabla^2 p + \operatorname{div}\mathbf{b} = 0 \tag{10.61}$$

where upon noting (10.60) we obtain finally

$$\left(1 + \frac{4G}{3K}\right)\nabla^2 p + \operatorname{div}\mathbf{b} = 0 \tag{10.62}$$

Thus, in general, the pressure must satisfy a Poisson equation, or in the absence of body forces, a Laplace equation.

We have noted the dangers of artificially raising the order of the differential equation in introducing spurious solutions. However, in the context of constructing approximate solutions to the incompressible problem the above is useful in providing additional terms to the weak form which otherwise would be zero. Brezzi and Pitkäranta [42] suggested adding Eq. (10.62) to Eq. (10.8) and (on setting the body force to zero for simplicity) obtain

$$\int_{\Omega} \delta p\left(\mathbf{m}^{\mathrm{T}}\boldsymbol{\epsilon} - \frac{1}{K}p\right)\mathrm{d}\Omega + \sum_e \beta \int_{\Omega_e} \delta p \nabla^2 p\, \mathrm{d}\Omega = 0 \tag{10.63}$$

where $\beta$ is a parameter introduced to control accuracy. The last term may be integrated by parts to yield a form which is more amenable to computation as

$$\int_\Omega \delta p\left(\mathbf{m}^{\mathrm{T}}\boldsymbol{\epsilon} - \frac{1}{K}p\right)\mathrm{d}\Omega - \sum_e \beta \int_{\Omega_e} \frac{\partial \delta p}{\partial x_i}\frac{\partial p}{\partial x_i}\,\mathrm{d}\Omega = 0 \tag{10.64}$$

in which the resulting boundary terms are ignored. Upon discretization using equal order linear interpolation on triangles for $\mathbf{u}$ and $p$ we obtain a form identical to that for the bubble in Example 10.1 with the exception that $\boldsymbol{\tau}$ (viz. Eq. [10.20b]) is now given by

$$\boldsymbol{\tau} = \beta\mathbf{I} \tag{10.65}$$

On dimensional considerations with the first term in Eq. (10.64) the parameter $\beta$ should have a value proportional to $L^4/F$, where $L$ is length and $F$ is force. We defer discussion on the particular value until after presenting the Galerkin least squares method.

### 10.7.2 Galerkin least squares method

The Galerkin least squares (GLS) approach is a general scheme for solving the differential equations (9.1) by a finite element method. We may write the GLS form as

$$\int_\Omega \delta\mathbf{u}^{\mathrm{T}}\boldsymbol{\mathcal{A}}(\mathbf{u})\mathrm{d}\Omega + \sum_e \int_{\Omega_e} \delta\boldsymbol{\mathcal{A}}(\mathbf{u})^{\mathrm{T}}\boldsymbol{\tau}\boldsymbol{\mathcal{A}}(\mathbf{u})\mathrm{d}\Omega = 0 \tag{10.66}$$

where the first term represents the normal Galerkin form and the added terms are computed for each element individually including a weight $\boldsymbol{\tau}$ to provide dimensional balance and scaling. Generally, the $\boldsymbol{\tau}$ will involve parameters which have to be selected for good performance. Discontinuous terms on boundaries between elements that arise from higher order terms in $\mathcal{A}(\mathbf{u})$ are commonly omitted.

The form given above has been used by Hughes et al. [45] as a means of stabilizing the fluid flow equations, which for the case of the incompressible Stokes problem coincide with those for incompressible linear elasticity. For this problem only the momentum equation is used in the least squares terms. After substituting Eq. (10.60) into Eq. (10.59) the momentum equation may be written as (assuming that $G$ and $K$ are constant in each element)

$$G\frac{\partial^2 u_j}{\partial x_i^2} + \left(1 + \frac{G}{3K}\right)\frac{\partial p}{\partial x_j} = 0 \tag{10.67}$$

A more convenient form results by using a single parameter defined as

$$\overline{G} = \frac{G}{1 + G/3K} \tag{10.68}$$

With this form the least squares term to be appended to each element may be written as

$$\int_{\Omega_e} \left( \overline{G} \frac{\partial^2 \delta u_i}{\partial x_k^2} + \frac{\partial \delta p}{\partial x_i} \right) \tau_{ij} \left( \overline{G} \frac{\partial^2 u_j}{\partial x_m^2} + \frac{\partial p}{\partial x_j} \right) \mathrm{d}\Omega \tag{10.69}$$

This leads to terms to be added to the standard Galerkin equations and is expressed as

$$\begin{bmatrix} \mathbf{A}^s & \mathbf{C}^s \\ \mathbf{C}^{s\mathrm{T}} & \mathbf{V}^s \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}} \\ \tilde{\mathbf{p}} \end{Bmatrix}$$

where

$$\begin{aligned} \mathbf{A}^s_{ab} &= \int_{\Omega_e} \overline{G}^2 \nabla^2 N_a \boldsymbol{\tau} \nabla^2 N_b \, \mathrm{d}\Omega \\ \mathbf{C}^s_{ab} &= \int_{\Omega_e} \overline{G} \nabla^2 N_a \boldsymbol{\tau} \boldsymbol{\nabla} N_b \, \mathrm{d}\Omega \\ V^s_{ab} &= \int_{\Omega_e} \left( \boldsymbol{\nabla} N_a \right)^{\mathrm{T}} \boldsymbol{\tau} \boldsymbol{\nabla} N_b \, \mathrm{d}\Omega \end{aligned}$$

and the operators on the shape functions are given in two dimensions by

$$\nabla^2 N_a = \frac{\partial^2 N_a}{\partial x_1^2} + \frac{\partial^2 N_a}{\partial x_2^2} \quad \text{and} \quad \boldsymbol{\nabla} N_a = \begin{bmatrix} \dfrac{\partial N_a}{\partial x_1} & \dfrac{\partial N_a}{\partial x_2} \end{bmatrix}^{\mathrm{T}}$$

Note again that all *infinite* terms between elements are ignored (i.e., those arising from second derivatives when $C_0$ functions are used).

For linear triangular elements the second derivatives of the shape functions are identically zero within the element and only the $\mathbf{V}$ term remains and is now nearly identical to the Brezzi-Pitkäranta form if $\beta$ coincides with the definition of $\boldsymbol{\tau}$. In the work of Hughes et al., $\boldsymbol{\tau}$ is given by

$$\boldsymbol{\tau} = -\frac{\alpha h^2}{2G} \mathbf{I} \tag{10.70}$$

where $\alpha$ is a parameter which is recommended to be of $O(1)$ for linear triangles and quadrilaterals.

### 10.7.3 Direct pressure stabilization

In the previous two sections we have discussed the procedures which needed certain disregard for consistency to be introduced. In particular, in both methodologies certain integrals were allowed over the individual elements with high order derivatives and interelement values being omitted especially if these reached infinity, as happens for instance in the GLS method when second derivatives on the interface between elements or boundary terms in the Brezzi-Pitkäranta method are ignored.

In this section we introduce another process proposed by Dohrmann and Bochev [47] which seems to be totally correct and is arrived at without ignoring any terms in the overall integrals. In this procedure we try to ensure that the difference between the $C_0$ interpolated pressures give answers consistent with those in which a lower order, discontinuous approximation is used—i.e., one which is consistent with the general approximation for stresses. Thus, for instance, in triangular elements in which linear displacements are used the stresses are only allowed to be constant within any element and the assumption of any component being also linear is not consistent. For this reason the method looks at the difference between the interpolated pressure which is of the same order as the displacement and its projection onto one order lower expansion consistent with that of the stresses.

The work of Dohrmann and Bochev consider a two-field mixed approximation given by

$$\begin{aligned} &\int_\Omega \delta\boldsymbol{\epsilon}^T \mathbf{D}_d \boldsymbol{\epsilon}\, \mathrm{d}\Omega + \int_\Omega \delta\boldsymbol{\epsilon}^T \mathbf{m}\, p\, \mathrm{d}\Omega - \int_\Omega \delta\mathbf{u}^{\mathrm{T}} \mathbf{b}\, \mathrm{d}\Omega - \int_{\Gamma_t} \delta\mathbf{u}^{\mathrm{T}} \bar{\mathbf{t}}\, \mathrm{d}\Gamma = 0 \\ &\int_\Omega \delta p \left[\mathbf{m}\boldsymbol{\epsilon} - \frac{1}{K} p\right] \mathrm{d}\Omega - \sum_e \int_{\Omega_e} \left(\delta p - \delta \check{p}\right) \frac{\alpha}{G} \left(p - \check{p}\right) \mathrm{d}\Omega = 0 \end{aligned} \tag{10.71}$$

in which the displacements $\mathbf{u}$ and pressure $p$ are approximated by $k$ order continuous polynomial shape functions, $\check{p}$ is a discontinuous projection of $p$ onto a polynomial space of order $k-1$, and $\alpha$ is a parameter to be selected for stability. When $K \to \infty$ the above form represents a stable approximation for the incompressible problem for all order of elements provided $\alpha$ is set at a nominal value. In the examples we use $\alpha = 2$.[5]

We note that the form (10.71) requires no integration by parts in which terms are ignored. Thus, the method has considerable theoretical advantages over the previously discussed stabilization methods.

The pressure stabilization is computed for *each element* individually using

$$\int_{\Omega_e} \delta\check{p}(p - \check{p})\mathrm{d}\Omega = 0 \tag{10.72}$$

and, thus, has low additional cost. Due to this form, the stabilization term may also be written as

$$\int_\Omega \left(\delta p - \delta\check{p}\right) \frac{\alpha}{G} \left(p - \check{p}\right) \mathrm{d}\Omega = \int_\Omega \frac{\alpha}{G} \left(\delta p\, p - \delta\check{p}\, \check{p}\right) \mathrm{d}\Omega \tag{10.73}$$

which is now in the form of a difference of two "mass" type arrays. If we approximate the pressure by

$$p \approx \hat{p} = \sum_a N_a\, \tilde{p}_a = \mathbf{N}\,\tilde{\mathbf{p}}$$

[5] Dohrmann and Bochev [47] use $\alpha = 1$. While this leads to convergence we find this value is somewhat small for our examples.

in which $N_a$ contain the set of polynomials of order $k$ and

$$\check{p} = \sum_b h_b(\mathbf{x})\,\alpha_b = \mathbf{h}(\mathbf{x})\,\boldsymbol{\alpha}$$

where $h_b(\mathbf{x})$ are the polynomials of order $k-1$ the solution to (10.72) is determined from

$$\int_{\Omega_e} \mathbf{h}^{\mathrm{T}}\,\mathbf{h}\,\mathrm{d}\Omega\boldsymbol{\alpha} = \int_{\Omega_e} \mathbf{h}^{\mathrm{T}}\,\mathbf{N}\,\mathrm{d}\Omega\,\tilde{\mathbf{p}}$$
$$\mathbf{H}\,\boldsymbol{\alpha} = \mathbf{G}\,\tilde{\mathbf{p}}$$

Thus, the pressure projection is given by

$$\check{p} = \mathbf{h}(\mathbf{x})\,\mathbf{H}^{-1}\mathbf{G}\,\tilde{\mathbf{p}} \tag{10.74}$$

Using the usual finite element approximation for the displacement

$$\mathbf{u} = \sum_a N_a\,\tilde{\mathbf{u}}_a$$

the stabilized weak form may be written in matrix form as

$$\begin{bmatrix} \mathbf{K}_d & \mathbf{C} \\ \mathbf{C}^{\mathrm{T}} & -\mathbf{V} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}} \\ \tilde{\mathbf{p}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ \mathbf{0} \end{Bmatrix} \tag{10.75}$$

where the arrays are given by

$$\mathbf{K}_d = \int_\Omega \mathbf{B}^{\mathrm{T}}\mathbf{D}_d\mathbf{B}\,\mathrm{d}\Omega, \quad \mathbf{C} = \int_\Omega \mathbf{B}^{\mathrm{T}}\mathbf{m}_d\mathbf{N}\,\mathrm{d}\Omega$$
$$\mathbf{V} = \int_\Omega \mathbf{N}^{\mathrm{T}}\left[\frac{1}{K} + \frac{\alpha}{G}\right]\mathbf{N}\,\mathrm{d}\Omega - \frac{\alpha}{G}\mathbf{G}^{T}\mathbf{H}^{-1}\mathbf{G}$$

and $\mathbf{f}$ is the usual force due to boundary traction and body loads. It is clear from the definition of $\mathbf{V}$ that, when $\mathbf{G}$ is much smaller than $K$, the effect of the direct pressure stabilization is a *penalty* form on the difference of the interpolated and projected pressure. Since for consistency in a patch test, it is only the pressures of order $k-1$ (i.e., the projected pressures) that are required now explains the success of the approach.

**Example 10.3.** Direct stabilization for three-node triangular element

As an example consider the two-dimensional plane strain problem in which the solution is achieved using linear triangles ($k = 1$) with shape functions given by

$$N_a = L_a, \quad a = 1, 2, 3$$

Here the projection for $\check{p}$ is given by a constant ($k = 0$) value

$$\check{p} = \frac{1}{3}\left(\tilde{p}_1 + \tilde{p}_2 + \tilde{p}_3\right)$$

Note that numerical integration of the stabilizing term may not be performed using one-point quadrature at the element baricenter as then no contribution to the stabilizing term would be found.

Performing the integrations for the stabilizing term (10.73) gives the result

$$\mathbf{v}_{\text{stab}} = \frac{\alpha\Delta}{12G}\begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} - \frac{\alpha\Delta}{9G}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \frac{\alpha\Delta}{36G}\begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} \tag{10.76}$$

where $\Delta$ is the area of the triangular element. We recognize this result to have the same form as the deviatoric projection array which is positive semi-definite. The singular nature of this array permits the constant values of $p$ to be unaffected by the stabilizing terms, thus maintaining optimal accuracy for the method.

If we assemble the stabilization array given in (10.76) for the four-element patch shown in Fig. 10.11a we obtain an equation for node 0

$$\frac{\alpha\Delta}{36G}\left[4\,p_0 - \sum_1^4 p_i\right]$$

which we recognize as a Laplacian-type form. Similarly, for the six-element patch shown in Fig. 10.11b we obtain

$$\frac{\alpha\Delta}{36G}\left[4\,p_0 - \sum_1^6 p_i\right]$$

which has a similar form but is not the same as the Laplacian operator for this mesh pattern.

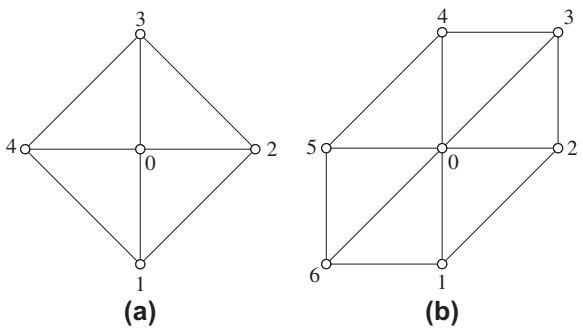



**FIGURE 10.11**
Mesh patterns for pressure stabilization matrix evaluation: (a) four-element patch and (b) six-element patch.

The simplicity of the direct pressure stabilization is one of its main advantages. However, it also permits applications on elements of other order and shape without significant complication. For example, if degenerate element forms for quadratic elements are used as discussed in Chapter 6, the direct approach provides a means of stabilizing computations for incompressible forms without need to add any second derivative terms (as, for example, needed for the GLS form).

We will show later with numerical examples how well the direct stabilization approach works.

### 10.7.4 Incompressibility by time stepping

The fully incompressible case (i.e., $K = \infty$) has been studied by Zienkiewicz and Wu [48] using various time-stepping procedures. Their applications concern the solution of fluid problems in which the rate effects for the Stokes equation appear as first derivatives of time. We can consider such a method here as a procedure to obtain the static solutions of elasticity problems in the limit as the rate terms become zero. Thus, this approach is considered here as a method for either the Stokes equation or the case of static incompressible elasticity.

The governing equations for slightly compressible Stokes flow may be written as

$$\begin{aligned} \rho_0 \frac{\partial u_i}{\partial t} - \frac{\partial \sigma_{ij}^d}{\partial x_j} - \frac{\partial p}{\partial x_i} &= 0 \\ \frac{1}{\rho_0 c^2} \frac{\partial p}{\partial t} - \frac{\partial u_i}{\partial x_i} &= 0 \end{aligned} \tag{10.77}$$

where $\rho_0$ is density (taken as unity in subsequent developments), $c = (K/\rho_0)^{1/2}$ is the speed of compressible waves, $p$ is the pressure (here taken as positive in tension), and $u_i$ is a velocity (or for elasticity interpretations a displacement) in the $i$-coordinate direction. Note that the above form assumes some compressibility in order to introduce the pressure rate term. At the steady limit this term is not involved, consequently, the solution will correspond to the incompressible case. Deviatoric stresses $\sigma_{ij}^d$ are related to deviatoric strains (or strain rates for fluids) as described by Eq. (10.58).

Zienkiewicz and Wu consider many schemes for integrating the above equations in time. Here we introduce only one of the forms, which is widely used in the solution of the fluid equations which include transport effects (see Ref. [1]). For the full fluid equations the algorithm is part of the *characteristic-based split* (CBS) method [1,49,50,52–54].

The equations are discretized in time using the approximations $u(t_n) \approx u^n$ and time derivatives

$$\frac{\partial u_i}{\partial t} \approx \frac{u^{n+1} - u^n}{\Delta t} \tag{10.78}$$

where $\Delta t = t_{n+1} - t_n$. The time discretized equations are given by

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{\partial \sigma_{ij}^{d,n}}{\partial x_j} + \frac{\partial p^n}{\partial x_i} + \theta_2 \frac{\partial \Delta p}{\partial x_i} \tag{10.79a}$$

and

$$\frac{1}{c^2}\frac{p^{n+1}-p^n}{\Delta t}=\frac{\partial u_i^n}{\partial x_i}+\theta_1\frac{\partial \Delta u_i}{\partial x_i} \tag{10.79b}$$

where $\Delta p = p^{n+1}-p^n$, $\Delta u_i = u_i^{n+1}-u_i^n$, $\theta_1$ can vary between 1/2 and 1, and $\theta_2$ can vary between 0 and 1. In all that follows we shall use $\theta_1 = 1$.

The form to be considered uses a split of the equations by defining an intermediate approximate velocity $u_i^\star$ at time $t_{n+1}$ when integrating the equilibrium equation (10.79a). Accordingly, we consider

$$\frac{u_i^\star - u_i^n}{\Delta t}=\frac{\partial \sigma_{ij}^{d,n}}{\partial x_j} \tag{10.80a}$$

and

$$\frac{u_i^{n+1}-u_i^\star}{\Delta t}=\frac{\partial p^n}{\partial x_i}+\theta_2\frac{\partial \Delta p}{\partial x_i} \tag{10.80b}$$

Differentiating the second of these with respect to $x_i$ to get the divergence of $u_i^{n+1}$ and combining with the discrete pressure equation (10.79b) results in

$$\frac{1}{c^2}\frac{\Delta p}{\Delta t}-\theta_2\Delta t\frac{\partial^2 \Delta p}{\partial x_i\partial x_i}=\Delta t\frac{\partial^2 p^n}{\partial x_i\partial x_i}+\frac{\partial u_i^\star}{\partial x_i} \tag{10.80c}$$

Thus, the original problem has been replaced by a set of three equations which need to be solved successively.

Equations (10.8a–c) may be written in a weak form using as weighting functions $\delta\mathbf{u}^\star$, $\delta\mathbf{u}$, and $\delta p$, respectively (viz. Chapter 3). They are then discretized in space using the approximations

$$\begin{aligned}
&\mathbf{u}^n \approx \hat{\mathbf{u}}^n = \mathbf{N}_u\tilde{\mathbf{u}}^n \quad \text{and} \quad \delta\mathbf{u} \approx \delta\hat{\mathbf{u}} = \mathbf{N}_u\delta\tilde{\mathbf{u}}\\
&\mathbf{u}^\star \approx \hat{\mathbf{u}}^\star = \mathbf{N}_u\tilde{\mathbf{u}}^\star \quad \text{and} \quad \delta\mathbf{u}^\star \approx \delta\hat{\mathbf{u}}^\star = \mathbf{N}_u\delta\tilde{\mathbf{u}}^\star\\
&p^n \approx \hat{p}^n = \mathbf{N}_p\tilde{\mathbf{p}}^n \quad \text{and} \quad \delta p \approx \delta\hat{p} = \mathbf{N}_p\delta\tilde{\mathbf{p}}
\end{aligned}$$

with similar expressions for $\mathbf{u}^{n+1}$ and $p^{n+1}$. The final discrete form is given by the three equation sets

$$\begin{aligned}
\frac{1}{\Delta t}\mathbf{M}_u\left(\tilde{\mathbf{u}}^\star-\tilde{\mathbf{u}}^n\right) &= -\mathbf{A}\tilde{\mathbf{u}}^n+\mathbf{f}_1\\
\left[\frac{1}{\Delta t}\mathbf{M}_p+\theta_2\Delta t\mathbf{H}\right]\Delta\tilde{\mathbf{p}} &= -\mathbf{C}\tilde{\mathbf{u}}^\star-\Delta t\mathbf{H}\tilde{\mathbf{p}}^n+\mathbf{f}_2\\
\frac{1}{\Delta t}\mathbf{M}_u\left(\tilde{\mathbf{u}}^{n+1}-\tilde{\mathbf{u}}^\star\right) &= -\mathbf{C}^{\mathrm{T}}\left(\tilde{\mathbf{p}}^n+\theta_2\Delta\tilde{\mathbf{p}}\right)
\end{aligned} \tag{10.81}$$

In the above we have integrated by parts all the terms which involve derivatives on deviator stress ($\sigma_{ij}^d$), pressure ($p$), and displacements (velocities). In addition we

consider only the case where $u_i^{n+1} = u_i^\star = \bar{u}_i$ on the boundary $\Gamma_u$ (thus requiring $\delta u_i = \delta u_i^\star = 0$ on $\Gamma_u$). Accordingly, the matrices are defined as

$$\begin{aligned}
\mathbf{M}_u &= \int_\Omega \mathbf{N}_u^{\mathrm{T}} \mathbf{N}_u \, \mathrm{d}\Omega, \quad \mathbf{M}_p = \int_\Omega \frac{1}{c^2} \mathbf{N}_p^{\mathrm{T}} \mathbf{N}_p \, \mathrm{d}\Omega \\
\mathbf{A} &= \int_\Omega \mathbf{B}^{\mathrm{T}} \mathbf{D}_d \mathbf{B} \, \mathrm{d}\Omega, \quad \mathbf{C} = \int_\Omega \frac{\partial \mathbf{N}_p}{\partial x_i} \mathbf{N}_u \, \mathrm{d}\Omega \\
\mathbf{H} &= \int_\Omega \frac{\partial \mathbf{N}_p^{\mathrm{T}}}{\partial x_i} \frac{\partial \mathbf{N}_p}{\partial x_i} \, \mathrm{d}\Omega, \quad \mathbf{f}_1 = \int_{\Gamma_t} \mathbf{N}_u^{\mathrm{T}} (\bar{\mathbf{t}} - k \, \mathbf{n} p^n) \, \mathrm{d}\Gamma \\
\mathbf{f}_2 &= \int_{\Gamma_u} \mathbf{N}_p^{\mathrm{T}} \mathbf{n}^{\mathrm{T}} \bar{\mathbf{u}} \, \mathrm{d}\Gamma
\end{aligned} \tag{10.82}$$

in which $\mathbf{D}_d$ are the deviatoric moduli defined previously. The parameter $k$ denotes an option on alternative methods to split the boundary traction term and is taken as either zero or unity. We note that a choice of zero simplifies the computation of boundary contributions; however, some would argue that unity is more consistent with the integration by parts.

The boundary pressure acting on $\Gamma_t$ is computed from the specified surface tractions ($\bar{t}_i$) and the "best" estimate for the deviator stress at step $n+1$ which is given by $\sigma_{ij}^{d,\star}$. Accordingly,

$$\bar{p}^{n+1} \approx n_i \bar{t}_i - n_i \sigma_{ij}^{d,\star} n_j$$

is imposed at each node on the boundary $\Gamma_t$.

In general we require that $\Delta t < \Delta t_{\mathrm{crit}}$ where the critical time step is $h^2/2G$ (in which $h$ is the element size). Such a quantity is obviously calculated independently for each element and the lowest value occurring in any element governs the overall stability. It is possible and useful to use here the value of $\Delta t$ calculated for each element separately when calculating incompressible stabilizing terms in the pressure calculation and the overall time step elsewhere (we shall label the time increments multiplying $\mathbf{H}$ in Eq. (10.81)$_3$ as $\Delta t_{\mathrm{int}}$). A ratio of $\gamma = \Delta t_{\mathrm{int}}/\Delta t$ greater than unity improves considerably the stabilizing properties. As the third of Eq. (10.81) has greater stability than the first two equations in (10.81), and for $\theta_2 \geq 1/2$ is unconditionally stable, we recommend that the time step used in this equation be $\gamma \Delta t_{\mathrm{crit}}$ for each node. Generally a value of 2 is good as we shall show in the examples (for additional details see Ref. [52]).

Equation (10.81) defines a value of $\tilde{\mathbf{u}}^\star$ entirely in terms of known quantities at the $n$ step. If the mass matrix $\mathbf{M}_u$ is made diagonal by lumping (see Chapter 12 and Appendix H) the solution is thus trivial. Such an equation is called *explicit*. The equation for $\Delta\tilde{\mathbf{p}}$, on the other hand, depends on both $\mathbf{M}_p$ and $\mathbf{H}$ and it is not possible to make the latter diagonal easily.[6] It is possible to make $\mathbf{M}_p$ diagonal using a similar method as that employed for $\mathbf{M}_u$. Thus, if $\theta_2$ is zero this equation will also be explicit,

[6] It is possible to diagonalize the matrix by solving an eigenproblem as shown in Chapter 14—for large problems this requires more effort than is practical.

otherwise it is necessary to solve a set of algebraic equations and the method for this equation is called *implicit*. Once the value of $\Delta\tilde{\mathbf{p}}$ is known the solution for $\tilde{\mathbf{u}}^{n+1}$ is again explicit. In practice the above process is quite simple to implement, however, it is necessary to satisfy *stability* requirements by limiting the size of the time increment. This was discussed briefly in Chapter 3 and in detail in Ref. [49]. Here we only wish to show the limit result as the changes in time go to zero (i.e., for a constant in time load value) and when full incompressibility is imposed.

At the steady limit the solutions become

$$\tilde{\mathbf{u}}^n = \tilde{\mathbf{u}}^{n+1} = \tilde{\mathbf{u}} \quad \text{and} \quad \tilde{\mathbf{p}}^n = \tilde{\mathbf{p}}^{n+1} = \tilde{\mathbf{p}} \tag{10.83}$$

Eliminating $\mathbf{u}^\star$ the discrete equations reduce to the mixed problem

$$\begin{bmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C}^{\mathrm{T}} & \Delta t \left(\mathbf{C}^{\mathrm{T}}\mathbf{M}_u^{-1}\mathbf{C} - \theta_1 \mathbf{H}\right) \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}} \\ \tilde{\mathbf{p}} \end{Bmatrix} + \begin{Bmatrix} \mathbf{f} \\ \mathbf{0} \end{Bmatrix} = \mathbf{0} \tag{10.84}$$

At the steady limit we again recover a term on the diagonal which stabilizes the solution. This term is again of a Laplace equation type—indeed, it is now the difference between two discrete forms for the Laplace equation. The term $\mathbf{C}^{\mathrm{T}}\mathbf{M}_u^{-1}\mathbf{C}$ makes the bandwidth of the resulting equations larger—thus this form is different from all the previously discussed methods.

### 10.7.5 Numerical comparisons

To provide some insight into the behavior of the above methods we consider two example problems. The first is a problem often used to assess the performance of codes to solve steady-state Stokes flow problems—which is identical to the case for incompressible linear elasticity. The second example is a problem in nearly incompressible linear elasticity.

**Example 10.4.** Driven cavity

A two-dimensional plane (strain) case is considered for a square domain with unit side lengths. The material properties are assumed to be fully incompressible ($\nu = 0.5$) with unit viscosity (elastic shear modulus, $G$, of unity). All boundaries of the domain are restrained in the $x$- and $y$-directions with the top boundary having a unit tangential velocity (displacement) at all nodes except the corner ones. Since the problem is incompressible it is necessary to prescribe the pressure at one point in the mesh—this is selected as the center node along the bottom edge. The $10 \times 10$ element mesh of triangular elements (200 elements total) used for the comparison is shown in Fig. 10.12a. The elements used for the analysis use linear velocity (displacement) and pressure on three-node triangles. Results are presented for the horizontal velocity along the vertical center line AA and for vertical velocity and pressure along the horizontal center line BB. Three forms of stabilization are considered:

1. Galerkin least squares (GLS)/Brezzi-Pitkäranta (BP) where the effect of $\alpha$ on $\boldsymbol{\tau}$ is assessed. The results for the horizontal velocity are given in Fig. 10.12b and for the vertical velocity and pressure in Fig. 10.12c and d, respectively. From the analysis it is assessed that the stabilization parameter $\alpha$ should be about 0.5–1

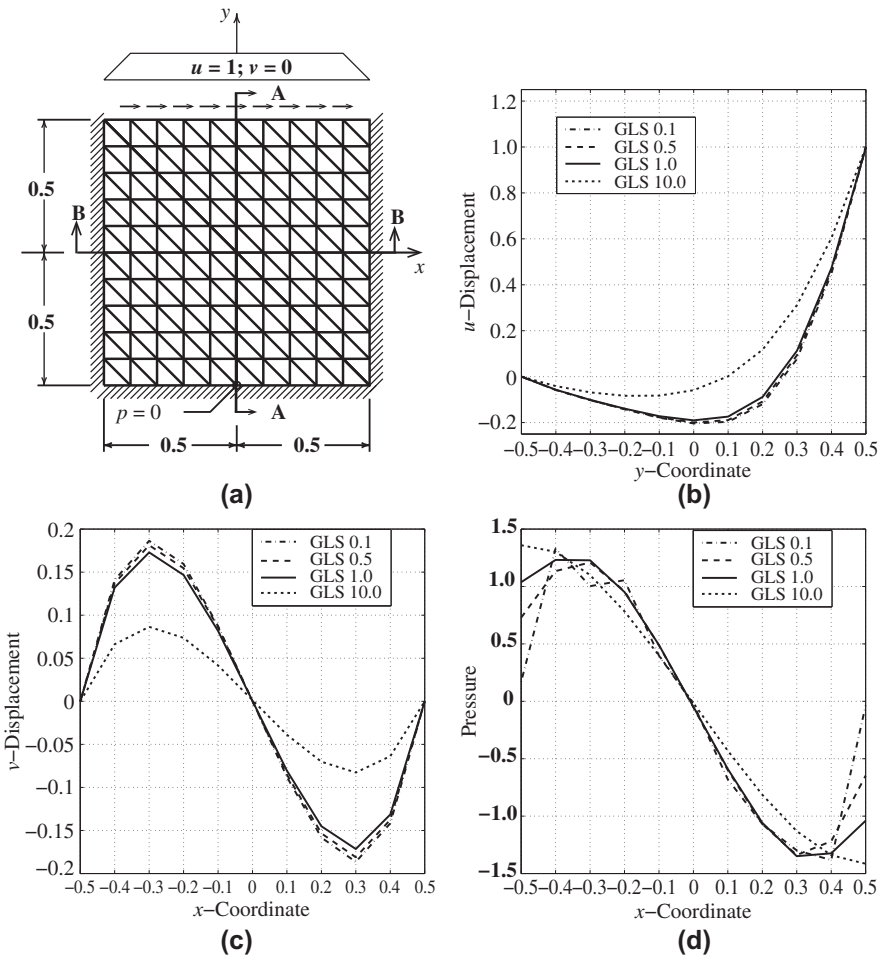



**FIGURE 10.12**

Mesh and GLS/Brezzi-Pitkäranta results: (a) problem definition, (b) horizontal velocity on AA, (c) vertical velocity on BB, and (d) pressure on BB.

(as also indicated by Hughes et al. [45]). Use of lower values leads to excessive oscillation in pressure and use of higher values to strong dissipation of pressure results.

2. Cubic bubble (MINI) element stabilization. Results for vertical velocity are nearly indistinguishable from the GLS results as indicated in Fig. 10.13; however, those for pressure show oscillation. Such oscillation has also been observed by others along with some suggested boundary modifications [55]. No free parameters exist for this element (except possible modification of the bubble mode used); thus, no artificial "tuning" is possible. Use of more refined meshes leads to a strong decrease in the oscillation.

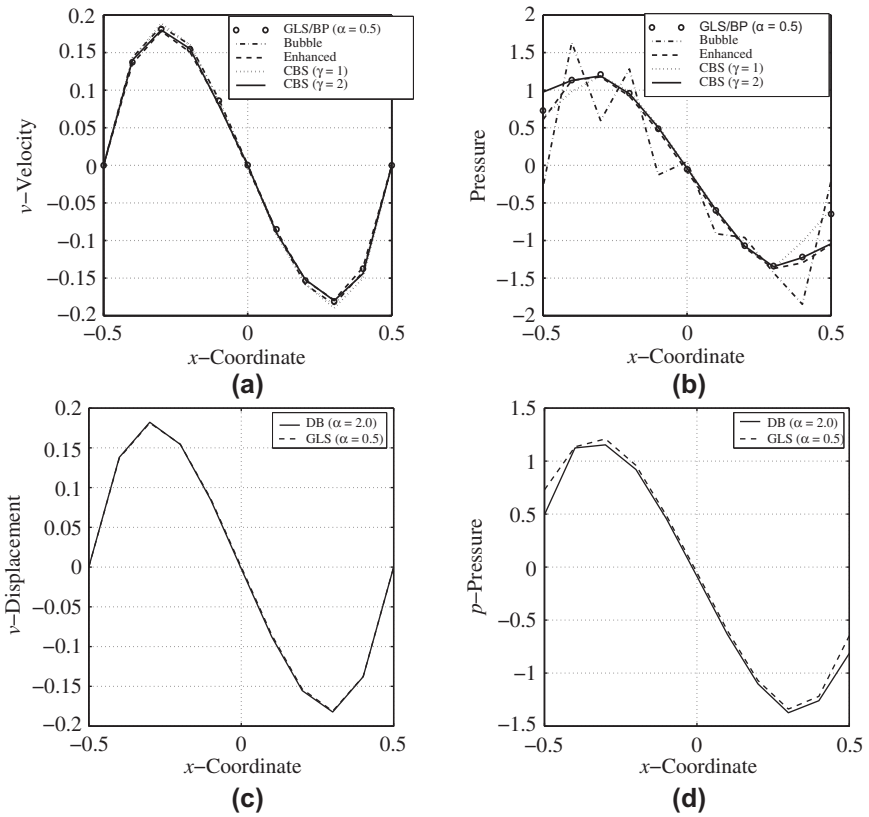



**FIGURE 10.13**
Vertical velocity and pressure for driven cavity problem: (a) vertical velocity on BB, (b) pressure on BB, (c) vertical velocity on BB, and (d) pressure on BB.

3. Direct pressure stabilization. Results for vertical velocity are again well captured as shown in Fig. 10.13; pressure results are also smooth and give good peak answers. We have not explored the range of $\alpha$ which may be used for the stabilization.
4. The CBS algorithm. Finally in Fig. 10.13 we present results using the CBS solution which may be compared with GLS, $\alpha = 0.5$. Once again the reader will observe that with $\gamma = 2$, the results of CBS reproduce very closely those of GLS, $\alpha = 0.5$. However, in results for $\gamma = 1$ no oscillations are observed and they are quite reasonable. This ratio for $\gamma$ is where the algorithm gives excellent results in incompressible flow modeling as will be demonstrated further in results presented in Ref. [1].

**Example 10.5.** Tension strip with slot

As our next example we consider a plane strain linear problem on a square domain with a central slot. The domain is two units square and the central slot has a total

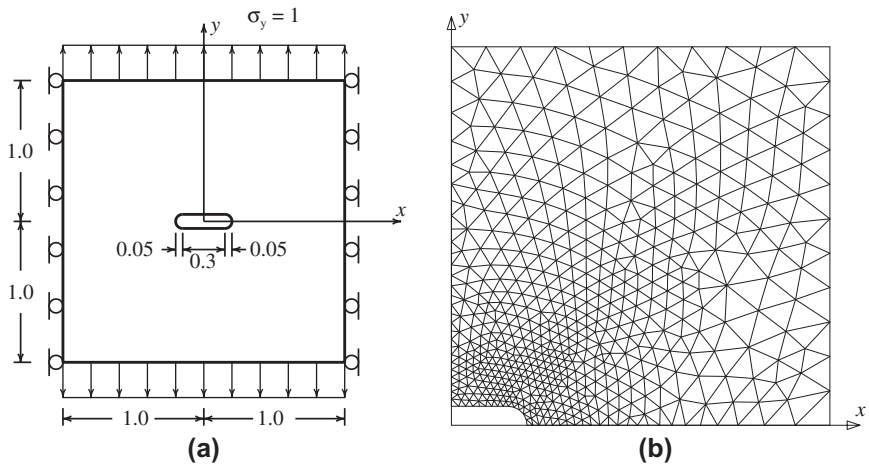



**FIGURE 10.14**
Region and mesh used for slotted tension strip: (a) problem definition and (b) mesh for quadrant.

width of 0.4 units and a height of 0.1 units. The ends of the slot are semicircular. Lateral boundaries have specified normal displacement and zero tangential traction. The top and bottom boundaries are uniformly stretched by a uniform axial loading and lateral boundaries are maintained at zero horizontal displacement. We consider the linear elastic problem with elastic properties $E = 24$ and $\nu = 0.5$, thus giving an incompressible situation. An unstructured mesh of triangles is constructed as shown in Fig. 10.14b. The problem is solved using direct pressure (DB), Galerkin least squares (GLS), and characteristic-based split (CBS) stabilization methods. Results for pressure along the horizontal and vertical center lines (i.e., the $x$- and $y$-axes) are presented in Fig. 10.15a and b. The distribution of the vertical displacement at the top is shown in Fig. 10.15c. We note that the results for this problem cause very strong gradients in stress near the ends of the slot. The mesh used for the analysis is not highly refined in this region and hence results from different analyses can be expected to differ in this region. The results obtained elsewhere using all three formulations are indistinguishable on the plot. In general the results achieved with all forms are satisfactory and indicate that stabilized methods may be considered for use in problems where constraints, such as incompressibility, are encountered.

## 10.8 Concluding remarks

In this chapter we have considered in some detail the application of mixed methods to incompressible problems and we have indicated some alternative procedures. The extension to nonisotropic problems and nonlinear problems are presented in Ref. [2], but will follow similar lines. Here we note how important the problem is in the context

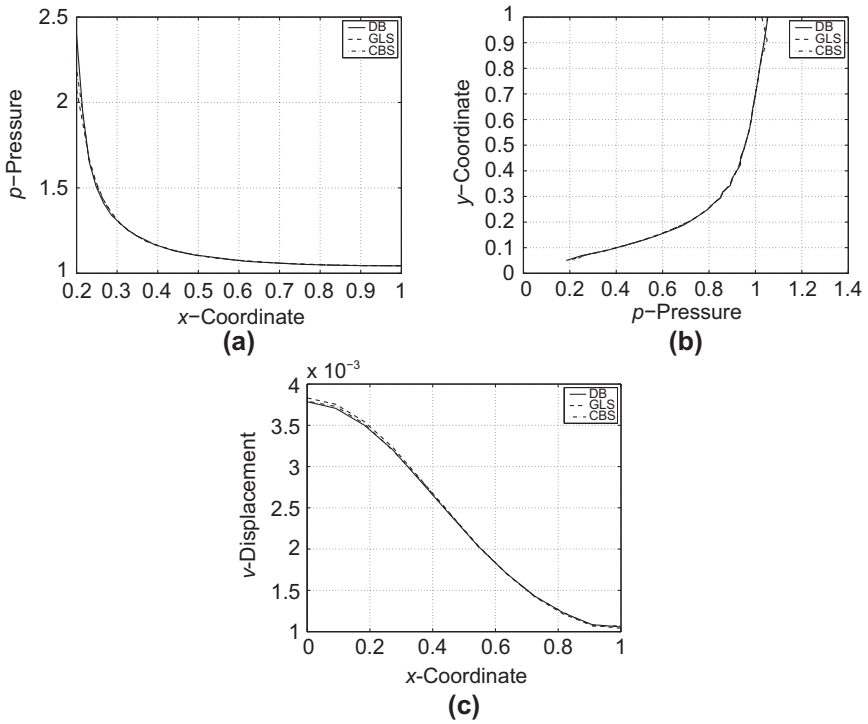



**FIGURE 10.15**

Pressures and displacement for slot problems: (a) pressure at *x*-axis, (b) pressure at *y*-axis, and (c) *x*-coordinate.

of fluid mechanics and it is there that much of the attention to it has been given as presented in Ref. [1].

In concluding this chapter we would like to point out three matters:

1. The mixed formulation discovers immediately the nonrobustness of certain irreducible (displacement) elements and, indeed, helps us to isolate those which perform well from those that do not. Thus it has merit at all times as a test that is applicable to many irreducible forms.
2. In elasticity, certain mixed forms work quite well at the near incompressible limit without resort to splits into deviatoric and mean parts. These include the two-field quadrilateral element of Pian-Sumihara and the enhanced strain quadrilateral element of Simo-Rifai which were presented in the previous chapter. There we noted how well such elements work for Poisson's ratio approaching one-half as compared to the standard irreducible element of a similar type.
3. Use of stabilizing forms such as the direct pressure or time-stepping form allows use of mixed $\mathbf{u}$-$p$ elements with equal order interpolation—a form which otherwise fails the mixed patch test (or Babuška-Brezzi condition).

## 10.9 Problems

**10.1** Show that the variational theorem

$$\Pi_{HR} = \int_\Omega \frac{1}{2}\boldsymbol{\epsilon}^T \mathbf{D}_d \boldsymbol{\epsilon}\, d\Omega - \int_\Omega \mathbf{u}^{\mathrm{T}}\mathbf{b}\, d\Omega - \int_\Gamma \mathbf{u}^{\mathrm{T}}\bar{\mathbf{t}}\, d\Gamma + \int_\Omega \left[p\mathbf{m}^{\mathrm{T}}\boldsymbol{\epsilon} - \frac{1}{2K}p^2\right] d\Omega$$

generates the problem given in Eq. (10.10) as its first variation.

**10.2** Show that the variational theorem given in Eq. (10.26) generates the problem given by (10.21).

**10.3** If the approximation for $p$ contains all the terms that are in the approximation to $\mathbf{m}^{\mathrm{T}}\boldsymbol{\epsilon}$ the limitation principle yields the result that the formulation will be identical to the standard displacement approximation given in Chapter 7.
An element will not lock if: (a) the number of internal displacement $\mathbf{u}$ degrees of freedom is equal to the number of parameters in the pressure $p$ and (b) the mixed patch test count and consistency conditions are passed.
Consider the case where $\mathbf{u}$ is $C_0$ continuous and $p$ is discontinuous:

- **(a)** The first four members of the rectangular Lagrangian family of two-dimensional plane strain elements is shown in Fig. 6.6. Consider the general member of this class in which the displacement $\mathbf{u}$ is of order $n$ (i.e., has $n+1$ nodes in each direction) and show on the Pascal triangle (viz. Fig. 6.3) the polynomial terms contained in the divergence term for volumetric strain $\mathbf{m}^{\mathrm{T}}\boldsymbol{\epsilon}$.
- **(b)** If the pressure $p$ is approximated by an order $m$ Lagrangian interpolation, determine the lowest order for $m$ which will contain all the polynomial terms found in (a).
- **(c)** Determine the lowest order of $n$ for which the limitation principle is satisfied and the element will not lock at the nearly incompressible limit.

**10.4** Repeat Problem 10.3 for the triangular family of elements in plane strain (viz. Fig. 6.2). Let the displacement $\mathbf{u}$ be approximated by an order $n$ polynomial (i.e., it has $n+1$ nodes on each edge):

- **(a)** Show on the Pascal triangle (viz. Fig. 6.3) the polynomial terms contained in the divergence term $\mathbf{m}^{\mathrm{T}}\boldsymbol{\epsilon}$.
- **(b)** What order approximation for $p$ will contain all the polynomial terms in (a)?
- **(c)** Determine the lowest order of $n$ for which the limitation principle is satisfied. Note that this order of approximation will yield a displacement formulation which will not "lock" near the incompressible limit.
- **(d)** Is the result valid for an axisymmetric geometry? Explain you answer.

**10.5** For a plane strain problem consider a linear triangular element in which the displacement approximation is given by

$$\mathbf{u} \approx \sum_{a=1}^{3} L_a \, \tilde{\mathbf{u}}$$

together with a constant approximation for the pressure $p$.
Using Eq. (10.23) compute $\bar{\mathbf{B}}$. Can this formulation be used to model nearly incompressible problems? Justify your answer.

**10.6** For an axisymmetric problem consider a linear triangular element in which the displacement approximation is given by

$$\mathbf{u} \approx \sum_{a=1}^{3} L_a \, \tilde{\mathbf{u}}$$

together with a constant approximation for the pressure $p$.
Using Eq. (10.23) compute $\bar{\mathbf{B}}$. Can this formulation be used to model nearly incompressible problems? Justify your answer.

**10.7** Consider a plane strain problem which is to be solved using a linear quadrilateral element with the displacement approximation

$$\mathbf{u} \approx \sum_{a=1}^{4} N_a \, \tilde{\mathbf{u}}$$

where $N_a = 1/4(1 + \xi_a \xi)(1 + \eta_a \eta)$ together with a constant approximation for the pressure $p$.
Let the element have a rectangular form with sides $a$ and $b$ in the $x$- and $y$-directions, respectively. Using Eq. (10.23) compute $\bar{\mathbf{B}}$. Can this formulation be used to model nearly incompressible problems? Justify your answer.

**10.8** Consider an axisymmetric problem which is to be solved using a linear quadrilateral element with the displacement approximation

$$\mathbf{u} \approx \sum_{a=1}^{4} N_a \, \tilde{\mathbf{u}}$$

where $N_a = 1/4(1 + \xi_a \xi)(1 + \eta_a \eta)$ together with a constant approximation for the pressure $p$.
Let the element have a rectangular form with sides $a$ and $b$ in the $r$- and $z$-directions, respectively. Using Eq. (10.23) compute $\bar{\mathbf{B}}$. Can this formulation be used to model nearly incompressible problems? Justify your answer.

**10.9** Consider a rectangular plane element with sides $a$ and $b$ in the $x$- and $y$-directions, respectively:

- **(a)** Compute the matrix $\mathbf{V}$ for GLS stabilization. Ignore second derivatives of $\mathbf{u}$.
- **(b)** Consider four elements of equal size with a central node $c$ and compute the assembled equation for the pressure $p$ at this node.

**10.10** Consider a rectangular axisymmetric element with sides $a$ and $b$ in the $r$- and $z$-directions, respectively. Let the inner radius of the element be located at $r_i > a/2$:

**(a)** Compute the matrix $\mathbf{V}$ for GLS stabilization. Ignore second derivatives of $\mathbf{u}$.

**(b)** Consider four elements of equal size with a central node $c$ located at $r_i$ and compute the assembled equation for this node.

**10.11** Consider a rectangular plane element with sides $a$ and $b$ in the $x$- and $y$-directions, respectively:

**(a)** Compute the matrix $\mathbf{V}$ for direct pressure stabilization.

**(b)** Consider four elements of equal size with a central node $c$ and compute the assembled equation for this node.

**10.12** Consider a rectangular axisymmetric element with sides $a$ and $b$ in the $r$- and $z$-directions, respectively. Let the inner radius of the element be located at $r_i > a/2$:

**(a)** Compute the matrix $\mathbf{V}$ for direct pressure stabilization.

**(b)** Consider four elements of equal size with a central node $c$ located at $r_i$ and compute the assembled equation for this node.

**10.13** The steel-rubber composite bearing shown in Fig. 10.16a is used to support a heavy machine. The bearing is to have high vertical stiffness but be flexible in shear (similar bearings are also used to support structures in seismic

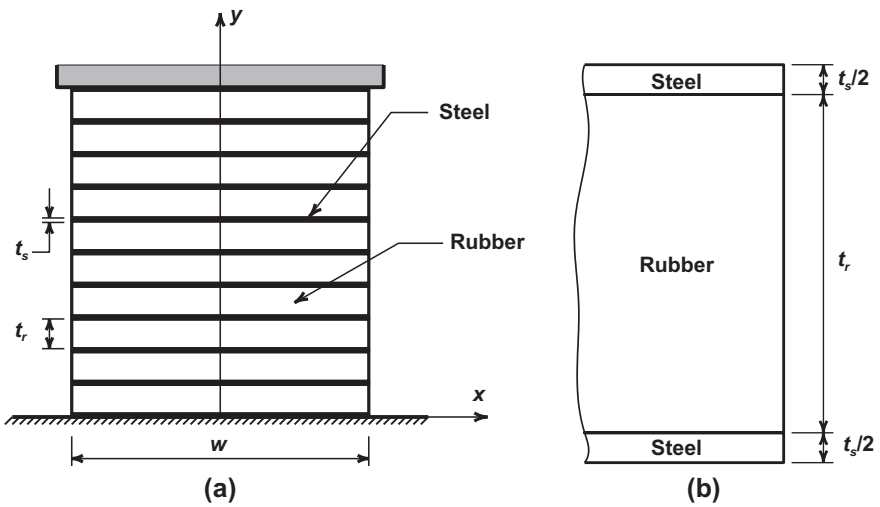



**FIGURE 10.16**
Support bearing for Problem 10.13: (a) steel-rubber layers and (b) layer for analysis.

regions). Consider a typical layer where $t_r = 1$cm and $t_s = 0.1$cm with a width $w = 5$ cm. Let the properties be $E_s = 200$ GPa, $\nu_s = 0.3$ and $E_r = 5$ GPa, $\nu_r = 0.495$. For a state of plane strain use *FEAPpv* (or any appropriate available program) to compute the stiffness for a vertical and a horizontal applied loading.

Use four-node and nine-node mixed $\mathbf{u}$-$p$-$\varepsilon_v$ elements to compute the vertical and horizontal stiffness of a single layer [viz. Fig. 10.16b]. Compare your solution to answers from a standard displacement formulation in $\mathbf{u}$.

**10.14** Program development project. Extend your program system started in Problem 7.20 to permit solution of a stabilized method as described in Section 10.7. You may select a stabilization scheme from either the GLS method of Section 10.7.2 or direct pressure stabilization method of Section 10.7.3.

Use your program to solve the driven cavity problem described in Example 10.4 Set the boundary velocity as shown in Fig. 10.12a and the nodal pressure to zero at the center of the bottom. Plot the values shown in Figs. 10.12 and 10.13. Also plot the contours for velocity components and pressure. Briefly discuss your findings.

## References

[1] O.C. Zienkiewicz, R.L. Taylor, P. Nithiarasu, The Finite Element Method for Fluid Dynamics, seventh ed., Elsevier, Oxford, in press.

[2] O.C. Zienkiewicz, R.L. Taylor, D. Fox, The Finite Element Method for Solid and Structural Mechanics, seventh ed., Elsevier, Oxford, in press.

[3] L.R. Herrmann, Finite element bending analysis of plates, in: Proceedings of the First Conference on Matrix Methods in Structural Mechanics, AFFDL-TR-66-80, Wright-Patterson Air Force Base, Ohio, 1965, pp. 577–602.

[4] S.W. Key, Variational principle for incompressible and nearly incompressible anisotropic elasticity, Int. J. Solids Struct. 5 (1969) 951–964.

[5] O.C. Zienkiewicz, R.L. Taylor, J.A.W. Baynham, Mixed and irreducible formulations in finite element analysis, in: S.N. Atluri, R.H. Gallagher, O.C. Zienkiewicz (Eds.), Hybrid and Mixed Finite Element Methods, John Wiley & Sons, 1983, pp. 405–431.

[6] D.N. Arnold, F. Brezzi, M. Fortin, A stable finite element for the Stokes equations, Calcolo 21 (1984) 337–344.

[7] M. Fortin, N. Fortin, Newer and newer elements for incompressible flow, in: R.H. Gallagher, G.F. Carey, J.T. Oden, O.C. Zienkiewicz (Eds.), Finite Elements in Fluids, vol. 6, John Wiley & Sons, 1985, pp. 171–188 (Chapter 7).

[8] J.T. Oden, R.I.P. methods for Stokesian flow, in: R.H. Gallagher, D.N. Norrie, J.T. Oden, O.C. Zienkiewicz (Eds.), Finite Elements in Fluids, vol. 4, John Wiley & Sons, 1982, pp. 305–318 (Chapter 15).

[9] M. Crouzcix, P.-A. Raviart, Conforming and non-conforming finite element methods for solving stationary Stokes equations, RAIRO 7 (R3) (1973) 33–76.

[10] D.S. Malkus, Eigenproblems associated with the discrete LBB condition for incompressible finite elements, Int. J. Eng. Sci. 19 (1981) 1299–1370.
[11] M. Fortin, Old and new finite elements for incompressible flow, Int. J. Numer. Methods Fluids 1 (1981) 347–364.
[12] C. Taylor, P. Hood, A numerical solution of the Navier-Stokes equations using the finite element technique, Comput. Fluids 1 (1973) 73–100.
[13] R.E. Bank, B.D. Welfert, A comparison between the MINI-element and the Petrov-Galerkin formulations for the generalized Stokes problem, Comput. Methods Appl. Mech. Eng. 83 (1990) 61–68.
[14] T.J.R. Hughes, Generalization of selective integration procedures to anisotropic and non-linear media, Int. J. Numer. Methods Eng. 15 (1980) 1413–1418.
[15] J.C. Simo, R.L. Taylor, K.S. Pister, Variational and projection methods for the volume constraint in finite deformation plasticity, Comput. Methods Appl. Mech. Eng. 51 (1985) 177–208.
[16] J.C. Simo, T.J.R. Hughes, Computational Inelasticity, Interdisciplinary Applied Mathematics, vol. 7, Springer-Verlag, Berlin, 1998.
[17] D.J. Naylor, Stresses in nearly incompressible materials for finite elements with application to the calculation of excess pore pressures, Int. J. Numer. Methods Eng. 8 (1974) 443–460.
[18] O.C. Zienkiewicz, P.N. Godbole, Viscous incompressible flow with special reference to non-Newtonian (plastic) flows, in: R.H. Gallagher et al. (Eds.), Finite Elements in Fluids, vol. 1, John Wiley & Sons, 1975, pp. 25–55 (Chapter 2).
[19] T.J.R. Hughes, R.L. Taylor, J.F. Levy, High Reynolds number, steady, incompressible flows by a finite element method, in: R.H. Gallagher et al. (Eds.), Finite Elements in Fluids, vol. 3, John Wiley & Sons, 1978.
[20] O.C. Zienkiewicz, J. Too, R.L. Taylor, Reduced integration technique in general analysis of plates and shells, Int. J. Numer. Methods Eng. 3 (1971) 275–290.
[21] S.F. Pawsey, R.W. Clough, Improved numerical integration of thick slab finite elements, Int. J. Numer. Methods Eng. 3 (1971) 575–586.
[22] O.C. Zienkiewicz, E. Hinton, Reduced integration, function smoothing and non-conformity in finite element analysis, J. Franklin Inst. 302 (1976) 443–461.
[23] D.S. Malkus, T.J.R. Hughes, Mixed finite element methods – reduced and selective integration techniques: a unification of concepts, Comput. Methods Appl. Mech. Eng. 15 (1978) 63–81.
[24] O.C. Zienkiewicz, S. Nakazawa, On variational formulations and its modification for numerical solution, Comput. Struct. 19 (1984) 303–313.
[25] M.S. Engleman, R.L. Sani, P.M. Gresho, H. Bercovier, Consistent vs. reduced integration penalty methods for incompressible media using several old and new elements, Int. J. Numer. Methods Fluids 2 (1982) 25–42.
[26] D.N. Arnold, Discretization by finite elements of a model parameter dependent problem, Numer. Math. 37 (1981) 405–421.

[27] M.J.D. Powell, A method for nonlinear constraints in minimization problems, in: R. Fletcher (Ed.), Optimization, Academic Press, London, 1969.
[28] M. Vogelius, An analysis of the *p*-version of the finite element method for nearly incompressible materials: uniformly optimal error estimates, Numer. Math. 41 (1983) 39–53.
[29] S.W. Sloan, M.F. Randolf, Numerical prediction of collapse loads using finite element methods, Int. J. Numer. Anal. Methods Geomech. 6 (1982) 47–76.
[30] J.C. Nagtegaal, D.M. Parks, J.R. Rice, On numerical accurate finite element solutions in the fully plastic range, Comput. Methods Appl. Mech. Eng. 4 (1974) 153–177.
[31] O.C. Zienkiewicz, J.P. Vilotte, S. Toyoshima, S. Nakazawa, Iterative method for constrained and mixed approximation. An inexpensive improvement of FEM performance, Comput. Methods Appl. Mech. Eng. 51 (1985) 3–29.
[32] K.J. Arrow, L. Hurwicz, H. Uzawa, Studies in Non-Linear Programming, Stanford University Press, Stanford, CA, 1958.
[33] M.R. Hestenes, Multiplier and gradient methods, J. Opt. Theory Appl. 4 (1969) 303–320.
[34] C.A. Felippa, Iterative procedure for improving penalty function solutions of algebraic systems, Int. J. Numer. Methods Eng. 12 (1978) 165–185.
[35] M. Fortin, F. Thomasset, Mixed finite element methods for incompressible flow problems, J. Comput. Phys. 31 (1973) 113–145.
[36] M. Fortin, R. Glowinski, Augmented Lagrangian Methods: Applications to Numerical Solution of Boundary-Value Problems, North-Holland, Amsterdam, 1983.
[37] D.G. Luenberger, Linear and Nonlinear Programming, Addison-Wesley, Reading, Mass, 1984.
[38] J.H. Argyris, Three-dimensional anisotropic and inhomogeneous media – matrix analysis for small and large displacements, Ing. Arch. 34 (1965) 33–55.
[39] O.C. Zienkiewicz, S. Valliappan, Analysis of real structures for creep plasticity and other complex constitutive laws, in: M. Te'eni (Ed.), Structure of Solid Mechanics and Engineering Design, vol. Part 1, 1971, pp. 27–48.
[40] O.C. Zienkiewicz, The Finite Element Method in Engineering Science, second ed., McGraw-Hill, London, 1971.
[41] R. Courant, Variational methods for the solution of problems of equilibrium and vibration, Bull. Am. Math. Soc. 49 (1943) 1–61.
[42] F. Brezzi, J. Pitkäranta, On the stabilization of finite element approximations of the Stokes problem, in: W. Hackbusch (Ed.), Efficient Solution of Elliptic Problems, Notes on Numerical Fluid Mechanics, vol. 10, Vieweg, Wiesbaden, 1984.
[43] T.J.R. Hughes, L.P. Franca, M. Balestra, A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška-Brezzi condition: a stable Petrov-Galerkin formulation of the Stokes problem accommodating equal-order interpolations, Comput. Methods Appl. Mech. Eng. 59 (1986) 85–99.

[44] T.J.R. Hughes, L.P. Franca, A new finite element formulation for computational fluid dynamics: VII. The Stokes problem with various well-posed boundary conditions: symmetric formulation that converge for all velocity/pressure spaces, Comput. Methods Appl. Mech. Eng. 65 (1987) 85–96.
[45] T.J.R. Hughes, L.P. Franca, G.M. Hulbert, A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective-diffusive equations, Comput. Methods Appl. Mech. Eng. 73 (1989) 173–189.
[46] E. Oñate, Derivation of stabilized equations for numerical solution of advective-diffusive transport and fluid flow problems, Comput. Methods Appl. Mech. Eng. 151 (1998) 233–265.
[47] C.R. Dohrmann, P.B. Bochev, A stabilized finite element method for the Stokes problem based on polynomial pressure projections, Int. J. Numer. Methods Fluids 46 (2004) 183–201.
[48] O.C. Zienkiewicz, J. Wu, Incompressibility without tears! How to avoid restrictions of mixed formulations, Int. J. Numer. Methods Eng. 32 (1991) 1184–1203.
[49] O.C. Zienkiewicz, R. Codina, A general algorithm for compressible and incompressible flow — Part I: The split, characteristic-based scheme, Int. J. Numer. Methods Fluids 20 (1995) 869–885.
[50] O.C. Zienkiewicz, P. Nithiarasu, R. Codina, M. Vazquez, P. Ortiz, The characteristic-based-split (CBS) procedure: an efficient and accurate algorithm for fluid problems, Int. J. Numer. Methods Fluids 31 (1999) 359–392.
[51] O.C. Zienkiewicz, K. Morgan, B.V.K. Satya Sai, R. Codina, M. Vasquez, A general algorithm for compressible and incompressible flow – Part II: Tests on the explicit form, Int. J. Numer. Methods Fluids 20 (1995) 887–913.
[52] P. Nithiarasu, O.C. Zienkiewicz, On stabilization of the CBS algorithm. Internal and external time steps, Int. J. Numer. Methods Eng. 48 (2000) 875–880.
[53] P. Nithiarasu, An efficient artificial compressibility (AC) scheme based on the characteristic based split (CBS) method for incompressible flows, Int. J. Numer. Methods Eng. 56 (2003) 1815–1845.
[54] O.C. Zienkiewicz, Origins, milestones and directions of the finite element method. A personal view, in: P.G. Ciarlet, J.L. Lyons (Eds.), Handbook of Numerical Analysis, vol. IV, North Holland, 1996, pp. 3–65.
[55] R. Pierre, Simple $C^0$ approximations for the computation of incompressible flows, Comput. Methods Appl. Mech. Eng. 68 (1988) 205–227.

This page is intentionally left blank

CHAPTER

# 11 Multidomain Mixed Approximations

## 11.1 Introduction

In the previous chapters we have assumed in the approximations that all the variables were defined in the same manner throughout the domain of the analysis. This process can, however, be conveniently abandoned on occasion with the same or different formulations adopted in different subdomains of the problem. In this case some variables are only approximated on surfaces joining such subdomains.

There are two motivations for separating the whole domain into several subdomain regions. In the first of these the concept of parallel computation is paramount. Such parallel computation has become very important in many fields of engineering and allows us to use completely different methodologies for solving the problem in each individual part and even if this is not used allows us to very much increase the computer power by having separate operations going on simultaneously. In general the process we have just mentioned is referred to as *domain decomposition*. This volume is not concerned in detail with the process of calculation and therefore does not discuss the subject of parallel computing in extended form. We refer the reader to references on the subject [1–7].

It is of interest to note, however, that the methodologies for connecting separate subdomains can have other outcomes and objectives. In many applications the mesh parts for an analysis are prepared by different people and need to be merged for analysis. In other applications two or more parts of a problem may come into "contact" at different stages of the analysis. Thus, the second motivation for considering multidomain problems is to develop analysis procedures for these types of applications. In this chapter we consider the basic problem of merging two or more domains into a single object. These form the basis for treating contact problems, which due to its intermittent nature is a nonlinear problem which we treat in Ref. [8].

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00011-3

## 11.2 Linking of two or more subdomains by Lagrange multipliers

In this section we deal with the problem of connecting two or more subdomains in which standard finite element approximations of one form or another have been used. In particular, we shall give as examples the process in which "irreducible" formulations are used but, of course, other approximations could be introduced. The linking of such subdomains can be easily accomplished by the introduction of Lagrange multiplier methods to which we already referred to in Chapter 4 and elsewhere. The Lagrange multipliers for this case are defined on the boundary interface of the connecting subdomains.

In the present case we consider two subdomains, $\Omega^1$ and $\Omega^2$, which are to be joined together along an interface $\Gamma_I$. The generalization to multiple domains follows the same pattern. Independently approximated *Lagrange multipliers* (fluxes or tractions) are used on the interface to join the subdomains, as in Fig. 11.1.

In the first problem considered we treat the quasi-harmonic equation expressed in terms of the scalar potential function $\phi$. This is followed by treatment for the elasticity problem.

### 11.2.1 Linking subdomains for quasi-harmonic equations

In Chapter 5 we considered the general problem for field problems. This problem resulted in a weak form in terms of a potential function $\phi$. The approximation in the two domains for a steady-state problem may be expressed as [viz. see Eq. (5.7)]

$$\begin{aligned}
&\int_{\Omega^1}\left[\left(\nabla\delta\phi^1\right)^{\mathrm{T}}\left(\mathbf{k}^1\,\nabla\phi^1\right)+\delta\phi^1\,Q\right]\mathrm{d}\Omega+\int_{\Gamma_q}\delta\phi^1\left(\bar{q}^1+H^1\,\phi^1\right)\mathrm{d}\Gamma\\
&\quad+\int_{\Gamma_I}\delta\phi^1\,\lambda\,\mathrm{d}\Gamma=0\\
&\int_{\Omega^2}\left[\left(\nabla\delta\phi^2\right)^{\mathrm{T}}\left(\mathbf{k}^2\,\nabla\phi^2\right)+\delta\phi^2\,Q\right]\mathrm{d}\Omega+\int_{\Gamma_q}\delta\phi^2\left(\bar{q}^2+H^2\,\phi^2\right)\mathrm{d}\Gamma\\
&\quad-\int_{\Gamma_I}\delta\phi^2\,\lambda\,\mathrm{d}\Gamma=0
\end{aligned}\tag{11.1}$$

where the normal flux has been replaced by a Lagrange multiplier function $\lambda$ defined on the interface $\Gamma_I$ by

$$G_\lambda=\int_{\Gamma_I}\delta\left[\lambda\left(\phi^1-\phi^2\right)\right]\mathrm{d}\Gamma\tag{11.2}$$

to satisfy continuity at the interface. The two subdomain equations are completed by the remaining weak statement of continuity of the potential between the two subdomains given by

$$\int_{\Gamma_I}\delta\lambda\,\left(\phi^1-\phi^2\right)\mathrm{d}\Gamma=0\tag{11.3}$$

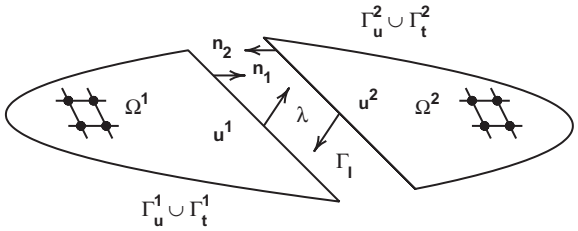



**FIGURE 11.1**

Linking two (or more) domains by traction variables defined only on the interfaces. The variables in each domain are displacement $\mathbf{u}$ (irreducible form).

Discretization of the potential in each domain and the Lagrange multiplier on the interface yields the final set of equations. Thus expressing the independent approximations as

$$\phi^1 = \mathbf{N}_1 \tilde{\boldsymbol{\phi}}^1, \quad \phi^2 = \mathbf{N}_2 \tilde{\boldsymbol{\phi}}^2 \quad \text{and} \quad \lambda = \mathbf{N}_\lambda \tilde{\boldsymbol{\lambda}} \tag{11.4}$$

we have the standard Lagrange multiplier form

$$\begin{bmatrix} \mathbf{H}^1 & \mathbf{0} & \mathbf{Q}^1 \\ \mathbf{0} & \mathbf{H}^2 & \mathbf{Q}^2 \\ \mathbf{Q}^{1\mathrm{T}} & \mathbf{Q}^{2\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \tilde{\boldsymbol{\phi}}^1 \\ \tilde{\boldsymbol{\phi}}^2 \\ \tilde{\boldsymbol{\lambda}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}^1 \\ \mathbf{f}^2 \\ \mathbf{0} \end{Bmatrix} \tag{11.5}$$

where

$$\begin{aligned}
\mathbf{H}^1 &= \int_{\Omega^1} (\nabla \mathbf{N}^1)^{\mathrm{T}} (\mathbf{k}^1 \nabla \mathbf{N}^1)\, \mathrm{d}\Omega, & \mathbf{H}^2 &= \int_{\Omega^2} (\nabla \mathbf{N}^2)^{\mathrm{T}} (\mathbf{k}^2 \nabla \mathbf{N}^2)\, \mathrm{d}\Omega \\
&\quad + \int_{\Gamma_q^1} \mathbf{N}_1^{\mathrm{T}} H^1 \mathbf{N}_1\, \mathrm{d}\Gamma, & &\quad + \int_{\Gamma_q^2} \mathbf{N}_2^{\mathrm{T}} H^2 \mathbf{N}_2\, \mathrm{d}\Gamma \\
\mathbf{Q}^1 &= \int_{\Gamma_I} \mathbf{N}_1^{\mathrm{T}} \mathbf{N}_\lambda\, \mathrm{d}\Gamma, & \mathbf{Q}^2 &= -\int_{\Gamma_I} \mathbf{N}_2^{\mathrm{T}} \mathbf{N}_\lambda\, \mathrm{d}\Gamma \\
\mathbf{f}^1 &= -\int_{\Omega^1} \mathbf{N}_1^{\mathrm{T}} Q^1\, \mathrm{d}\Omega - \int_{\Gamma_q^1} \mathbf{N}_1^{\mathrm{T}} \bar{q}^1\, \mathrm{d}\Gamma, & \mathbf{f}^2 &= -\int_{\Omega^2} \mathbf{N}_2^{\mathrm{T}} Q^2\, \mathrm{d}\Omega - \int_{\Gamma_q^2} \mathbf{N}_2^{\mathrm{T}} \bar{q}^2\, \mathrm{d}\Gamma
\end{aligned} \tag{11.6}$$

The formulation outlined above for two domains can obviously be extended to many subdomains and in many cases of practical analysis is useful in ensuring a better matrix conditioning and allowing the solution to be obtained with reduced computational effort [9].

The variables $\tilde{\boldsymbol{\phi}}^1$ and $\tilde{\boldsymbol{\phi}}^2$ appear as internal and boundary variables within each subdomain (or superelement) and can be eliminated locally provided the matrices $\mathbf{H}^1$ and $\mathbf{H}^2$ are nonsingular. Such nonsingularity presupposes, however, that each of the subdomains has enough prescribed values to prevent the singular modes. If this is not the case partial elimination is always possible, retaining the singular modes until the complete solution is achieved.

We note that in the derivation of the matrices in Eq. (11.6) the shape function $\mathbf{N}_\lambda$ and hence $\lambda$ itself are only specified along the interface surface. The choice of appropriate functions for the $\mathbf{N}_\lambda$ must, of course, satisfy the mixed patch requirement with counts performed for the interface degree of freedoms and stability evaluated as discussed previously. Here the count condition can be more difficult to satisfy when multiple subdomains are connected at a point or along a line due to the presence of multiple $\lambda$ functions at these locations. One procedure to satisfy the condition is to use *mortar* or *dual mortar* methods. This matter is taken up later in this section. However, prior to this we consider the use of the above to include the Dirichlet boundary condition $\phi - \bar{\phi} = 0$ as part of the weak solution to the problem.

#### 11.2.1.1 Treatment for forced boundary conditions

We note that the above form also may be used to satisfy the forced boundary condition $\phi = \bar{\phi}$ on $\Gamma_\phi$. For this we let $\Gamma_I = \Gamma_\phi$ and from Eq. (11.1) (dropping the superscript "1") obtain

$$\int_\Omega \left[ \left(\nabla \delta\phi\right)^{\mathrm{T}} \left(\mathbf{k}\, \nabla\phi\right) + \delta\phi\, Q \right] \mathrm{d}\Omega + \int_{\Gamma_q} \delta\phi \left(\bar{q} + H\,\phi\right) \mathrm{d}\Gamma + \int_{\Gamma_\phi} \delta\phi\, \lambda \, \mathrm{d}\Gamma = 0 \tag{11.7}$$

Similarly, from Eq. (11.3) with $\phi^2 = \bar{\phi}$ we have

$$\int_{\Gamma_\phi} \delta\lambda \, \left(\phi - \bar{\phi}\right) \mathrm{d}\Gamma = 0 \tag{11.8}$$

The discrete form of the equations becomes

$$\begin{bmatrix} \mathbf{H} & \mathbf{Q} \\ \mathbf{Q}^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \tilde{\boldsymbol{\phi}} \\ \tilde{\boldsymbol{\lambda}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ \mathbf{f}_\lambda \end{Bmatrix} \tag{11.9}$$

where

$$\begin{aligned} \mathbf{H} &= \int_\Omega \left(\nabla \mathbf{N}\right)^T \left(\mathbf{k} \nabla \mathbf{N}\right) \mathrm{d}\Omega + \int_{\Gamma_q} \mathbf{N}^T H \mathbf{N} \, \mathrm{d}\Gamma \\ \mathbf{Q} &= \int_{\Gamma_\phi} bf N^T \mathbf{N}_\lambda \mathrm{d}\Gamma; \mathbf{f}_\lambda = \int_{\Gamma_\phi} \mathbf{N}_\lambda^T \bar{\phi} \, \mathrm{d}\Gamma \\ \mathbf{f} &= -\int_\Omega \mathbf{N}^T Q \, \mathrm{d}\Omega - \int_{\Gamma_q} \mathbf{N}^T \bar{q} \, \mathrm{d}\Gamma \end{aligned} \tag{11.10}$$

in which

$$\phi = \mathbf{N}\, \tilde{\boldsymbol{\phi}} \quad \text{and} \quad \lambda = \mathbf{N}_\lambda\, \tilde{\boldsymbol{\lambda}} \tag{11.11}$$

#### 11.2.1.2 Mortar and dual mortar methods

The *mortar method* is a procedure which is used to join multiple subdomains [10]. Consider as an example a two-dimensional problem in which we use four-node (bilinear) quadrilaterals in $\Omega^1$ and nine-node (biquadratic) quadrilaterals in $\Omega^2$. To connect subdomains the Lagrange multiplier may be approximated as shown in Fig. 11.2 for

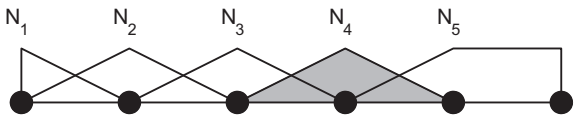



**FIGURE 11.2**

Mortar function for Lagrange multiplier. Form for linear edges on $\Omega^1$ elements.

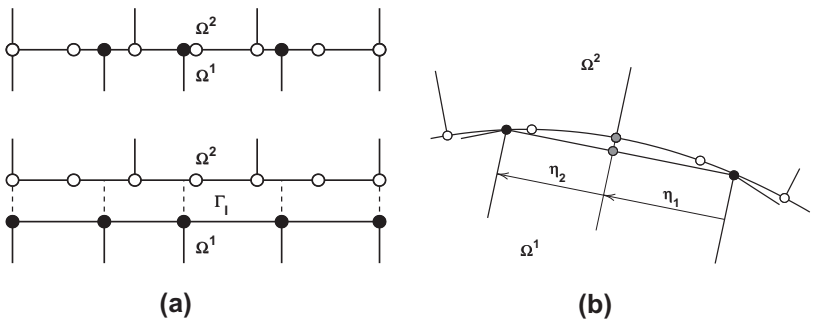



**FIGURE 11.3**

Two-dimensional mortar interface: (a) interface for $\Omega^1$ and $\Omega^2$ and (b) subincrements for quadrature.

a subdomain with five segments along the interface. The use of the constant part at an end is required only if multiple subdomains exist at the end point, otherwise the interpolation may be continued with normal linear interpolation as shown for the left end in Fig. 11.2. Along the interface we may connect subdomains with different number of segments as shown in Fig. 11.3a. Thus, if we assume the Lagrange multiplier interpolation uses $\mathbf{N}_\lambda = \mathbf{N}_1$ (except at end points)[1], the interface term resulting from (11.13) yields

$$\mathbf{Q}^1 = \int_{\Gamma_I} \mathbf{N}_1^{\mathrm{T}} \mathbf{N}_1 \, \mathrm{d}\Gamma \quad \text{and} \quad \mathbf{Q}^2 = -\int_{\Gamma_I} \mathbf{N}_2^{\mathrm{T}} \mathbf{N}_1 \, \mathrm{d}\Gamma \tag{11.12}$$

The integral for $\mathbf{Q}^1$ may be evaluated for each element edge using quadrature described in Chapter 3; however, evaluation by quadrature of the integral for $\mathbf{Q}^2$ requires further subdivision into *subincrements* along the element edges as indicated in Fig. 11.3b.

The *dual mortar method* is an alternate form of the mortar method which has advantages for Lagrange multiplier and penalty forms. The dual shape functions are defined to satisfy

$$\int_{\Gamma_e} \widehat{N}_a \, N_b \, \mathrm{d}\Omega = \delta_{ab} \int_{\Gamma_e} N_b \mathrm{d}\Omega \tag{11.13}$$

[1] An alternative avoiding modification at end points is to use a stabilization method such as one equivalent to the direct pressure stabilization presented in Section 10.7.3.

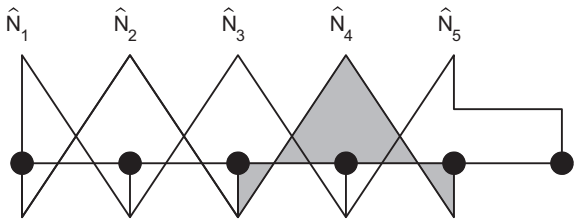



**FIGURE 11.4**

Dual mortar function for Lagrange multiplier. Form for linear edges on $\Omega^1$ elements.

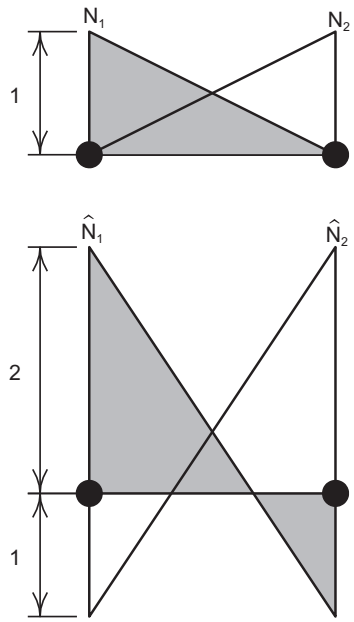



**FIGURE 11.5**

Mortar and dual mortar shape functions for two-dimensional linear edge.

where $\widehat{N}_a$ denotes a dual shape function and $\delta_{ab}$ is a Kronecker delta function. Figure 11.4 shows the dual functions computed for the standard linear functions shown in Fig. 11.2. The dual functions are *discontinuous* between elements, which is permitted since no derivatives appear for the Lagrange multipliers $\boldsymbol{\lambda}$.

The dual functions may be computed for each element edge separately. For linear edges the result is shown in Fig. 11.5. The process may be repeated for higher order functions without difficulty; however, nodes appear between the segment ends and, thus, with arbitrary spacing the computation must be computed for each case separately.

The advantage of the dual functions is evident from the definition of $\mathbf{Q}^1$ (assuming again that $\mathbf{N}_\lambda = \mathbf{N}_1$ on the boundary). Here we observe that

$$\mathbf{Q}^1 = \int_{\Gamma_I} \mathbf{N}_1^{\mathrm{T}} \widehat{\mathbf{N}}_\lambda \mathrm{d}\Gamma = \widehat{\mathbf{Q}}^1 \tag{11.14}$$

where $\widehat{\mathbf{Q}}^1$ is *diagonal* by the properties of Eq. (11.13).

### 11.2.2 Linking subdomains for elasticity equations

In this problem we formulate the approximation in domain $\Omega^1$ in terms of displacements $\mathbf{u}^1$ resulting from an irreducible (displacement) form of the elasticity equations. The traction $\mathbf{t}^1$ on the interface is denoted by $\boldsymbol{\lambda}$, thus traction continuity requires $\mathbf{t}^2 = -\boldsymbol{\lambda}$. With the weak form using a standard weak form expression [see Eq. (7.9)] we have[2]

$$\begin{aligned}
\int_{\Omega^1} \delta(\mathcal{S}\mathbf{u}^1)^{\mathrm{T}} \mathbf{D}^1 \mathcal{S}\mathbf{u}^1 \mathrm{d}\Omega - \int_{\Gamma_I} \delta\mathbf{u}^{1\mathrm{T}} \boldsymbol{\lambda} \mathrm{d}\Gamma - \int_{\Omega^1} \delta\mathbf{u}^{1\mathrm{T}} \mathbf{b}^1 \mathrm{d}\Omega - \int_{\Gamma_t^1} \delta\mathbf{u}^{1\mathrm{T}} \bar{\mathbf{t}}^1 \mathrm{d}\Gamma = 0 \\
\int_{\Omega^2} \delta(\mathcal{S}\mathbf{u}^2)^{\mathrm{T}} \mathbf{D}^2 \mathcal{S}\mathbf{u}^2 \,\mathrm{d}\Omega + \int_{\Gamma_I} \delta\mathbf{u}^{2\mathrm{T}} \boldsymbol{\lambda} \,\mathrm{d}\Gamma - \int_{\Omega^2} \delta\mathbf{u}^{2\mathrm{T}} \mathbf{b}^2 \,\mathrm{d}\Omega - \int_{\Gamma_t^2} \delta\mathbf{u}^{2\mathrm{T}} \bar{\mathbf{t}}^2 \,\mathrm{d}\Gamma = 0 \\
\int_{\Gamma_I} \delta\boldsymbol{\lambda}^{\mathrm{T}} (\mathbf{u}^2 - \mathbf{u}^1) \,\mathrm{d}\Gamma = 0
\end{aligned} \tag{11.15}$$

in which as usual we assume that the satisfaction of the prescribed displacement on $\Gamma_u^i$ is implied by the approximation for $\mathbf{u}^i$.

Discretization of displacements in each domain and of the Lagrange multipliers (tractions) $\boldsymbol{\lambda}$ on the interface yields the final system of equations. Thus putting the independent approximations as

$$\mathbf{u}^1 = \mathbf{N}_1 \tilde{\mathbf{u}}^1; \quad \mathbf{u}^2 = \mathbf{N}_2 \tilde{\mathbf{u}}^2; \quad \boldsymbol{\lambda} = \mathbf{N}_\lambda \tilde{\boldsymbol{\lambda}} \tag{11.16}$$

we have

$$\begin{bmatrix} \mathbf{K}^1 & \mathbf{0} & \mathbf{Q}^1 \\ \mathbf{0} & \mathbf{K}^2 & \mathbf{Q}^2 \\ \mathbf{Q}^{1\mathrm{T}} & \mathbf{Q}^{2\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}}^1 \\ \tilde{\mathbf{u}}^2 \\ \tilde{\boldsymbol{\lambda}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}^1 \\ \mathbf{f}^2 \\ \mathbf{0} \end{Bmatrix} \tag{11.17}$$

where

$$\begin{aligned}
\mathbf{K}^1 &= \int_{\Omega^1} \mathbf{B}^{1\mathrm{T}} \mathbf{D}^1 \mathbf{B}^1 \,\mathrm{d}\Omega & \mathbf{K}^2 &= \int_{\Omega^2} \mathbf{B}^{2\mathrm{T}} \mathbf{D}^2 \mathbf{B}^2 \,\mathrm{d}\Omega \\
\mathbf{Q}^1 &= -\int_{\Gamma_I} \mathbf{N}_1^{\mathrm{T}} \mathbf{N}_\lambda \,\mathrm{d}\Gamma & \mathbf{Q}^2 &= \int_{\Gamma_I} \mathbf{N}_2^{\mathrm{T}} \mathbf{N}_\lambda \,\mathrm{d}\Gamma \\
\mathbf{f}^1 &= \int_{\Omega^1} \mathbf{N}_1^{\mathrm{T}} \mathbf{b}^1 \,\mathrm{d}\Omega + \int_{\Gamma^1} \mathbf{N}_1^{\mathrm{T}} \bar{\mathbf{t}}^1 \,\mathrm{d}\Gamma & \mathbf{f}^2 &= \int_{\Omega^2} \mathbf{N}_2^{\mathrm{T}} \mathbf{b}^2 \,\mathrm{d}\Omega + \int_{\Gamma^2} \mathbf{N}_2^{\mathrm{T}} \bar{\mathbf{t}}^2 \,\mathrm{d}\Gamma
\end{aligned} \tag{11.18}$$

[2]Here we replace $\boldsymbol{\epsilon}$ and $\delta\boldsymbol{\epsilon}$ by $\mathcal{S}\mathbf{u}$ and $\mathcal{S}\delta\mathbf{u}$, respectively.

The process described here is very similar to that introduced by Kron [11] at a very early date and, more recently, used by Farhat et al. in the FETI (finite element tearing and interconnecting) method [12] which uses the process on many individual element partitions as a means of iteratively solving large problems.

The formulation is, of course, subject to limitations imposed by the stability and consistency conditions of the mixed patch test for selection of the appropriate number of $\boldsymbol{\lambda}$ variables.

The formulation just used can, of course, be applied to a single-field displacement formulation in which we are required to specify the displacement on the boundaries in a weak sense—rather than imposing these directly on displacement shape functions.

This problem can be approached directly or can be derived simply using (11.15) in which we put $\mathbf{u}^2 = \bar{\mathbf{u}}$, the specified displacement on $\Gamma_I \equiv \Gamma_u$.

Now the equation system is simply

$$\begin{bmatrix} \mathbf{K}^1 & \mathbf{Q}^1 \\ \mathbf{Q}^{1\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}}^1 \\ \tilde{\boldsymbol{\lambda}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{f}_\lambda \end{Bmatrix} \tag{11.19}$$

where

$$\mathbf{f}_\lambda = -\int_{\Gamma_I} \mathbf{N}_\lambda^{\mathrm{T}} \bar{\mathbf{u}} \, \mathrm{d}\Gamma \tag{11.20}$$

This formulation is often convenient for imposing a prescribed displacement on an element field when the boundary values cannot fit the shape function form.

We have approached the above formulation directly via weak or weighted residual forms. A variational principle could be given here simply as the minimization of total potential energy (see Chapter 4 or 7) subject to a Lagrange multiplier $\boldsymbol{\lambda}$ imposing subdomain continuity. The stationarity of

$$\begin{aligned} \Pi = \sum_{i=1}^{2} \left[ \frac{1}{2} \int_{\Omega^i} (\boldsymbol{\mathcal{S}}\mathbf{u}^i)^{\mathrm{T}} \mathbf{D}^i \boldsymbol{\mathcal{S}}\mathbf{u}^i \, \mathrm{d}\Omega - \int_{\Omega^i} \mathbf{u}^{i\mathrm{T}} \mathbf{b} \, \mathrm{d}\Omega - \int_{\Gamma_t^i} \mathbf{u}^{i\mathrm{T}} \bar{\mathbf{t}} \, \mathrm{d}\Gamma \right] \\ + \int_{\Gamma_I} \boldsymbol{\lambda}^{\mathrm{T}} (\mathbf{u}^2 - \mathbf{u}^1) \, \mathrm{d}\Gamma \end{aligned} \tag{11.21}$$

would result in the equation set (11.15). The formulation is, of course, subject to limitations imposed by the stability and consistency conditions of the mixed patch test for selection of the appropriate number of $\boldsymbol{\lambda}$ variables.

**Example 11.1.** A mortar method for two dimensional elasticity

Mortar and dual mortar methods may also be used in the solution of elasticity problems. The formulation follows that given for the quasi-harmonic equation with appropriate change in variables. To indicate the type of result which occurs using mortar or dual mortar forms we consider the problem of a strip loaded by a uniform pressure along a short segment. The problem is solved as a single region using a fine mesh over the whole domain and also by a two subdomain form in which fine elements are in the top layer only (see Fig. 11.6). Contours for the vertical displacement are

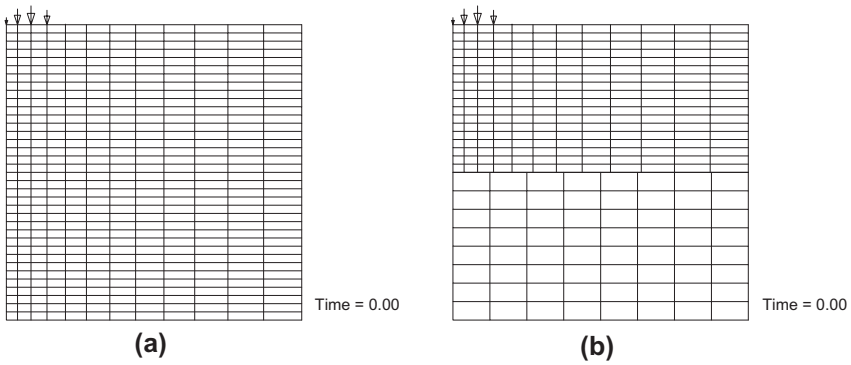



**FIGURE 11.6**
Mesh and nodal loading for vertically loaded strip: (a) no interface and (b) mortar interface.

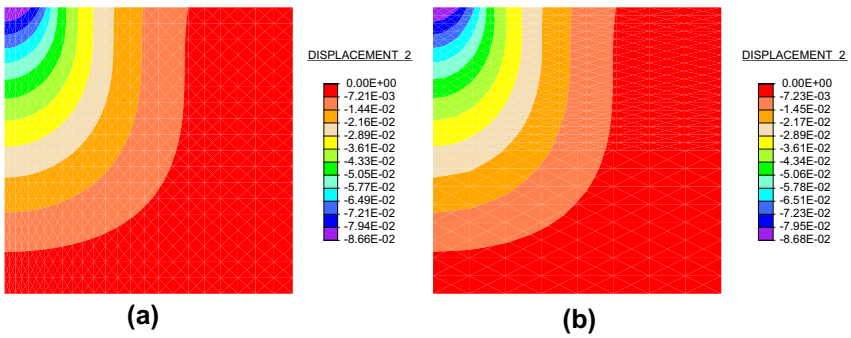



**FIGURE 11.7**
Vertical displacement for strip loaded over short segment of top: (a) no interface and (b) mortar interface.

presented in Fig. 11.7 for the two cases. It is evident that the mortar treatment produces excellent continuity in displacement.

## 11.3 Linking of two or more subdomains by perturbed Lagrangian and penalty methods

In the previous section we have shown how linking can be achieved using Lagrange multipliers. A disadvantage of the Lagrange multiplier approach is the addition of extra unknowns (the Lagrange multipliers $\boldsymbol{\lambda}$) and the creation of equations which have zero on the diagonal. As we have shown previously (viz. Chapter 4) it is possible to avoid both of these situations using a *perturbed Lagrangian* or *penalty* form.

The perturbed Lagrangian form of the equations may be achieved by modifying the Lagrange multiplier terms in (11.15) to

$$\int_{\Gamma_I} \delta\boldsymbol{\lambda}^{\mathrm{T}}(\mathbf{u}^2 - \mathbf{u}^1)\,\mathrm{d}\Gamma - \frac{1}{\alpha}\int_{\Gamma_I} \delta\boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{\lambda}\,\mathrm{d}\Gamma = 0 \tag{11.22}$$

in which $\alpha$ is a large (penalty) parameter. Inserting the approximation (11.16) into (11.15) and (11.22) results in the form

$$\begin{bmatrix} \mathbf{K}^1 & \mathbf{0} & \mathbf{Q}^1 \\ \mathbf{0} & \mathbf{K}^2 & \mathbf{Q}^2 \\ \mathbf{Q}^{1\mathrm{T}} & \mathbf{Q}^{2\mathrm{T}} & -\frac{1}{\alpha}\mathbf{V} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}}^1 \\ \tilde{\mathbf{u}}^2 \\ \tilde{\boldsymbol{\lambda}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}^1 \\ \mathbf{f}^2 \\ \mathbf{0} \end{Bmatrix} \tag{11.23}$$

where in addition to the arrays defined in Eq. (11.18)

$$\mathbf{V} = \int_{\Gamma_I} \mathbf{N}_\lambda^{\mathrm{T}}\mathbf{N}_\lambda\,\mathrm{d}\Gamma \tag{11.24}$$

Clearly, as the parameter $\alpha$ tends to infinity the result becomes identical to the Lagrange multiplier form. Such approximation thus behaves as a *penalty* type form. Formally, we can eliminate the Lagrange multiplier parameters from (11.23) to obtain

$$\begin{bmatrix} (\mathbf{K}^1 + \alpha\mathbf{Q}^1\mathbf{V}^{-1}\mathbf{Q}^{1\mathrm{T}}) & \alpha\mathbf{Q}^1\mathbf{V}^{-1}\mathbf{Q}^{2\mathrm{T}} \\ \alpha\mathbf{Q}^2\mathbf{V}^{-1}\mathbf{Q}^{1\mathrm{T}} & (\mathbf{K}^2 + \alpha\mathbf{Q}^2\mathbf{V}^{-1}\mathbf{Q}^{2\mathrm{T}}) \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}}^1 \\ \tilde{\mathbf{u}}^2 \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}^1 \\ \mathbf{f}^2 \end{Bmatrix} \tag{11.25}$$

which we recognize as a *penalty* type form

$$\left[\mathbf{K} + \alpha\mathbf{K}_2\right]\tilde{\mathbf{u}} = \mathbf{f}$$

An alternative to the above solves (11.22) *for each point on the boundary* $\Gamma_I$ yielding

$$\boldsymbol{\lambda} = \alpha\left(\mathbf{u}^2 - \mathbf{u}^1\right) \tag{11.26}$$

Substituting this into (11.15) then gives

$$\begin{aligned} &\int_{\Omega^1} \delta(\mathcal{S}\mathbf{u}^1)^{\mathrm{T}}\mathbf{D}^1\,\mathcal{S}\mathbf{u}^1\,\mathrm{d}\Omega - \alpha\int_{\Gamma_I} \delta\mathbf{u}^{1\mathrm{T}}(\mathbf{u}^2 - \mathbf{u}^1)\,\mathrm{d}\Gamma - \int_{\Omega^1} \delta\mathbf{u}^{1\mathrm{T}}\mathbf{b}^1\,\mathrm{d}\Omega \\ &\quad - \int_{\Gamma_t^1} \delta\mathbf{u}^{1\mathrm{T}}\bar{\mathbf{t}}^1\,\mathrm{d}\Gamma = 0 \end{aligned} \tag{11.27}$$

and

$$\begin{aligned} &\int_{\Omega^2} \delta(\mathcal{S}\mathbf{u}^2)^{\mathrm{T}}\mathbf{D}^2\,\mathcal{S}\mathbf{u}^2\,\mathrm{d}\Omega + \alpha\int_{\Gamma_I} \delta\mathbf{u}^{2\mathrm{T}}(\mathbf{u}^2 - \mathbf{u}^1)\,\mathrm{d}\Gamma - \int_{\Omega^2} \delta\mathbf{u}^{2\mathrm{T}}\mathbf{b}^2\,\mathrm{d}\Omega \\ &\quad - \int_{\Gamma_t^2} \delta\mathbf{u}^{2\mathrm{T}}\bar{\mathbf{t}}^2\,\mathrm{d}\Gamma = 0 \end{aligned} \tag{11.28}$$

Introducing now the approximations for $\mathbf{u}^1$ and $\mathbf{u}^2$ produces the penalty form

$$\begin{bmatrix} (\mathbf{K}^1 + \alpha \mathbf{K}_2^{11}) & -\alpha \mathbf{K}_2^{12} \\ -\alpha \mathbf{K}_2^{21} & (\mathbf{K}^2 + \alpha \mathbf{K}_2^{22}) \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}}^1 \\ \tilde{\mathbf{u}}^2 \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}^1 \\ \mathbf{f}^2 \end{Bmatrix} \tag{11.29}$$

where

$$\mathbf{K}_2^{ij} = \int_{\Gamma_I} \mathbf{N}_i^{\mathrm{T}} \mathbf{N}_j \, \mathrm{d}\Gamma, \quad i, j = 1, 2 \tag{11.30}$$

which we again recognize as a penalty form.

The differences between the penalty form (11.25) and that of (11.29) are significant:

1. The form given by (11.25) will not exhibit *locking* provided the choice for the $\mathbf{N}_\lambda$ satisfy the conditions for the mixed patch test.
2. The form given by (11.29) and (11.30) usually requires use of *reduced quadrature* on $\mathbf{K}_2^{ij}$ in order to avoid locking for reasons we discussed in Chapter 10.
3. Using standard or dual mortar methods the form (11.25) satisfies consistency conditions (e.g., constant stress) across the interface. [13] Generally, the form (11.29) does not transmit a constant stress condition correctly at the interface unless perfect matching of meshes occurs on $\Gamma_I$.

The above remarks clearly favor the form (11.25); however, this form requires the inversion of the matrix $\mathbf{V}$ (or a solution process equivalent to such inversion) and this can present difficulties. For the dual mortar method discussed in Section 11.2.1.2, the Lagrange multiplier can be eliminated by a perturbed Lagrangian approach using the discretized form of (11.22) approximated as

$$\delta \boldsymbol{\lambda}^{\mathrm{T}} \left[ \mathbf{Q}^2 - \widehat{\mathbf{Q}}^1 \right] - \frac{1}{\alpha} \widehat{\mathbf{Q}}^{1\mathrm{T}} \widehat{\mathbf{Q}}^1 \boldsymbol{\lambda} = \mathbf{0} \tag{11.31}$$

In this case the matrix to be inverted is also diagonal and the Lagrange multiplier may be locally eliminated to give a penalty form.

### 11.3.1 Nitsche method and discontinuous Galerkin approximation

An alternative to Lagrange multiplier and penalty methods for including the Dirichlet boundary condition was introduced by Nitsche [14]. Here we consider the procedure to include the condition $\phi = \bar{\phi}$ in the weak form of the quasi-harmonic equation. We first add together Eqs. (11.7) and (11.8) to obtain

$$\begin{aligned} &\int_\Omega \left[ (\nabla \delta\phi)^{\mathrm{T}} (\mathbf{k} \nabla \phi) + \delta\phi \, Q \right] \mathrm{d}\Omega + \int_{\Gamma_q} \delta\phi \left( \bar{q} + H \, \phi \right) \mathrm{d}\Gamma + \int_{\Gamma_\phi} \delta\phi \, \lambda \, \mathrm{d}\Gamma \\ &+ \int_{\Gamma_\phi} \delta\lambda \left( \phi - \bar{\phi} \right) \mathrm{d}\Gamma = 0 \end{aligned} \tag{11.32}$$

The normal flux $q_n$ on the boundary part $\Gamma_\phi$ is replaced by

$$\lambda = q_n(\phi) = -\mathbf{n}^{\mathrm{T}}(\mathbf{k}\nabla\phi) \qquad \delta\lambda = q_n(\delta\phi) = -\mathbf{n}^{\mathrm{T}}(\mathbf{k}\nabla\delta\phi)$$

thus eliminating the appearance of the Lagrange multiplier and giving a weak form expressed entirely in terms of $\phi$. One now can note that these two terms on $\Gamma_\phi$ can be zero for $\phi$ and $\delta\phi$ having constant values. Thus, to make the method *stable* Nitsche adds a penalty-like term

$$\int_{\Gamma_\phi} \delta\phi\, \alpha\, \left(\phi - \bar{\phi}\right)\, \mathrm{d}\Gamma$$

However, it is not required that $\alpha$ be large to ensure a good satisfaction of the boundary condition. The value recommended by Nitsche for linear elements is

$$\alpha = c\,\frac{|\mathbf{k}|}{h}, \quad c = O(10)$$

where $h$ is an element size and $|\mathbf{k}|$ a norm of the diffusion matrix.

The above steps give the weak form

$$\begin{aligned}\int_\Omega \left[\left(\nabla\delta\phi\right)^{\mathrm{T}}\left(\mathbf{k}\,\nabla\phi\right) + \delta\phi\, Q\right]\,\mathrm{d}\Omega + \int_{\Gamma_q} \delta\phi\,\left(\bar{q} + H\,\phi\right)\,\mathrm{d}\Gamma \\ + \int_{\Gamma_\phi} \delta\phi\, q_n(\phi)\,\mathrm{d}\Gamma + \underbrace{\int_{\Gamma_\phi} q_n(\delta\phi)\,[\phi - \bar{\phi}]\,\mathrm{d}\Gamma}_{\text{Symmetry}} + \underbrace{\int_{\Gamma_\phi} \delta\phi\,\alpha\,[\phi - \bar{\phi}]\,\mathrm{d}\Gamma}_{\text{Stability}} = 0\end{aligned} \tag{11.33}$$

Substituting the approximation for $\phi$ given by Eq. (11.11) into (11.33) gives

$$\mathbf{H}\,\tilde{\boldsymbol{\phi}} = \mathbf{f} \tag{11.34}$$

where

$$\begin{aligned}\mathbf{H} = &\int_\Omega (\nabla\mathbf{N})^{\mathrm{T}}\mathbf{k}\,(\nabla\mathbf{N})\,\mathrm{d}\Omega + \int_{\Gamma_q} \mathbf{N}^{\mathrm{T}} H\,\mathbf{N}\,\mathrm{d}\Gamma \\ &- \int_{\Gamma_\phi} \mathbf{N}^{\mathrm{T}}\left(\mathbf{n}^{\mathrm{T}}[\mathbf{k}(\nabla\mathbf{N})]\right)\,\mathrm{d}\Gamma - \int_{\Gamma_\phi} \left(\mathbf{n}^{\mathrm{T}}[\mathbf{k}(\nabla\mathbf{N})]\right)^{\mathrm{T}}\mathbf{N}\,\mathrm{d}\Gamma + \int_{\Gamma_\phi} \mathbf{N}^{\mathrm{T}}\alpha\,\mathbf{N}\,\mathrm{d}\Gamma \\ \mathbf{f} = &-\int_\Omega \mathbf{N}^{\mathrm{T}}\,Q\,\mathrm{d}\Omega - \int_{\Gamma_q} \mathbf{N}^{\mathrm{T}}\bar{q}\,\mathrm{d}\Gamma - \int_{\Gamma_\phi} \left(\mathbf{n}^{\mathrm{T}}[\mathbf{k}(\nabla\mathbf{N})]\right)^{\mathrm{T}}\bar{\phi}\,\mathrm{d}\Gamma + \int_{\Gamma_\phi} \mathbf{N}^{\mathrm{T}}\alpha\,\bar{\phi}\,\mathrm{d}\Gamma\end{aligned}$$

The Nitsche method results in a form in terms of the original primary variables of the problem. We can easily extend this to consider the connection of multiple subdomains.

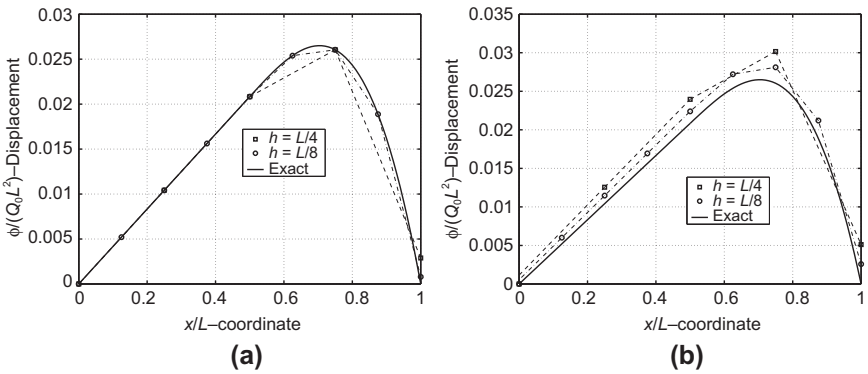



**FIGURE 11.8**

Solution of one-dimensional heat equation of Example 3.11 in Chapter 3: (a) Nitsche method ($c = 10$) and (b) penalty method: ($c = 10$).

**Example 11.2.** Dirichlet boundary condition

To indicate the performance of the Nitsche method in satisfaction of the Dirichlet boundary condition, we consider the one-dimensional problem given in Chapter 3 as Example 3.11. There the differential equation was given as

$$-\frac{\mathrm{d}^2\phi}{\mathrm{d}x^2} + Q(x) = 0, \quad 0 < x < L$$

with boundary conditions $\phi(0) = \phi(L) = 0$. We shall consider two domains: $\Omega^1$ for $0 \leq x < L/2$ and $\Omega^2$ for $L/2 < x \leq L$. The loading on $\Omega^2$ is a linear continuous function and that on $\Omega^1$ is zero. Using the Lagrange multiplier solution for this problem results in exact satisfaction of the boundary conditions $\phi(0) = \phi(L) = 0$, and, consequently, the same solution as given for the standard finite element solution in Chapter 3 (Fig. 3.10). Using the Nitsche method with $c = 10$ and $h = L/4$ and $L/8$ ($k = 1$) gives the solution shown in Fig. 11.8a. For comparison we drop the terms on the boundary with $q(\phi)$ and $q(\delta\phi)$ (i.e., use the penalty form alone) but keep the same value for $c$. This solution is shown in Fig. 11.8b. Of course, increasing the size of $c$ with either approach will improve the satisfaction of the boundary condition—but with an increased sensitivity in equation solution. The overall improvement of the Nitsche method is clearly evident and is accomplished without an increase in equation condition number. The results using quadratic elements are even better as shown in Fig. 11.9.

#### 11.3.1.1 *Multiple subdomain problems*

We again return to the problem of connecting two subdomains defined in $\Omega^1$ and $\Omega^2$ in which the common interface is $\Gamma_I$. The weak form of the problem may be

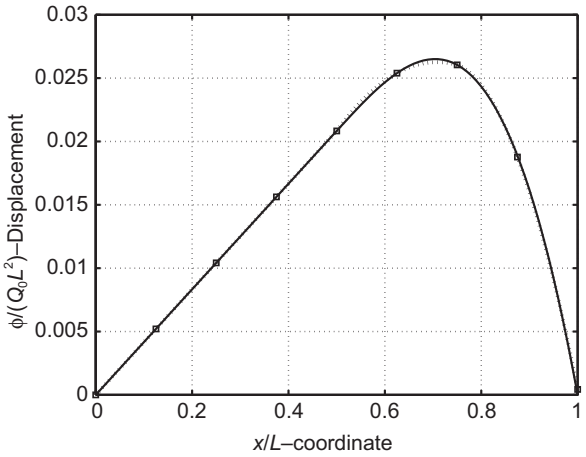



**FIGURE 11.9**
Solution of one-dimensional heat equation of Example 3.11 in Chapter 3. Quadratic elements.

written now as

$$G = \sum_{i=1}^{2} \left\{ \int_{\Omega^i} \left[ \left( \nabla \delta\phi^i \right)^{\mathrm{T}} \left( \mathbf{k}^i \nabla \phi^i \right) + \delta\phi^i \, Q^i \right] \mathrm{d}\Omega + \int_{\Gamma_q^i} \delta\phi^i \left( \bar{q}^i + H^i \phi^i \right) \mathrm{d}\Gamma \right\}$$
$$+ \int_{\Gamma_I} \left[ \delta\phi^1 - \delta\phi^2 \right] q_n \left( \phi^1, \phi^2 \right) \mathrm{d}\Gamma + \int_{\Gamma_I} q_n \left( \delta\phi^1, \delta\phi^2 \right) \left[ \phi^1 - \phi^2 \right] \mathrm{d}\Gamma$$
$$+ \int_{\Gamma_I} \left[ \delta\phi^1 - \delta\phi^2 \right] \alpha \left[ \phi^1 - \phi^2 \right] \mathrm{d}\Gamma = 0 \tag{11.35}$$

which results from adding Eqs. (11.1) and (11.3) and setting the Lagrange multiplier to

$$\lambda = q_n(\phi^1, \phi^2) \quad \text{and} \quad \delta\lambda = q_n(\delta\phi^1, \delta\phi^2) \tag{11.36}$$

which now becomes a function of the flux from both sides of the interface.

This form is an extension of the concept of Nitsche and, of course, can be effectively used to consider multiple subdomains in an obvious manner. When extended to the case where each element becomes a subdomain, the problem assumes a form known as the *discontinuous Galerkin method* [15–19].

The discontinuous Galerkin method was first introduced by Reed & Hill [20] for analysis of neutron transport problems. It was analyzed by Lesaint & Raviart for its mathematical properties [21]. As shown in the paper by Zienkiewicz et al. the method is most effective in problems which have significant convection effects—and is less accurate than standard (continuous) finite elements for problems which possess only diffusion effects [22]. Here we are interested in the method primarily for connecting

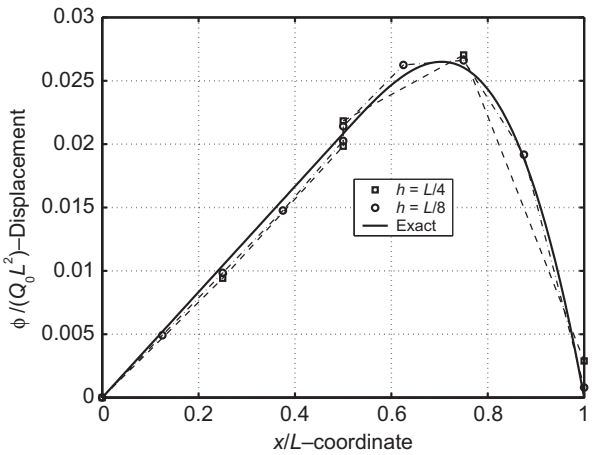



**FIGURE 11.10**

Two subdomain solution using Nitsche method for one-dimensional heat equation of Example 3.11 in Chapter 3. Linear elements.

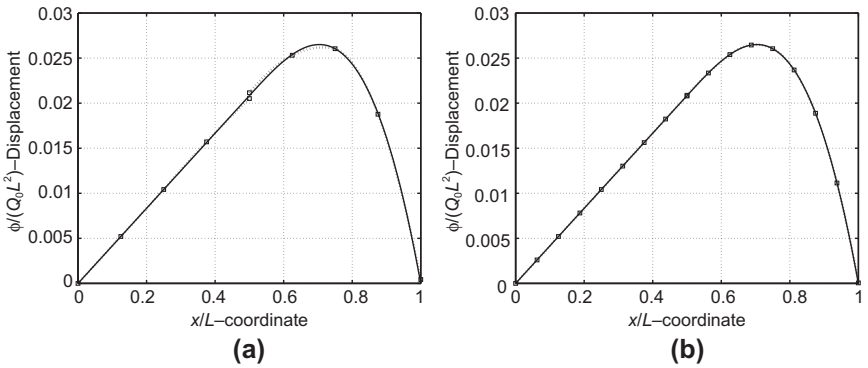



**FIGURE 11.11**

Two subdomain solution using Nitsche method for one-dimensional heat equation of Example 3.11 in Chapter 3. Quadratic elements: (a) four elements ($c = 10$) and (b) eight elements ($c = 10$).

subdomains which contain either large number of standard elements or have high order expansions with significant number of parameters not associated with the boundary.

**Example 11.3.** Two domain problem

To indicate the performance in the presence of multiple domains we again consider the one-dimensional form of Example 11.2. We shall consider two domains: $\Omega^1$ for $0 \leq x < L/2$ and $\Omega^2$ for $L/2 < x \leq L$. The loading on $\Omega^2$ is a linear continuous

function and that on $\Omega^1$ is zero. The Nitsche method is used with four and eight elements (2 and 4 in each subdomain, respectively) and a value of $c = 10$. The solution is shown in Fig. 11.10 and again indicates quite rapid convergence with increased number of elements. We also show results for the same problem with quadratic elements (Fig. 11.11), in which no discernible jump exists for the eight-elements case.

## 11.4 Problems

**11.1** Compute explicit relations for linear one-dimensional dual shape functions using Eq. (11.13). Verify the results shown in Fig. 11.5.

**11.2** Compute explicit relations for quadratic one-dimensional dual shape functions using Eq. (11.13). Assume the element side is straight and the interior node is at the center of the edge.

**11.3** Compute an explicit relation at node $a$ for four-node dual shape functions. Use Eq. (11.13) and assume the surface mesh for elements is as shown in Fig. 11.12. Sketch the shape of the global dual function at node $a$ (e.g., as shown for a 2-D edge in Fig. 11.4).

**11.4** The mesh segment shown in Fig. 11.13 occurs in a problem in which the two sides are to be joined using a standard mortar method. If node $a$ is located at $0.4\,h$ from node $b$ perform the integrals necessary to construct the contributions to the $\mathbf{Q}_i$ arrays appearing in Eq. (11.12).

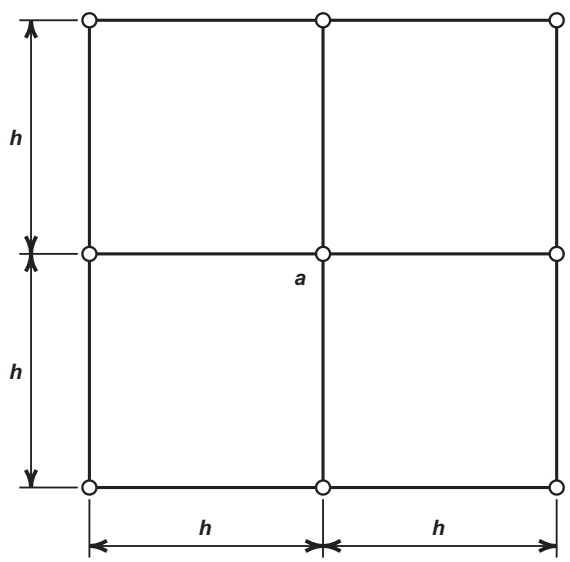



**FIGURE 11.12**
Surface description for Problem 11.3.

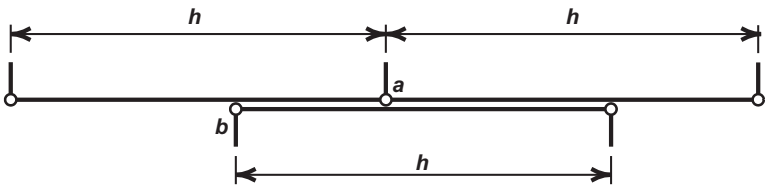



**FIGURE 11.13**
Tied segment for Problems 11.4 and 11.5.

**11.5** The mesh segment shown in Fig. 11.13 occurs in a problem in which the two sides are to be joined using a dual mortar method. If node $a$ is located at $0.4\,h$ from node $b$ perform the integrals necessary to construct the contributions to the $\mathbf{Q}_i$ arrays appearing in Eq. (11.12). [Note: It is necessary to replace one $N$ by $\widehat{N}$ for the dual approach.]

**11.6** Write a MATLAB (or GNU Octave) program to solve the one-dimensional problem of Example 3.11 in Chapter 3. Modify the program to enforce the boundary conditions using the Nitsche method described in Section 11.3.1. Verify your program by solving the example illustrated in Fig. 11.8a.

## References

[1] C. Farhat, J. Mandel, The two-level FETI method. I. An optimal iterative solver for biharmonic systems, Comput. Methods Appl. Mech. Eng. 155 (1998) 129–151.

[2] C. Farhat, Chen Po-shu, J. Mandel, F.X. Roux, The two-level FETI method. II. Extension to shell problems, parallel implementation and performance results, Comput. Methods Appl. Mech. Eng. 155 (1998) 153–179.

[3] C. Farhat, K. Pierson, M. Lesoinne, The second generation FETI methods and their application to the parallel solution of large-scale linear and geometrically non-linear structural analysis problems, Comput. Methods Appl. Mech. Eng. 184 (2000) 333–374.

[4] G. Karypis, METIS: Family of multilevel partitioning algorithms, <http://www-users.ce.umn.edu/~karypis/metis/>.

[5] G. Karypis, ParMETIS parallel graph partitioning, <http://www-users.cs.unm.edu/karypis/metis/parmetis/>, 2003.

[6] PETSc: Portable, Extensible Toolkit for Scientific Computation, <http://www.mcs.anl.gov/petsc>.

[7] Parallel computing, Wikipedia.

[8] O.C. Zienkiewicz, R.L. Taylor, D. Fox, The Finite Element Method for Solid and Structural Mechanics, seventh ed., Elsevier, Oxford, 2013.

[9] N.-E. Wiberg, Matrix structural analysis with mixed variables, Int. J. Numeric. Methods Eng. 8 (1974) 167–194.

[10] B.I. Wohlmuth, Discretization Methods and Iterative Solvers Based on Domain Decomposition, Springer-Verlag, Heidelberg, 2001.
[11] G. Kron, Tensor Analysis of Networks, John Wiley & Sons, New York, 1939.
[12] Ch Farhat, F.-X. Roux, A method of finite element tearing and interconnecting and its parallel solution algorithm, Int. J. Numeric. Methods Eng. 32 (1991) 1205–1227.
[13] M.A. Puso, T.A. Laursen, Mesh tying on curved interfaces in 3D, Eng. Comput. 20 (2003) 305–319.
[14] J.A. Nitsche, Über ein Variationsprinzip zur Lösung Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen uneworfen sind, Abh. Math. Sem. Univ. Hamburg 36 (1971) 9–15.
[15] C.G. Makridakis, I. Babuška, On the stability of the discontinuous Galerkin method for the heat equation, SIAM J. Num. Anal. 34 (1997) 389–401.
[16] J.T. Oden, I. Babuška, C.E. Baumann, A discontinuous *hp* finite element method for diffusion problems, J. Comp. Physics 146 (2) (1998) 491–519.
[17] T.J.R. Hughes, G. Engel, L. Mazzei, M.G. Larson, A comparison of discontinuous and continuous Galerkin methods based on error estimates, conservation, robustness and efficiency, in: Discontinuous Galerkin Methods: Theory, Computation and Applications, Springer-Verlag, Berlin, 2000, pp. 135–146.
[18] B. Cockburn, G.E. Karniadakis, Chi-Wang Shu. Discontinuous Galerkin Methods: Theory, Computation and Applications. Springer-Verlag, Berlin, 2000.
[19] D.N. Arnold, F. Brezzi, B. Cockburn, D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, SIAM J. Numer. Anal. 39 (2002) 1749–1779.
[20] W.H. Reed, T.R. Hill, Triangular Mesh Methods for the Neutron Transport Equation, Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.
[21] P. Lesaint, P.-A. Raviart, On a finite element method for solving the neutron transport equation, in: C. de Boor (Ed.), Mathematical Aspects of Finite Elements in Partial Differential Equations, Academic Press, New York, 1974.
[22] O.C. Zienkiewicz, R.L. Taylor, S.J. Sherwin, J. Peiró, On discontinuous Galerkin methods, Int. J. Numeric. Methods Eng. 58 (2003) 1119–1148.

CHAPTER

# 12 The Time Dimension: Semi-Discretization of Field and Dynamic Problems

## 12.1 Introduction

In most of the problems considered so far in this text conditions that do not vary with time were generally assumed. There is little difficulty in extending the finite element idealization to situations that are time dependent as indicated in Chapters 3, 5, and 7.

The range of practical problems in which the time dimension has to be considered is great. Transient heat conduction, wave transmission in fluids, and dynamic behavior of structures are typical examples. While it is usual to consider these various problems separately—sometimes classifying them according to the mathematical structure of the governing equations as "parabolic" or "hyperbolic" [1]—we shall group them into one category to show that the formulation is identical.

In the first part of this chapter we shall formulate, by a simple extension of the methods used so far, matrix differential equations governing such problems for a variety of physical situations. Here a finite element discretization in the space dimension only will be used and a semi-discretization process followed (see Chapters 3 and 5). In the remainder of this chapter various analytical procedures of the solution for the resulting ordinary linear differential equation system will be dealt with. These form the basic arsenal of steady-state and transient analysis.

## 12.2 Direct formulation of time-dependent problems with spatial finite element subdivision

### 12.2.1 The "quasi-harmonic" equation with first and second time derivative

In Chapters 2 and 5 we considered the quasi-harmonic equation with a first time derivative. Here we extend the problem to include both first and second derivatives. In the three-dimensional case typically we might have

$$\frac{\partial}{\partial x}\left(k\frac{\partial \phi}{\partial x}\right)+\frac{\partial}{\partial y}\left(k\frac{\partial \phi}{\partial y}\right)+\frac{\partial}{\partial z}\left(k\frac{\partial \phi}{\partial z}\right)+Q=c\frac{\partial \phi}{\partial t}+\rho\frac{\partial^2 \phi}{\partial t^2} \quad (12.1)$$

where the added term is associated with the parameter $\rho$. In the above, quite generally, all the parameters may be prescribed functions of space $\mathbf{x}$ and some also of time, i.e.,

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00012-5

$$k \quad k(\mathbf{x}), \quad Q \quad Q(\mathbf{x},t) \quad \text{etc.} \tag{12.2}$$

Writing (12.1) as a weak form and introducing the finite element approximations

$$\phi \qquad N_a\phi_a \quad \mathbf{Nu} \quad \text{with} \quad \mathbf{N} \quad \mathbf{N}(x,\ y,\ z) \quad \text{and} \quad \mathbf{u} \quad \mathbf{u}(t) \tag{12.3}$$

for each element yields the semi-discrete form

$$\begin{gathered} \mathbf{Mu} \quad \mathbf{Cu} \quad \mathbf{Ku} \quad \mathbf{f} \quad \mathbf{0} \\ \mathbf{u} \quad \frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} \quad \text{and} \quad \mathbf{u} \quad \frac{\mathrm{d}^2\mathbf{u}}{\mathrm{d}t^2} \end{gathered} \tag{12.4}$$

in which all the matrices are assembled from element submatrices in the standard manner and we have replaced by **u**, **H** by **K**, and **s** by **f** for consistency with the elasticity problem consider later in this chapter. The reader can verify that the element matrices in (12.4) are given by

$$\begin{aligned} \mathbf{K}^e \quad \mathbf{H}^e \quad & \int_{\Omega_e} \mathbf{b}^T\mathbf{k}\mathbf{b}\,\mathrm{d}\Omega \qquad \int_{\Gamma_{qe}} \mathbf{N}^T H\mathbf{N}\,\mathrm{d}\Gamma \\ \mathbf{C}^e \quad & \int_{\Omega_e} \mathbf{N}^T c\mathbf{N}\,\mathrm{d}\Omega \\ \mathbf{M}^e \quad & \int_{\Omega_e} \mathbf{N}^T \rho\mathbf{N}\,\mathrm{d}\Omega \\ \mathbf{f}^e \quad \mathbf{s}^e \quad & \int_{\Omega_e} \mathbf{N}^{\mathrm{T}} Q\,\mathrm{d}\Omega \qquad \int_{\Gamma_{qe}} \mathbf{N}^T (q \quad H\phi_0)\,\mathrm{d}\Gamma \end{aligned} \tag{12.5}$$

Once again these matrices are symmetric as seen from the above relations. Boundary conditions imposed at any time instant are treated in the standard manner.

The variety of physical problems governed by Eq. (12.1) is so large that a comprehensive discussion of them is beyond the scope of this book. A few typical examples will, however, be quoted.

**Equation (12.1) with $\rho$ 0**
This is the standard *transient heat conduction equation* [1,2] which has been discussed in Chapter 5 and by several authors [3–6]. This same equation is applicable in other physical situations—one of these being the *soil consolidation equations* [7] associated with *transient seepage forms* [8].

**Equation (12.1) with $c$ 0**
Now the relationship becomes the famous *Helmholtz wave equation* governing a wide range of physical phenomena. Electromagnetic waves [9], fluid surface waves [10], and compression waves [11] are but a few cases to which the finite element process has been applied.

**Equation (12.1) with $c$ $\rho$ 0**
This damped wave equation is of yet more general applicability and has particular significance in fluid mechanics (wave) problems.

### 12.2.2 Dynamic behavior of elastic structures with linear damping

While in the previous section we have been concerned with, apparently, a purely mathematical problem, identical reasoning can be applied directly to the wide class of dynamic behavior of elastic structures following precisely the general lines of Chapters 3, 4, and 7.

When displacements of an elastic body vary with time two sets of additional forces are called into play. The first is the inertia force, which we included in developments in Chapter 7. A second force is that due to (frictional) resistances opposing the motion. These may be due to microstructure movements, air resistance, etc., and are often related in a nonlinear way to the velocity $\mathbf{u}$. For simplicity of treatment, however, only a linear viscous-type resistance will be considered, resulting in unit volume forces of magnitude $\mu\mathbf{u}$. In the above $\mu$ is a set of viscosity parameters which can presumably be given numerical values [12].

The transient problem may now be written in Cartesian coordinates as

$$\boldsymbol{\mathcal{S}}^{\mathrm{T}} \quad \mathbf{b} \quad \rho\mathbf{u} \quad \mu\mathbf{u} \tag{12.6}$$

from which a standard weak form may be constructed. Discretizing precisely in the manner of Chapters 3 and 7 gives the semi-discrete form

$$\mathbf{Mu} \quad \mathbf{Cu} \quad \mathbf{Ku} \quad \mathbf{f} \quad \mathbf{0} \tag{12.7}$$

in which $\mathbf{K}$, $\mathbf{M}$, and $\mathbf{f}$ are assembled stiffness, mass, and load matrices obtained by the usual addition of element matrices. The new matrix $\mathbf{C}$ is assembled by the usual rule from

$$\mathbf{C}^e_{ab} \qquad \int_{\Omega_e} \mathbf{N}_a^{\mathrm{T}} \ \mathbf{N}_b \, \mathrm{d}\Omega \tag{12.8}$$

which is known as the element consistent damping matrix. For completeness we recall the mass, stiffness, and load arrays are given by

$$\begin{aligned} \mathbf{M}^e_{ab} &\qquad \int_{\Omega_e} \mathbf{N}_a^{\mathrm{T}} \rho \mathbf{N}_b \, \mathrm{d}\Omega \\ \mathbf{K}^e_{ab} &\qquad \int_{\Omega_e} \mathbf{B}_a^{\mathrm{T}} \mathbf{D} \mathbf{B}_b \, \mathrm{d}\Omega \\ \mathbf{f}^e_a &\qquad \int_{\Omega_e} \mathbf{N}_a^{\mathrm{T}} \mathbf{b} \, \mathrm{d}\Omega \qquad \int_{\Gamma_t} \mathbf{N}_a^{\mathrm{T}} \mathbf{t} \, \mathrm{d}\Gamma \end{aligned} \tag{12.9}$$

Determination of the damping matrix $\mathbf{C}$ is in practice difficult as knowledge of the viscous matrix $\mu$ is lacking. It is often assumed, therefore, that the damping matrix is a linear combination of stiffness and mass matrices, i.e.,

$$\mathbf{C} \quad \alpha\mathbf{M} \quad \beta\mathbf{K} \tag{12.10}$$

Here the parameters $\alpha$ and $\beta$ are determined experimentally [12,13]. Such damping is known as "Rayleigh damping" and has certain mathematical advantages which we shall discuss later. On occasion $\mathbf{C}$ may be completely specified and such approximation devices are not necessary.

### 12.2.3 "Mass" or "damping" matrices for some typical elements

It is impractical to present in an explicit form all the mass or damping matrices for the various elements discussed in previous chapters. Some selected examples only will be discussed here.

**Example 12.1.** Plane stress and plane strain

Using triangular three-node elements the matrix $\mathbf{N}^e$ is defined as

$$\mathbf{N}^e \quad \mathbf{N}_1^e \ \mathbf{N}_2^e \ \mathbf{N}_3^e \quad \text{where } \mathbf{N}_a^e \quad N_a\mathbf{I}, \ a \quad 1, 2, 3 \quad \text{and } \mathbf{I} \quad \begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix}$$

Equation (5.19) gives the shape functions as

$$N_a \quad \frac{a_a \quad b_a x \quad c_a y}{2\Delta}, \quad a \quad 1, 2, 3$$

where $\Delta$ is the area of the triangular element.

If the thickness of the element is $h$ and this is assumed to be constant within the element, we have, for the mass matrix, the first equation in (12.9),

$$\mathbf{M}^e \quad \rho\, h \quad \mathbf{N}^{\mathrm{T}}\mathbf{N}\, \mathrm{d}x\, \mathrm{d}y \quad \text{or} \quad \mathbf{M}_{ab}^e \quad \rho\, h\, \mathbf{I} \quad N_a\, N_b\, \mathrm{d}x\, \mathrm{d}y$$

If the relationships of Eq. (5.19) are substituted, it is easy to verify that

$$N_a N_b\, \mathrm{d}x\, \mathrm{d}y \quad \begin{matrix} \frac{1}{6}\Delta & \text{when } a \quad b \\ \frac{1}{12}\Delta & \text{when } a \quad b \end{matrix} \tag{12.11}$$

Thus taking the total mass of the element as

$$m \quad \rho\, h\, \Delta$$

the (consistent) mass matrix becomes

$$\mathbf{M}^e = \frac{m}{12} \left[\begin{array}{cc|cc|cc} 2 & 0 & 1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 & 0 & 1 \\ \hline 1 & 0 & 2 & 0 & 1 & 0 \\ 0 & 1 & 0 & 2 & 0 & 1 \\ \hline 1 & 0 & 1 & 0 & 2 & 0 \\ 0 & 1 & 0 & 1 & 0 & 2 \end{array}\right] \tag{12.12}$$

If the mass is physically lumped at the nodes in three equal parts the "lumped" mass matrix contributed by the element is

$$\mathbf{M}^e = \frac{m}{3}\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{12.13}$$

Certainly both matrices differ considerably and yet in applications the results of the analysis can be almost identical.

A similar result holds for the element damping matrix $\mathbf{C}^e$.

**Example 12.2.** Damping matrix for isoparametric elements

The damping matrix for isoparametric elements may be computed by numerical integration as described in Chapter 6. For example, in two-dimensional elements the damping is given by

$$\mathbf{C}^e_{ab} \quad \sum_l N_a(\xi_l, \eta_l)\mu N_a(\xi_l, \eta_l)\, J_l\, W_l \tag{12.14}$$

for plane problems and

$$\mathbf{C}^e_{ab} \quad \sum_l N_a(\xi_l, \eta_l)\mu N_a(\xi_l, \eta_l)\, r_l\, J_l\, W_l \tag{12.15}$$

for axisymmetric problems. A similar construction occurs for the general three-dimensional problem.

Now since it is the shape functions which are integrated, for full rank of the damping matrix the order of quadrature needs to be selected according to the requirements given in Section 6.8.1.

### 12.2.4 Mass "lumping" or diagonalization

We have referred to the computational convenience of lumping of mass matrices and presenting these in diagonal form. On some occasions such lumping is physically obvious (see the linear triangle for instance), in others this is not the case and a "rational" procedure is required. For mass matrices of the type given in Eq. (12.9) several alternative approximations have been developed, as discussed in Appendix H. In all of these the essential requirement of mass preservation is satisfied, i.e.,

$$\sum_a M_{aa} \quad \int_\Omega \rho \, \mathrm{d}\Omega \tag{12.16}$$

where $M_{aa}$ is the diagonal for a component of the lumped mass matrix $\mathbf{M}$.

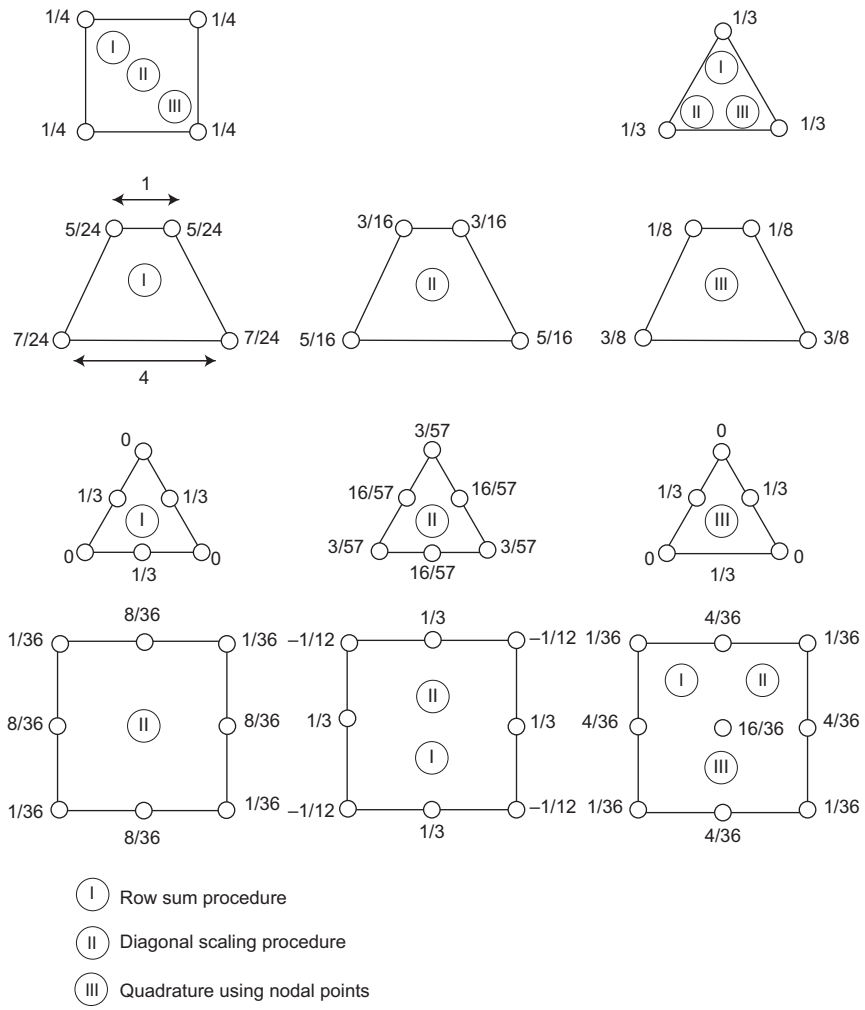



**FIGURE 12.1**
Mass lumping for some two-dimensional elements.

Three main procedures exist (see Fig. 12.1):

**1.** The row sum method in which

$$M_{aa} \qquad M_{ab}$$
$$b$$

**2.** Diagonal scaling in which

$$M_{aa} \quad \alpha\, M_{aa}$$

with $\alpha$ adjusted so that Eq. (12.16) is satisfied [14,15].

**3.** Evaluation of $M$ using a quadrature involving only the nodal points and thus automatically yielding a diagonal matrix for standard finite element shape functions [16,17] in which $N_a(\mathbf{x}_b) \quad \delta_{ab}$ where $\delta_{ab}$ is a Kronnecker delta.

It should be remarked that Eq. (12.16) does not hold for hierarchical shape functions.

The quadrature (numerical integration) process is mathematically most appealing but frequently leads to negative or zero lumped masses. Such a loss of positive definiteness is undesirable in some solution processes and cancels out the advantages of lumping. In Fig. 12.1 we show the effect of various lumping procedures on triangular and quadrilateral elements of linear and quadratic type. It is clear from these that the optimal choice to lump the mass is by no means unique.

In general we would recommend the use of lumped matrices only as a convenient numerical device generally paid for by some loss of accuracy. An exception to this is for "explicit" time integration of dynamics problems where the considerable efficiency of their use more than compensates for any loss in accuracy. However, we note that it has occasionally been shown that lumping can *improve* accuracy of some problems by error cancellation. It can be shown that in the transient approximation the lumping process introduces additional dissipation of the "stiffness matrix" form and this can help in canceling out numerical oscillation.

To demonstrate the nature of lumped and consistent mass matrices it is convenient to consider a typical one-dimensional problem specified by the equation

$$\frac{\partial \phi}{\partial t} \quad \frac{\partial}{\partial x} \quad \mu \frac{\partial}{\partial x} \quad \frac{\partial \phi}{\partial t} \qquad \frac{\partial}{\partial x} \quad k \frac{\partial \phi}{\partial x} \qquad 0$$

Semi-discretization here gives a typical nodal equation $a$ as

$$(M_{ab} \quad H_{ab})u_b \quad K_{ab}u_b \quad 0$$

where

$$M_{ab} \quad \int_\Omega N_a N_b \, \mathrm{d}x, \quad H_{ab} \quad \int_\Omega \frac{\mathrm{d}N_a}{\mathrm{d}x} \mu \frac{\mathrm{d}N_b}{\mathrm{d}x} \, \mathrm{d}x,$$

$$K_{ab} \quad \int_\Omega \frac{\mathrm{d}N_a}{\mathrm{d}x} k \frac{\mathrm{d}N_b}{\mathrm{d}x} \, \mathrm{d}x$$

and it is observed that **H** and **K** have identical structure. With linear elements of constant size $h$ the approximating equation at a typical node $a$ (and surrounding

nodes $a$ 1 and $a$ 1) can be written as follows:

$$M_{ab}u_b \quad \frac{h}{6}\ u_{a\ 1} \quad 4u_a \quad u_{a\ 1}$$
$$H_{ab}u_b \quad \frac{\mu}{h} \quad u_{a\ 1} \quad 2u_a \quad u_{a\ 1}$$
$$K_{ab}u_b \quad \frac{k}{h} \quad u_{a\ 1} \quad 2u_a \quad u_{a\ 1}$$

If a lumped approximation is used for **M**, that is **M**, we have, simply by adding coefficients using the row sum method,

$$M_{ab}u_b \quad hu_a$$

The difference between the two expressions is

$$M_{ab}u_b \quad M_{ab}u_b \quad \frac{h}{6}(\ u_{a\ 1} \quad 2u_a \quad u_{a\ 1})$$

and is clearly identical to that which would be obtained by increasing $\mu$ by $h^2/6$. As $\mu$ in the above example can be considered as a viscous dissipation we note that the effect of using a lumped matrix is that of adding an extra amount of such viscosity and can often result in smoother (though possibly less accurate) solutions.

## 12.3 Analytical solution procedures: General classification

We have seen that as a result of semi-discretization many time-dependent problems can be reduced to a system of ordinary differential equations of the characteristic form given by

$$\mathbf{Mu} \quad \mathbf{Cu} \quad \mathbf{Ku} \quad \mathbf{f} \quad \mathbf{0} \tag{12.17}$$

In this, in general, all the matrices are symmetric. Cases involving nonsymmetric matrices are also found in some fluid problems [18]. This second-order system often becomes first order if **M** is zero as, for instance, in transient heat conduction problems. We shall now discuss some methods of solution of such ordinary differential equation systems.

Systems of ordinary linear differential equations can always in principle be solved analytically without the introduction of additional approximations. While such solutions are possible they may be so complex that discrete time solution as described in Chapter 3 is required. The analytical approach provides, however, an insight into the behavior of the system which the authors always find helpful.

Some of the matter in this chapter will be an extension of standard well-known procedures used for the solution of constant coefficient differential equations that are encountered in most studies of dynamics or mathematics. In the following we shall deal successively with:

1. Determination of free response ($\mathbf{f} \quad \mathbf{0}$)
2. Determination of steady-state periodic response ($\mathbf{f}(t)$ periodic)
3. Determination of transient response ($\mathbf{f}(t)$ arbitrary)

In the first two, initial conditions of the system are not required and a general solution is simply sought. The transient response initial conditions are required and we will devote considerable attention to this type in Section 12.8.

## 12.4 Free response: Eigenvalues for second-order problems and dynamic vibration

### 12.4.1 Free dynamic vibration: Real eigenvalues

If no damping or forcing terms exist in the dynamic problem of Eq. (12.17) it reduces to

$$\mathbf{Mu} \quad \mathbf{Ku} \quad \mathbf{0} \tag{12.18}$$

A general solution of such an equation may be written as

$$\mathbf{u} \quad \mathbf{u}\exp(\mathrm{i}\omega t)$$

the real part of which simply represents a harmonic response as $\exp(\mathrm{i}\omega t) \quad \cos\omega t$ $\mathrm{i}\sin\omega t$. Then on substitution we find that $\omega$ can be determined from

$$\mathbf{K}\,\mathbf{U} \quad \mathbf{MU} \tag{12.19}$$

This is the *general linear eigenvalue* or *characteristic value problem* that was discussed in Section 3.8.2 and yields $n$ positive values of $\omega^2$ (or $\omega_j, \quad j \quad 1, 2, \ldots, n$) with associated eigenvectors $\mathbf{u}_j$. These are known as the natural frequencies and normal modes of the system.

### 12.4.2 Determination of eigenvalues

To find the actual eigenvalues and eigenvectors it is necessary to solve the general eigenproblem. Many extremely efficient procedures are available and the reader can find some interesting matter in Refs. [19–26].

In some processes the starting point is the *standard eigenvalue problem* given by

$$\mathbf{AV} \quad \mathbf{V} \quad , \ \mathbf{V} \quad \mathbf{v}_1, \mathbf{v}_2 \ldots \mathbf{v}_n \tag{12.20}$$

in which $\mathbf{A}$ is a symmetric matrix and hence has real eigenvalues and vectors $\mathbf{V}$. Equation (12.19) can be written as the standard eigenproblem

$$\mathbf{M}\ {}^{1}\mathbf{KU} \quad \mathbf{U} \tag{12.21}$$

on inverting $\mathbf{M}$, but symmetry is in general lost.

If, however, we write in triangular form (i.e., the Cholesky factors)

$$\mathbf{M} \quad \mathbf{L}\mathbf{L}^{\mathrm{T}} \quad \text{and} \quad \mathbf{M}^{-1} \quad \mathbf{L}^{-\mathrm{T}}\mathbf{L}^{-1}$$

in which $\mathbf{L}$ is a lower triangular matrix (i.e., has all zero coefficients above the diagonal), Eq. (12.19) may now be written as

$$\mathbf{K}\mathbf{U} \quad \mathbf{L}\mathbf{L}^{\mathrm{T}}\mathbf{U}$$

Calling

$$\mathbf{L}^{\mathrm{T}}\mathbf{U} \quad \mathbf{V} \tag{12.22}$$

and multiplying by $\mathbf{L}^{-1}$ we have finally

$$\mathbf{A}\mathbf{V} \quad \mathbf{V} \tag{12.23}$$

in which

$$\mathbf{A} \quad \mathbf{L}^{-1}\mathbf{K}\mathbf{L}^{-\mathrm{T}} \tag{12.24}$$

which is of the standard form of Eq. (12.20), as $\mathbf{A}$ is now symmetric.

Having determined $\omega_i^2$ (all, or only a few of the selected smallest values corresponding to fundamental periods) the modes of $\mathbf{V}$ are found, and hence by use of Eq. (12.22) the modes of $\mathbf{U}$.

If the matrix $\mathbf{M}$ is diagonal—as it will be if the masses have been "lumped"—the procedure of deriving the standard eigenvalue problem is simplified and here appears the first advantage of the diagonalization, which we have discussed in Section 12.2.4.

### 12.4.3 Free vibration with the singular $\mathbf{K}$ matrix

In static problems we have always introduced a suitable number of *support* conditions to allow the stiffness matrix $\mathbf{K}$ to be inverted, or what is equivalent to solve the static equations uniquely. If such "support" conditions are in fact not specified, as may well be the case with a rocket traveling in space, the arbitrary fixing of a minimum number of support conditions allows a static solution to be obtained without affecting the stresses. In dynamic situations such a fixing is not permissible and frequently one is faced with the problem of a free oscillation for which $\mathbf{K}$ is singular and therefore does not possess unique triangular factors or an inverse.

To preserve the applicability of methods which require an inverse (e.g., methods based on inverse power iteration) [24] a simple artifice is possible. Equation (12.19) is modified to

$$\mathbf{K} \quad \alpha\mathbf{M}\,\mathbf{U} \quad \mathbf{M}\mathbf{U}\,(\quad \alpha\mathbf{I}) \tag{12.25}$$

in which $\alpha$ is an arbitrary constant of the same order as the typical $\omega^2$ sought. The new matrix $(\mathbf{K} \quad \alpha\mathbf{M})$ is no longer singular and can be factored (or inverted) for use in the standard eigensolution procedure to find $(\omega_i^2 \quad \alpha)$ and hence $\omega_i^2$.

This simple but effective avoidance of an otherwise serious difficulty was first suggested by Cox [27] and Jennings [28]. Alternative methods of dealing with the above problem are given in Refs. [29,30].

### 12.4.4 Reduction of the eigenvalue system

Independent of which technique is used to determine the eigenpairs of the system (12.19), the effort for $n$ $n$ matrices is at least one order greater than that involved in an equivalent static situation. Further, while the number of eigenvalues of the real system is infinite, in practice, we are generally interested only in a relatively small number of the lower frequencies and it is possible to simplify the computation by reducing the size of the problem.

To achieve a reduced problem we assume that the unknown $\mathbf{u}$ can be expressed in terms of $m$ ( $n$) vectors $\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_m$ and corresponding participating factors $x_i$. We now write

$$\mathbf{u}_i \quad \mathbf{t}_1 x_{i1} \quad \mathbf{t}_2 x_{i2} \quad \mathbf{t}_m x_{im} \quad \mathbf{T}\mathbf{x}_i \tag{12.26}$$

and, thus, for the total problem

$$\mathbf{U} \quad \mathbf{T}\mathbf{X} \tag{12.27}$$

Inserting Eq. (12.27) into Eq. (12.19) and premultiplying by $\mathbf{T}^{\mathrm{T}}$ we have a reduced problem with only $m$ eigenpairs:

$$\mathbf{K}^{\star}\mathbf{X} \quad \mathbf{M}^{\star}\mathbf{X} \quad ^{\star} \tag{12.28}$$

where

$$\mathbf{M}^{\star} \quad \mathbf{T}^{\mathrm{T}}\mathbf{M}\mathbf{T} \quad \mathbf{K}^{\star} \quad \mathbf{T}^{\mathrm{T}}\mathbf{K}\mathbf{T}$$

and $\lambda_i^{\star}$ $(\omega_i^{\star})^2$ are now eigenvalues of the *reduced* system, which for the appropriate choice of the $\mathbf{t}_i$ vectors can be good approximations to the eigenvalues of the original system.

If by good fortune the trial vectors were to be chosen as eigenvectors of the original matrix the system would become diagonal and all eigenvalues (i.e., in this case $\omega$ $\omega$) could be determined by a trivial calculation. This indeed is what some iterative eigenproblem strategies achieve (e.g., subspace or Lanczos methods [24,26]). It is also of course possible by physical insight to find vectors $\mathbf{t}$ that correspond closely to the principal modes of the movement (e.g., see Ref. [31]).

### 12.4.5 Some examples

There are a variety of problems for which practical solutions exist, so only a few simple examples will be shown.

#### 12.4.5.1 *Vibration of a simply supported beam*

Figure 12.2 shows the first three vibration modes of a simply supported beam with length 40 and rectangular cross-section of width 1 and depth 2 units. The elastic properties are $E$ $30{,}000$, $\nu$ $0$, and $\rho$ $0.1$ units. The beam is modeled using nine-node quadrilateral elements of Lagrangian type. The central node at the left end is restrained in the $x$- and $y$-direction and the central node at the right end is restrained only in the $y$-direction. The problem is also solved using a mesh with 1000 two-node beam elements which include effects of transverse shearing deformation.

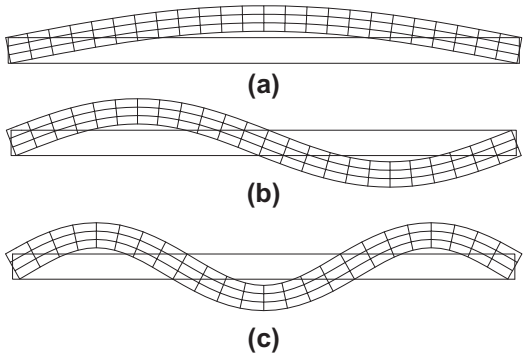



**FIGURE 12.2**
Simply supported beam: (a) $\omega_1$ 3.7785; (b) $\omega_2$ 59.2236; (c) $\omega_3$ 290.0804.

**Table 12.1** Frequencies for a Simply Supported Beam

| Solution Form | 1 | 2 | 3 |
|---|---|---|---|
| Nine-node element | 3.7785 | 59.2236 | 290.0804 |
| Two-node element | 3.7787 | 59.2338 | 290.1774 |
| Beam theory | 3.8050 | 60.8807 | 308.2080 |

In Table 12.1 we present the values for the first three frequencies obtained from the finite element analysis and compare them to the value obtained from an exact solution for the Euler-Bernoulli beam without shear deformation. It is evident that transverse shearing strains significantly affect the higher frequencies computed for this problem and, thus, illustrate the importance of using a correct theory for calculations.

#### 12.4.5.2 *Vibration of an earth dam*

Figure 12.3 shows the vibration of a two-dimensional earth dam resting on a rigid foundation. The earth dam is modeled by linear triangular elements and includes the effects of different material layers.

#### 12.4.5.3 *Electromagnetic fields*

The basic dynamic equation (12.7) also can be derived for a variety of nonstructural problems. The eigenvalue problem once again occurs with "stiffness" and "mass" matrices now having alternate physical meanings.

A particular form of the more general equations discussed earlier is the Helmholtz wave equation which, in two-dimensional form, is

$$\frac{\partial^2 \phi}{\partial x^2} \quad \frac{\partial^2 \phi}{\partial y^2} \quad \frac{1}{c^2} \frac{\partial^2 \phi}{\partial t^2} \quad 0 \tag{12.29}$$

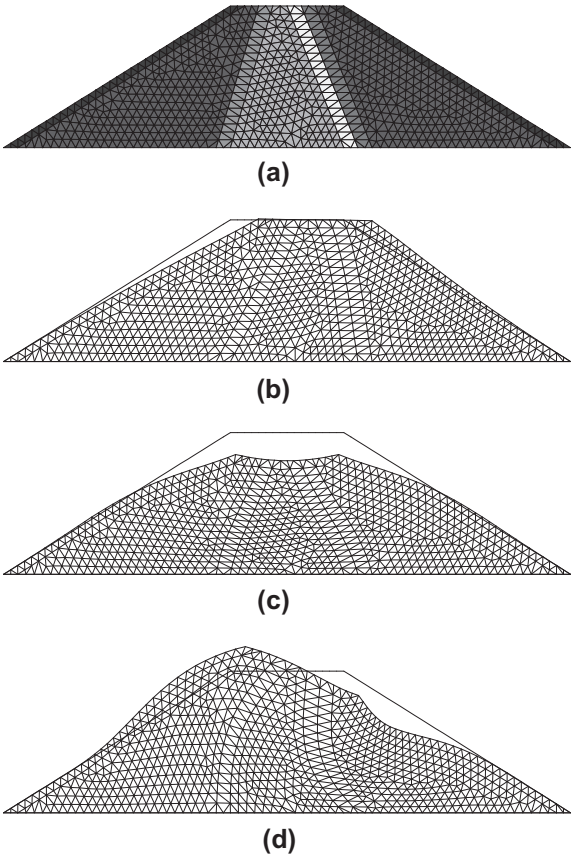



**FIGURE 12.3**
(a) Mesh showing layers considered; (b) earth dam, Mode 1; (c) earth dam, Mode 2; (d) earth dam, Mode 3.

If the boundary conditions do not force a response, an eigenvalue problem results which has significance in several fields of physical science.

The first application is to *electromagnetic fields*. Figure 12.4 shows a modal shape of a field for a *waveguide problem*. Simple linear triangular elements are used here. More complex three-dimensional oscillations are also discussed in Ref. [9].

#### 12.4.5.4 Waves in shallow water

A similar equation also describes to a reasonable approximation the behavior of shallow water waves in a body of water:

$$\frac{\partial}{\partial x}\left(h\frac{\partial \psi}{\partial x}\right)+\frac{\partial}{\partial y}\left(h\frac{\partial \psi}{\partial y}\right)+\frac{1}{g}\frac{\partial^2 \psi}{\partial t^2}=0 \tag{12.30}$$

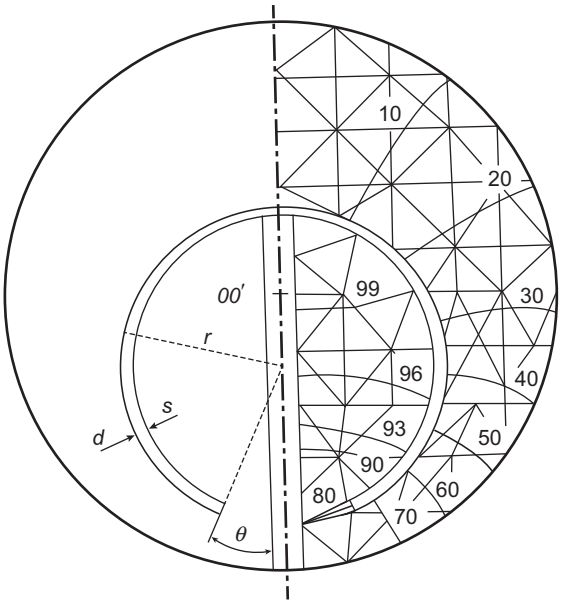



**FIGURE 12.4**

A "lunar" waveguide [9] mode of vibration for electromagnetic field. Outer diameter $d$, $OO$ $0.13d$, $r$ $0.29d$, $S$ $0.055d$, $\theta$ 22 .

in which $h$ is the average water depth, $\psi$ the surface elevation above average, and $g$ the gravity acceleration [18].

Thus natural frequencies of bodies of water contained in harbors of varying depths may easily be found [10]. Figure 12.5 shows the modal shape for a particular harbor.

## 12.5 Free response: Eigenvalues for first-order problems and heat conduction, etc.

If in Eq. (12.17) $\mathbf{M}$ $\mathbf{0}$, we have a form typical of the transient heat conduction equation [see Eq. (12.1)]. For free response we seek a solution of the homogeneous equation

$$\mathbf{Cu} \quad \mathbf{Ku} \quad \mathbf{0} \tag{12.31}$$

Once again an exponential form can be used:

$$\mathbf{u} \quad \mathbf{u}\exp(\ \lambda t)$$

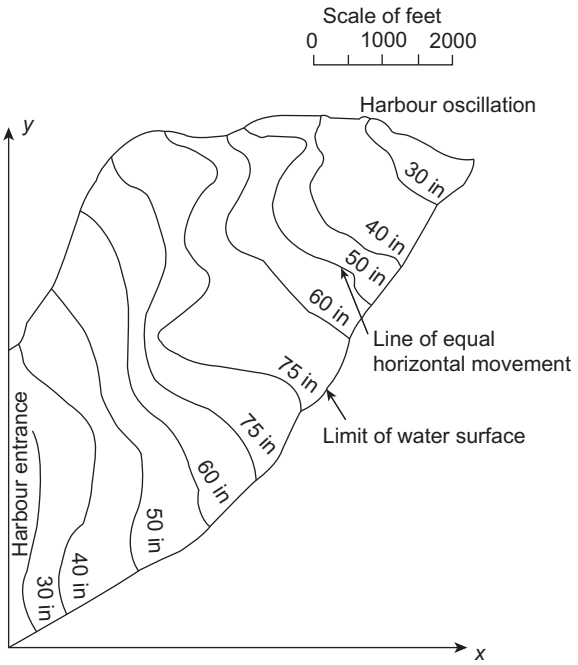



**FIGURE 12.5**
Oscillations of a natural harbor: contours of velocity amplitudes.

Substituting we have

$$\mathbf{K}\mathbf{U} \quad \mathbf{C}\mathbf{U} \quad \text{where now} \quad \begin{matrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{matrix} \tag{12.32}$$

which again gives an eigenvalue problem identical to that of Eq. (12.19). As **C** and **K** are usually positive definite, $\lambda$ will be positive and real. The solution therefore represents simply an exponential decay term and is not really steady state. Combination of such terms, however, can be useful in the solution of initial value transient problems.

## 12.6 Free response: Damped dynamic eigenvalues

We shall now consider the full Eq. (12.17) for free response conditions. Writing

$$\mathbf{M}\mathbf{u} \quad \mathbf{C}\mathbf{u} \quad \mathbf{K}\mathbf{u} \quad \mathbf{0} \tag{12.33}$$

and substituting

$$\mathbf{u} \quad \mathbf{u}\exp(\alpha t) \tag{12.34}$$

we have the characteristic equation

$$(\alpha^2\mathbf{M} \quad \alpha\mathbf{C} \quad \mathbf{K})\mathbf{u} \quad \mathbf{0} \tag{12.35}$$

where $\alpha$ and $\mathbf{u}$ will in general be found to be complex. The real part of the solution represents a decaying vibration.

The eigenvalue problem involved in Eq. (12.34) is more difficult than that arising in the previous sections. In solutions to date the problem is usually solved by splitting Eq. (12.33) into two first-order equations. This is accomplished by defining

$$\mathbf{u} \quad \mathbf{v}$$

and writing the split form as

$$\begin{matrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{M} \end{matrix} \quad \mathbf{v} \quad \begin{matrix} \mathbf{C} & \mathbf{K} \\ \mathbf{M} & \mathbf{0} \end{matrix} \quad \mathbf{v} \quad \begin{matrix} \mathbf{0} \\ \mathbf{0} \end{matrix} \tag{12.36}$$

Now substituting

$$\mathbf{u} \quad \mathbf{u}\exp(\alpha t) \qquad \mathbf{v} \quad \mathbf{v}\exp(\alpha t)$$

gives the general linear eigenproblem

$$\alpha \begin{matrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{M} \end{matrix} \quad \begin{matrix} \mathbf{C} & \mathbf{K} \\ \mathbf{M} & \mathbf{0} \end{matrix} \quad \mathbf{v} \quad \begin{matrix} \mathbf{0} \\ \mathbf{0} \end{matrix} \tag{12.37}$$

This form has been studied by Chen et al. [32–34]. Similar to the first-order problem, no steady-state solution exists and once more the concept of eigenvalues of the above kind is generally of importance only in modal analysis, as we shall see later.

## 12.7 Forced periodic response

If the forcing term in Eq. (12.17) is periodic or, more generally, if we can express it as

$$\mathbf{f} \quad \mathbf{f}\exp(\alpha t) \tag{12.38}$$

where $\alpha$ is complex, i.e.,

$$\alpha \quad \alpha_1 \quad \mathrm{i}\alpha_2 \tag{12.39}$$

then a general solution can once more be written as

$$\mathbf{u} \quad \mathbf{u}\exp(\alpha t) \tag{12.40}$$

Substituting the above in Eq. (12.17) gives

$$(\alpha^2\mathbf{M} \quad \alpha\mathbf{C} \quad \mathbf{K})\mathbf{u} \quad \mathbf{K}\mathbf{u} \quad \mathbf{f} \tag{12.41}$$

which is no longer an eigenvalue problem but can be solved formally as

$$\mathbf{u} \quad \mathbf{K}^{1}\mathbf{f} \tag{12.42}$$

The solution is thus precisely of the same form as that used for static problems but now, however, has to be determined in terms of complex quantities.

With periodic input the solution after an initial transient is not sensitive to the initial conditions and the above solution represents the finally established response. It is valid for problems of dynamic structural and fluid-structure responses as well as for problems typical of heat conduction in which we simply put $\mathbf{M} \quad \mathbf{0}$.

## 12.8 Transient response by analytical procedures

### 12.8.1 General

In the previous sections we have been concerned with steady-state general solutions which took no account of the initial conditions of the system or of the nonperiodic form of the forcing terms. The response taking these features into account is essential if we consider, for instance, the earthquake behavior of structures or the transient behavior of the heat conduction problem. The solution of such general cases requires either a full-time discretization, which we presented in earlier chapters, or the use of special analytical procedures.

### 12.8.2 Frequency response procedures

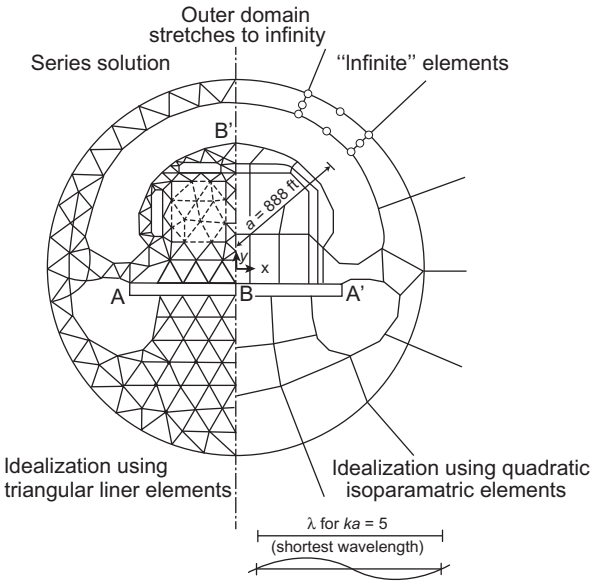
In Section 12.7 we showed how the response of the system to any forcing terms of the general periodic type or in particular to a periodic forcing function

$$\mathbf{f} \quad \mathbf{f}\exp(\mathrm{i}\,\omega t) \tag{12.43}$$

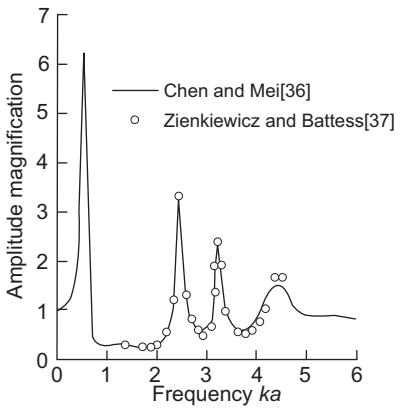
can be obtained by solving a simple equation system. As a completely arbitrary forcing function can be represented approximately by a Fourier series or in the limit, exactly, as a Fourier integral, the response to such an input can be obtained by a synthesis of a curve representing the response of any quantity of interest, e.g., the displacement at a particular point, etc., to all frequencies ranging from zero to infinity. In fact only a limited number of such forcing frequencies has to be considered and a result can be synthesized efficiently by fast Fourier transform techniques [35]. We shall not discuss the mathematical details for such procedures which can be found in standard texts on structural dynamics [12,13].

The technique of frequency response is readily adapted to problems where the damping matrix $\mathbf{C}$ is of an arbitrary specified form. This is not the case with the more widely used modal decomposition procedures which are to be described in the next section.

By way of illustration we show in Fig. 12.6 the frequency response of an artificial harbor [see Eq. (12.30)] to an input of waves with different frequencies and damping due to the radiation of reflected waves which imposes a very particular form on the damping matrix. Details of this problem are given elsewhere [18,36,37]. Similar techniques are frequently used in the analysis for the foundation response of structures where radiation of energy occurs [38].




**(a)** Geometric detail and FEM Idealization.
Wave forcing frequency $\omega = k\sqrt{gh} = ka, h$ = depth of water




**(b)** Amplitude magnification response of mean depth in harbour for various frequencies

**FIGURE 12.6**

Frequency response of an artificial harbor to an input of a periodic wave.

### 12.8.3 Modal decomposition analysis

This procedure is probably the most important and widely used in practice. Further, it provides an insight into the behavior of the whole system, which is of value where strictly numerical processes are used. We shall therefore describe it in detail in the

context of the general problem of Eq. (12.17), i.e.,

$$\mathbf{Mu} \quad \mathbf{Cu} \quad \mathbf{Ku} \quad \mathbf{f} \quad \mathbf{0} \tag{12.44}$$

where $\mathbf{f}$ is an arbitrary function of time.

We have seen that the general solution for the free response is of the form

$$\mathbf{u} \qquad \sum_{i \quad 1}^{n} \mathbf{u}_i \exp(\alpha_i t) \tag{12.45}$$

where $\alpha_i$ are the (complex) eigenvalues and $\mathbf{u}_i$ are the (complex) eigenvectors (Section 12.6). For forced response we shall assume that the problem is linear such that the solution can be written as a linear combination of the modes

$$\mathbf{u} \qquad \sum_{i \quad 1}^{n} \mathbf{u}_i y_i(t) \qquad \mathbf{u}_1, \quad \mathbf{u}_2, \quad \ldots \quad \mathbf{y}(t) \tag{12.46}$$

where the scalar modal participation factor $y_i$ is now a function of time. This shows in a clear manner the proportions of each mode occurring. Such a decomposition of an arbitrary vector presents no restriction as all the modes are linearly independent vectors (with those for repeated frequencies being constructed to be linearly independent as mentioned in Section 12.4).

If expression (12.46) is substituted into Eq. (12.44) and the result is premultiplied by the complex conjugate transposed, $\mathbf{u}_i^{\mathrm{T}}(i \quad 1, \ldots, n)$, then the result is simply a set of scalar, independent, equations

$$m_i y_i \quad c_i y_i \quad k_i y_i \quad f_i \quad 0 \tag{12.47}$$

where

$$m_i \quad \mathbf{u}_i^{\mathrm{T}}\mathbf{M}\mathbf{u}_i, \quad c_i \quad \mathbf{u}_i^{\mathrm{T}}\mathbf{C}\mathbf{u}_i, \quad k_i \quad \mathbf{u}_i^{\mathrm{T}}\mathbf{K}\mathbf{u}_i, \quad \text{and} \quad f_i \quad \mathbf{u}_i^{\mathrm{T}}\mathbf{f}$$

as for true eigenvectors $\mathbf{u}_i$

$$\mathbf{u}_i^{\mathrm{T}}\mathbf{M}\mathbf{u}_j \quad \mathbf{u}_i^{\mathrm{T}}\mathbf{C}\mathbf{u}_j \quad \mathbf{u}_i^{\mathrm{T}}\mathbf{K}\mathbf{u}_j \quad 0$$

when $i \quad j$ (this result was proved in Section 12.4 for real eigenpairs but is valid generally for complex pairs, as could be verified by the reader).

Each scalar equation of (12.47) can be solved by elementary procedures independently and the total vector of response obtained by superposition following Eq. (12.46). In the general case, as we have shown in Section 12.6, the eigenpairs are complex and their determination is not simple [29]. The more usual procedure is to use real eigenpairs corresponding to the solution of Eq. (12.18):

$$\mathbf{Ku} \quad \omega^2\mathbf{Mu}. \tag{12.48}$$

Decoupled equations with real variables $\mathbf{y}$ exist only if

$$\mathbf{u}_i^{\mathrm{T}}\mathbf{C}\mathbf{u}_j \quad 0, \qquad i \quad j$$

which generally does not occur as the eigenvectors now guarantee only orthogonality with **M** and **K** and not of the damping matrix. However, if the damping matrix **C** is of the form of Eq. (12.10), i.e., a linear combination of **M** and **K**, such orthogonality will obviously occur. Unless the damping is of a definite form which requires special treatment, an assumption of orthogonality is made and Eq. (12.47) is assumed valid in terms of such eigenvectors.

From Eq. (12.48) we have

$$\mathbf{K}\mathbf{u}_i \quad \omega_i^2 \mathbf{M}\mathbf{u}_i \tag{12.49a}$$

and on premultiplying by $\mathbf{u}_i^{\mathrm{T}}$ we obtain

$$k_i \quad \omega_i^2 m_i \tag{12.49b}$$

Writing the modal damping in the form

$$c_i \quad 2\omega_i \xi_i \tag{12.49c}$$

(where $\xi_i$ represents the ratio of damping to its critical value) and assuming that the modes have been normalized so that $m_i \quad 1$ (12.47) can be rewritten in standard second-order form:

$$y_i \quad 2\omega_i \xi_i y_i \quad \omega_i^2 y_i \quad f_i \quad 0 \tag{12.49d}$$

A general solution is then obtained as

$$\begin{aligned} y_i \quad & \exp(\ \xi_i \omega_i t) \ \frac{y_{i0} \quad \xi_i \omega_i y_{i0}}{\omega} \sin \omega_i t \quad y_{i0} \cos \omega_i t \\ & \frac{1}{\omega_i} \int_0^t \exp(\ \xi_i \omega_i \ t \quad \tau\ ) \sin \omega_i (t \quad \tau) f_i(\tau)\, \mathrm{d}\tau \end{aligned} \tag{12.50}$$

in which $\omega_i \quad \omega_i \sqrt{1 \quad \xi_i^2}$ and $y_{i0}$, $y_{i0}$ are initial conditions computed from

$$y_{i0} \quad \mathbf{u}_i^{\mathrm{T}} \mathbf{M}\mathbf{u}(0) \quad \text{and} \quad y_{i0} \quad \mathbf{u}_i^{\mathrm{T}} \mathbf{M}\mathbf{u}(0) \tag{12.51}$$

The solution of Eq. (12.50) can be carried out by assuming the forcing function is given by linear interpolation between discrete time points $t_k$ and then evaluating the resulting integrals exactly. Alternatively, a numerical solution can be carried out and the response obtained. In practice, often a single calculation is carried out for each mode to determine the maximum responses and a suitable addition of these results is used. Such processes are described in standard texts and are used as procedures to calculate the bounds on behavior of structures subjected to seismic loading [12,13,25].

### 12.8.4 Damping and participation of modes

The type of calculation implied in modal decomposition apparently necessitates the determination of all modes and eigenvalues, a task of considerable magnitude. In fact

only a limited number of modes usually need to be taken into consideration as often the response to higher frequency is critically damped and insignificant.

To show that this is true consider the form of the damping matrices. In Section 12.2 [Eq. (12.10)] we indicated that the damping matrix is often assumed as

$$\mathbf{C} \quad \alpha\mathbf{M} \quad \beta\mathbf{K} \tag{12.52}$$

Indeed a form of this type is necessary for the use of modal decomposition, although other generalizations are possible [39,40]. From the definition of $\xi_i$, the ratio to its critical ratio in Eq. (12.49c), we see that this can now be written as

$$\xi_i \quad \frac{1}{2\omega_i}\mathbf{u}_i^{\mathrm{T}}(\alpha\mathbf{M} \quad \beta\mathbf{K})\mathbf{u}_i \quad \frac{1}{2\omega_i}(\alpha \quad \beta\omega_i^2) \tag{12.53}$$

Thus if the coefficient $\beta$ is of greater importance, as is the case with most structural damping, $\xi_i$ grows with $\omega_i$ and at high frequency an overdamped condition will arise [12]. This is indeed fortunate as, in general, an infinite number of high frequencies exist which are not modeled by any finite element discretization.

## 12.9 Symmetry and repeatability

In concluding this chapter it is worth remarking that in dynamic calculation we have once again encountered all the general principles of assembly, etc., that are applicable to static problems. However, some aspects of symmetry and repeatability which were used previously (see Section 2.2.5) need amending. It is obviously possible for symmetric structures to vibrate in an unsymmetrical manner, for instance, and similarly a repeatable structure contains modes which are themselves nonrepeatable. However, even here considerable simplification can still be made; details of this are discussed by Williams [41], Thomas [42], and Evensen [43].

## 12.10 Problems

**12.1** Specialize the problem given in Section 12.2 for the case where $\rho \quad 0$. Construct $\mathbf{C}^e$ and $\mathbf{K}^e$ for a typical three-node triangular element and a six-node hierarchical triangle in which coordinates are given by

$$\mathbf{x} \quad \sum_{a \quad 1}^{3} L_a \mathbf{x}_a$$

**12.2** An axial bar under transient loading is governed by (see Section 3.2)

$$\frac{\partial}{\partial x} \quad EA\frac{\partial u}{\partial x} \quad q \quad \rho A\frac{\partial^2 u}{\partial t^2}$$

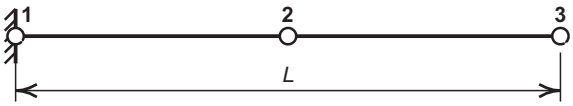



**FIGURE 12.7**
Two-element bar for Problem 12.2.

with boundary conditions

$$u(x,t) \quad g(t), \quad x \text{ on } \Gamma_1 \quad \text{or} \quad EA\frac{\partial u}{\partial x} \quad P(t), \quad x \text{ on } \Gamma_2$$

where $g(t)$ and $P(t)$ are specified displacement and force, respectively.

**(a)** Consider an isoparametric element interpolation

$$\begin{array}{lll} x^e(\xi) & N_1(\xi)\, x_1^e & N_2(\xi)\, x_2^e \\ u^e(\xi,t) & N_1(\xi)\, u_1^e(t) & N_2(\xi)\, u_2^e(t) \end{array}$$

where

$$u^e(t) \qquad u_j^e(x) \exp(i\omega_j t)$$
$$j$$

and determine the discrete eigenproblem resulting from the weak form for the problem. Is there a variational theorem for the problem?

**(b)** Consider a two-element problem shown in Fig. 12.7 and solve the eigenproblem developed in (b). Let $u(0,t)$ $P(L,t)$ 0 and use material properties $E$ $A$ $\rho$ $L$ 1 and $q$ 0.

**(c)** Write a MATLAB or GNU Octave program to solve the discrete eigenproblem. Check your program using the two-element solution; then solve the problem using 4, 8, and 16 elements. Plot the first two eigenvalues vs. the number of elements.

**(d)** Obtain an exact value for the first two eigenvalues from the differential equations and plot the error for each vs. the number of elements on a log-log plot. What is the rate of convergence?

**(e)** Replace the element mass matrix by a *lumped* form given by

$$\mathbf{M}^e \quad \frac{1}{2}\rho h \quad \begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix}$$

and repeat parts (c) and (d)

**12.3** Compute the lumped mass matrix by row sum (Method I) for a cubic order serendipity element which is a square with side length $a$. Note it is only necessary to compute one vertex and one mid-side value.

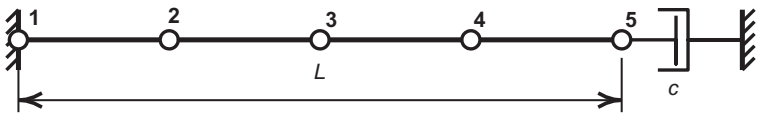



**FIGURE 12.8**
Four-element bar with end damper for Problem 12.8.

**12.4** Compute the lumped mass matrix by diagonal scaling (Method II) for a cubic order serendipity element which is a square with side length $a$. Note it is only necessary to compute one vertex and one mid-side value.

**12.5** Compute the lumped mass matrix by row sum (Method I) for a cubic order Lagrangian element which is a square with side length $a$. Note it is only necessary to compute one vertex, one mid-side value, and the interior node value.

**12.6** For a three-dimensional cube with side lengths $a$ in each direction compute the lumped mass matrix by row sum (Method I) for a quadratic order Lagrangian element. Note it is only necessary to compute one vertex, one mid-side value, and the interior node value.

**12.7** The bar shown in Fig. 12.8 is divided into four elements and has the right end attached to a damper. A weak form for the problem may be written as

$$\mathbf{M}\,\mathbf{u} \quad \mathbf{C}\,\mathbf{u} \quad \mathbf{K}\,\mathbf{u} \quad \mathbf{f} \quad \mathbf{0}$$

where $\mathbf{u} \quad u_2 \quad u_3 \quad u_4 \quad u_5 \;^T$:

**(a)** Construct **M**, **C**, and **K** for the problem.

**(b)** Use MATLAB or GNU Octave to compute the eigensolution for the problem. Let $E \quad A \quad \rho \quad L \quad 1$ and $c \quad 0.1$. Plot real and imaginary parts for the problem. On a separate plot show as vectors the real and imaginary parts of the complex frequencies. Are they proportional?

**12.8** Use *FEAPpv* to verify the results given in Table 12.1 and Fig. 12.2. Use a consistent mass matrix for the computation. Repeat the analysis using a lumped mass. If the mesh is refined several times, do you expect the results to converge? Why?

**12.9** Use *FEAPpv* to compute the first three eigenpairs for the rectangular beam problem described in Example 7.5. Use the same properties for $E$ and $\nu$ and let $\rho \quad 0.001$.

**(a)** For the case of linear variation of displacements on the boundary edge use *FEAPpv* to compute the reactions for a specified displacement $u_0$ and four equally spaced radial segments. Scale the reactions to obtain a unit force $P$.

**(b)** Using the first mode and $P \quad 1$ for $0 < t < 2$ obtain the solution for the first 3 s using $\Delta t \quad 0.1$. Plot the results for the vertical displacement at the end $x \quad y \quad 0$.

**(c)** Repeat the solution using three modes.

**12.10** The circular beam problem described in Example 7.6 is to be solved for the case where the boundary condition at $y$ 0 is specified by the shear stress of the exact solution.

- **(a)** For the case of linear variation of displacements on the boundary edge use *FEAPpv* to compute the reactions for a specified displacement $u_0$ and four equally spaced radial segments. Scale the reactions to obtain a unit force $P$.
- **(b)** For the mesh shown in Fig. 7.13(a) use *FEAPpv* to compute the first eigenpair (the one where $\omega_j$ is smallest) for the problem assuming the same properties for $E$ and $\nu$ and take $\rho$ 1.
- **(c)** Using the first mode and $u(x, 0)$ $\sin^2 t$ for $0 < t < \pi$ obtain the solution for the first 5 seconds using $\Delta t$ 0.1. Plot the results for the vertical displacement at the tip where loading is applied.
- **(d)** Repeat the solution using three modes.

**12.11** Program development project: Extend the program system started in Problem 7.20 to compute a lumped and a consistent mass matrix for three-node triangular and four-node quadrilateral plane stress elements. Use the generalized eigen-problem `[V,D] = EIG (K,M)` from MATLAB or GNU Octave to compute the eigenvectors (`V` $u_i$) and eigenvalues (`D` $\omega_i$).
Use your program to determine the eigenvalues and eigenvectors for the curved beam analyzed in Problem 12.10. Results may be checked using *FEAPpv*.

**12.12** Program development project: Extend the program system developed for Problem 12.11 to perform mode superposition as described in Section 12.8.3. You may omit the modal damping factors $\xi_i$ for simplicity.
For the rectangular beam considered in Problem 12.9 (using four-node quadrilateral elements), assume the end shear is applied suddenly at time zero and held constant for 2 seconds at which time it is suddenly removed.
Perform a modal solution in which only the lowest eigenvalue mode is used. For a time increment of $\Delta t$ 0.001, determine and plot the first 5 seconds of response for the vertical displacement at the tip centerline. Repeat the solution using the lowest three eigenvalue modes. Compare your solutions with that in which all modes are included (which is the exact solution for the *semi-discrete* equations). Comment on the differences obtained.

## References

[1] S. Crandall, Engineering Analysis, McGraw-Hill, New York, 1956.
[2] H.S. Carslaw, J.C. Jaeger, Conduction of Heat in Solids, second ed., Oxford University Press, Oxford, 1959.
[3] W. Visser, A finite element method for the determination of non-stationary temperature distribution and thermal deformation, in: Proceedings of 1st

Conference on Matrix Methods in Structural Mechanics, volume AFFDL-TR-66-80, Wright Patterson Air force Base, Ohio, 1966.
[4] O.C. Zienkiewicz, Y.K. Cheung, The Finite Element Method in Structural Mechanics, McGraw-Hill, London, 1967.
[5] E.L. Wilson, R.E. Nickell, Application of finite element method to heat conduction analysis, Nucl. Eng. Des. 4 (1966) 1–11.
[6] O.C. Zienkiewicz, C.J. Parekh, Transient field problems—two- and three-dimensional analysis by isoparametric finite elements, Int. J. Numer. Methods Eng. 2 (1970) 61–71.
[7] K. Terzhagi, R.B. Peck, Soil Mechanics in Engineering, John Wiley & Sons, New York, 1948.
[8] D.K. Todd, Ground Water Hydrology, John Wiley & Sons, New York, 1959.
[9] P.L. Arlett, A.K. Bahrani, O.C. Zienkiewicz, Application of finite elements to the solution of Helmholz's equation, Proc. IEE 115 (1968) 1762–1764.
[10] C. Taylor, B.S. Patil, O.C. Zienkiewicz, Harbour oscillation: a numerical treatment for undamped natural modes, Proc. Inst. Civ. Eng. 43 (1969) 141–156.
[11] O.C. Zienkiewicz, R.E. Newton, Coupled vibration of a structure submerged in a compressible fluid, in: Proceedings of International Symposium on Finite Element Techniques, Stuttgart, 1969, pp. 359–371.
[12] A.K. Chopra, Dynamics of Structures, Prentice-Hall, Upper Saddle River, N.J., 1995.
[13] R.W. Clough, J. Penzien, Dynamics of Structures, second ed., McGraw-Hill, New York, 1993.
[14] S.W. Key, Z.E. Beisinger, The transient dynamic analysis of thin shells in the finite element method, in: Proceedings of 1st Conference on Matrix Methods in Structural Mechanics, volume AFFDL-TR-66-80, Wright Patterson Air Force Base, Ohio, October 1966, pp. 667–710.
[15] E. Hinton, T. Rock, O.C. Zienkiewicz, A note on mass lumping and related processes in the finite element method, Earthquake Eng. Struct. Dyn. 4 (1976) 245–249.
[16] P. Tong, T.H.H. Pian, L.L. Bociovelli, Mode shapes and frequencies by the finite element method using consistent and lumped matrices, Comput. Struct. 1 (1971) 623–638.
[17] I. Fried, D.S. Malkus, Finite element mass matrix lumping by numerical integration with the convergence rate loss, Int. J. Solids Struct. 11 (1975) 461–465.
[18] O.C. Zienkiewicz, R.L. Taylor, P. Nithiarasu, The Finite Element Method for Fluid Dynamics, seventh ed., Elsevier, Oxford, 2013.
[19] J.H. Wilkinson, The Algebraic Eigenvalue Problem, Clarendon Press, Oxford, 1965.
[20] I. Fried, Gradient Methods for Finite Element Eigen Problems, J. AIAA 7 (1969) 739–741.
[21] J.H. Wilkinson, C. Reinsch, Linear Algebra, Handbook for Automatic Computation, vol. II, Springer-Verlag, Berlin, 1971.

[22] K.K. Gupta, Solution of eigenvalue problems by Sturm sequence method, Int. J. Numer. Methods Eng. 4 (1972) 379–404.
[23] A. Jennings, Mass condensation and similarity iterations for vibration problems, Int. J. Numer. Methods Eng. 6 (1973) 543–552.
[24] B.N. Parlett, The Symmetric Eigenvalue Problem, Prentice-Hall, Englewood Cliffs, N.J., 1980.
[25] K.-J. Bathe, Finite Element Procedures, Prentice Hall, Englewood Cliffs, N.J., 1996.
[26] J. Demmel, Applied Numerical Linear Algebra, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1997.
[27] H.L. Cox, Vibration of missiles, Aircraft Eng. 33 (1961) 2–7 and 48–55.
[28] A. Jennings, Natural vibration of a free structure, Aircraft Eng. 34 (1962) 8.
[29] W.C. Hurty, M.F. Rubinstein, Dynamics of Structures, Prentice Hall, Englewood Cliffs, N.J., 1974.
[30] A. Craig, M.C.C. Bampton, On the iterative solution of semi definite eigenvalue problems, Aero. J. 75 (1971) 287–290.
[31] O.C. Zienkiewicz, R.L. Taylor, The Finite Element Method, fourth ed., vol. 2, McGraw-Hill, London, 1991.
[32] H.-C. Chen, R.L. Taylor, Using Lanczos vectors and Ritz vectors for computing dynamic responses, Engrg. Comp. 6 (1989) 151–157.
[33] A. Ibrahimbegovic, H.-C. Chen, E.L. Wilson, R.L. Taylor, Ritz method for dynamic analysis of large discrete linear systems with non-proportional damping, Earthquake Eng. Struct. Dyn. 19 (1990) 877–889.
[34] H.-C. Chen and R.L. Taylor. Properties and solutions of the eigensystem of non-proportionally damped linear dynamic systems. Technical Report UCB/SEMM-86/10, University of California, Berkeley, November 1986.
[35] E.O. Brigham, The Fast Fourier Transform, Prentice Hall, Englewood Cliffs, N.J., 1974.
[36] H.S. Chen, C.C. Mei, Hybrid-element method for water waves, in: Proceedings of Modelling Techniques Conference (Modelling 1975), vol. 1, San Francisco, 1975, pp. 63–81.
[37] O.C. Zienkiewicz, P. Bettess, Infinite elements in the study of fluid-structure interaction problems, in: Second International Symposium on Computing Methods in Applied Science and Engineering, Versailles, France, December 1975.
[38] J. Penzien, Frequency domain analysis including radiation damping and water load coupling, in: O.C. Zienkiewicz, R.W. Lewis, K.G. Stagg (Eds.), Numerical Methods in Offshore Engineering, John Wiley & Sons, Chichester, 1978.
[39] E.L. Wilson, J. Penzien, Evaluation of orthogonal damping matrices, Int. J. Numer. Methods Eng. 4 (1972) 5–10.
[40] H.T. Thomson, T. Collins, P. Caravani, A numerical study of damping, Earthquake Eng. Struct. Dyn. 3 (1974) 97–103.
[41] F.W. Williams, Natural frequencies of repetitive structures. Q.J. Mech, Appl. Math. 24 (1971) 285–310.

[42] D.L. Thomas, Standing waves in rotationally periodic structures, J. Sound Vibr. 37 (1974) 288–290.
[43] D.A. Evensen, Vibration analysis of multi-symmetric structures, J. AIAA 14 (1976) 446–453.

This page is intentionally left blank

CHAPTER

# 13 Plate Bending Approximation: Thin and Thick Plates

## 13.1 Introduction

Plates and shells are but a particular form of a three-dimensional solid, the treatment of which presents no theoretical difficulties, at least in the case of elasticity. However, the thickness of such structures (denoted throughout this chapter as $h$) is very small when compared with other dimensions, and complete three-dimensional numerical treatment is not only costly but in addition often leads to serious numerical ill-conditioning problems. To ease the solution, even long before numerical approaches became possible, several classical assumptions regarding the behavior of such structures were introduced. Clearly, such assumptions result in a series of approximations. Thus numerical treatment will, in general, concern itself with the approximation to an already approximate theory (or mathematical model), the validity of which is restricted. On occasion we shall point out the shortcomings of the original assumptions, and indeed modify these as necessary or convenient. This can be done simply because now we are granted more freedom than that which existed in the "pre-computer" era.

The *thin plate* theory is based on the assumptions formalized by Kirchhoff in 1850 [1], and indeed his name is often associated with this theory, though an early version was presented by Sophie Germain in 1811 [2–4]. A relaxation of the assumptions was made by Reissner in 1945 [5] and in a slightly different manner by Mindlin [6] in 1951. These modified theories extend the field of application of the theory to *thick plates* and we shall associate this name with the Reissner-Mindlin postulates.

It turns out that the thick plate theory is simpler to implement in the finite element method, though in the early days of analytical treatment it presented more difficulties. As it is more convenient to introduce first the thick plate theory and by imposition of additional assumptions to limit it to thin plate theory we shall follow this path in the present chapter. However, when discussing numerical solutions we shall reverse the process and follow the historical procedure of dealing with the thin plate situations first. The extension to thick plates and to what turns out always to be a *mixed* formulation will be the subject of later sections.

In thick plate theory the kinematics is specified by a lateral displacement of the mid-surface $w$ along with a vector of rotations $\boldsymbol{\phi}$. The strains resulting from the displacement and rotation variables involve only first derivatives and, thus, it appears possible that one can employ the $C_0$ interpolations used in previous developments.

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00013-7

We shall find, however, that once again locking can occur unless we use carefully paired functions.

In the thin plate theory it is possible to represent the state of deformation by a single quantity $w$, the lateral displacement of the middle plane of the plate. Thus we find that thin plates share some of the same characteristics as the Euler-Bernoulli beam theory studied in introductory strength of materials [7–9]. Clearly, the formulation is *irreducible*. The achievement of this irreducible form introduces second derivatives of $w$ in the strain definition and continuity conditions between elements now have to be imposed not only on this quantity but also on its derivatives ($C_1$ continuity). This is to ensure that the plate remains continuous and does not "kink." Thus at nodes on element interfaces it will always be necessary to use both the value of $w$ and its slopes (first derivatives of $w$) to impose continuity.

## 13.2 Governing equations

### 13.2.1 One-dimensional theory: Cylindrical bending

The mechanics of plate action can be illustrated in one dimension by considering a plate of infinite extent in one dimension (here assumed the $y$). Here we consider the problem of cylindrical bending of plates [2]. In this problem the plate is to be loaded and supported by conditions independent of $y$. In this case we may analyze a strip of unit width subjected to some *stress resultants* $M_x$, $P_x$, and $Q_x$, which denote $x$-direction bending moment, axial force, and transverse shear force, respectively, as shown in Fig. 13.1. For cross-sections that are originally normal to the middle plane of the plate we introduce the approximation that plane sections will remain plane during the deformation process. The postulate that sections normal to the middle plane remain plane during deformation is thus the *first* and most important assumption of the theory

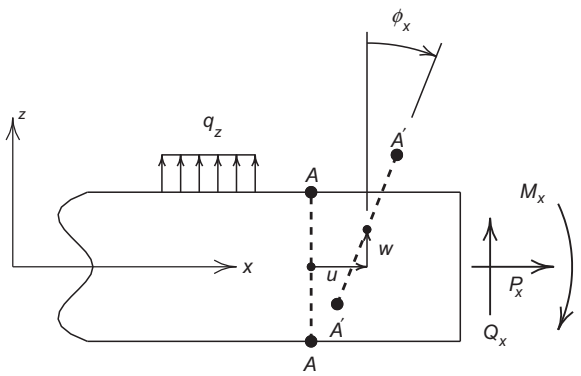



**FIGURE 13.1**
Displacements and force resultants for cylindrical bending of a plate.

of plates (and indeed shells). To this is added the *second* assumption. This simply observes that the direct stresses in the normal direction, $z$, are small, that is, of the order of applied lateral load intensities, $q_z$, and hence direct strains in that direction can be neglected. This "inconsistency" in approximation results in assuming a plane stress condition in each lamina of the plate.

#### 13.2.1.1 Plate kinematics

With the above two assumptions it is easy to see that the total state of deformation can be described by displacements $u$ and $w$ of the middle surface ($z = 0$) and a rotation $\phi_x$ of the normal (Fig. 13.1). Thus the local displacements in the directions of the $x$ and $z$ axes are taken as

$$u_x(x,\ z,\ t) = u(x,\ t) + z\phi_x(x,\ t) \ \text{ and } \ u_z(x,\ z,\ t) = w(x,\ t) \tag{13.1}$$

Immediately the strains in the $x$ and $z$ directions are available as

$$\begin{aligned} \varepsilon_x &= \frac{\partial u_x}{\partial x} = \frac{\partial u}{\partial x} + z\frac{\partial \phi_x}{\partial x} \\ \varepsilon_z &= \varepsilon_y = 0 \\ \gamma_{xz} &= \frac{\partial u_x}{\partial z} + \frac{\partial u_z}{\partial x} = \frac{\partial w}{\partial x} + \phi_x \\ \gamma_{xy} &= \gamma_{yz} = 0 \end{aligned} \tag{13.2}$$

#### 13.2.1.2 Stress resultants and constitution

For the cylindrical bending problem a state of linear elastic, plane stress for each lamina yields the stress-strain relations

$$\sigma_x = \frac{E}{1-\nu^2}\,\varepsilon_x \quad \text{and} \quad \tau_{xz} = G\gamma_{xz}$$

The stress resultants are obtained as

$$\begin{aligned} P_x &= \int_{-h/2}^{h/2} \sigma_x \,\mathrm{d}z = B\,\frac{\partial u}{\partial x} \\ Q_x &= \int_{-h/2}^{h/2} \tau_{xz} \,\mathrm{d}z = \kappa G h \left(\frac{\partial w}{\partial x} + \phi_x\right) \\ M_x &= \int_{-h/2}^{h/2} \sigma_x\, z \,\mathrm{d}z = D\,\frac{\partial \phi_x}{\partial x} \end{aligned} \tag{13.3a}$$

where $B$ is the in-plane plate stiffness and $D$ the bending stiffness and they are computed for an isotropic elastic material from

$$B = \frac{Eh}{1-\nu^2} \quad \text{and} \quad D = \frac{Eh^3}{12(1-\nu^2)} \tag{13.3b}$$

with $\nu$ Poisson's ratio, and $E$ and $G$ Young's and shear elastic moduli, respectively. A parameter $\kappa$ has been added to account for the fact that the shear stresses are not constant across the section. A value of $\kappa = 5/6$ is used for a rectangular, homogeneous section, and corresponds to a parabolic shear stress distribution.

#### 13.2.1.3 Equilibrium equations

Three equations of equilibrium may be computed directly from a differential element of the plate or by integration over the thickness of the local equilibrium equations. Using the latter approach we have for the axial resultant

$$\begin{aligned} \int_{-h/2}^{h/2} \left[ \frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xz}}{\partial z} + b_x \right] \mathrm{d}z &= \int_{-h/2}^{h/2} \rho \ddot{u}_x \, \mathrm{d}z \\ \frac{\partial}{\partial x} \int_{-h/2}^{h/2} \sigma_x \, \mathrm{d}z + \int_{-h/2}^{h/2} b_x \, \mathrm{d}z + \tau_{xz}\Big|_{h/2} - \tau_{xz}\Big|_{-h/2} &= \rho h \ddot{u} \\ \frac{\partial P_x}{\partial x} + q_x &= \rho h \ddot{u} \end{aligned} \tag{13.4a}$$

where $q_x$ is an axial load. Similarly, the shear resultant follows from

$$\begin{aligned} \int_{-h/2}^{h/2} \left[ \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \sigma_z}{\partial z} + b_z \right] \mathrm{d}z &= \int_{-h/2}^{h/2} \rho \ddot{u}_z \, \mathrm{d}z \\ \frac{\partial}{\partial x} \int_{-h/2}^{h/2} \tau_{xz} \, \mathrm{d}z + \int_{-h/2}^{h/2} b_x \, \mathrm{d}z + \sigma_z\Big|_{h/2} - \sigma_z\Big|_{-h/2} &= \rho h \ddot{w} \\ \frac{\partial Q_x}{\partial x} + q_z &= \rho h \ddot{w} \end{aligned} \tag{13.4b}$$

where the transverse loading $q_z$ arises from the body force and the resultant of the normal traction on the top and/or bottom surfaces. Finally, the moment equilibrium is deduced from

$$\begin{aligned} \int_{-h/2}^{h/2} z \left[ \frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xz}}{\partial z} + b_x \right] \mathrm{d}z &= \int_{-h/2}^{h/2} z \, \rho \ddot{u}_x \, \mathrm{d}z \\ \frac{\partial}{\partial x} \int_{-h/2}^{h/2} z \, \sigma_x \, \mathrm{d}z - \int_{-h/2}^{h/2} \tau_{xz} \, \mathrm{d}z & \\ + z \, \tau_{xz}\Big|_{h/2} - z \, \tau_{xz}\Big|_{-h/2} + \int_{-h/2}^{h/2} z \, b_x \, \mathrm{d}z &= \frac{1}{12} \rho h^3 \ddot{\phi}_x \\ \frac{\partial M_x}{\partial x} - Q_x + m_x &= \frac{1}{12} \rho h^3 \ddot{\phi}_x \end{aligned} \tag{13.4c}$$

The inertia term is termed a *rotary inertia* and is often ignored in thin plate theory. Generally, $m_x$ loads also are not included.

In the elastic case of a plate it is easy to see that the in-plane displacements and forces, $u$ and $P_x$, decouple from the other terms and the problem of lateral deformations can be dealt with separately. Indeed, the problem is similar to the one-dimensional elasticity problem treated in Chapter 3. We shall thus only consider bending in the present chapter, returning to the combined problem, characteristic of shell behavior, in the next chapter.

Equations (13.1)–(13.4c) are typical for thick plates, and the thin plate theory adds an additional assumption. This simply neglects the shear deformation and puts $G = \infty$ (or $\gamma_{xz} = 0$). Equation (13.3a) thus becomes

$$\frac{\partial w}{\partial x} + \phi_x = 0 \quad \text{or} \quad \phi_x = -\frac{\partial w}{\partial x} \tag{13.5}$$

This thin plate assumption is equivalent to stating that the normals to the middle plane remain normal to it during deformation and is the same as the Euler-Bernoulli assumption for thin beams. Indeed, treatment of beams merely replaces $D$ by $EI$, where $I$ is the section area moment of inertia. The thin, constrained theory is very widely used in practice and proves adequate for a large number of structural problems, though, of course, it should not be taken literally as the true behavior near supports or where local load action is important and is three-dimensional. In Table 13.1 we summarize the equations for treating cylindrical bending of thick and thin plates.

**Table 13.1** Equations for Thick and Thin Plates in Cylindrical Bending

| | Thick Plates | Thin Plates |
|---|---|---|
| Displacements: | $u_x = z\,\phi_x(x,t)$<br>$u_z = w(x,t)$ | $u_x = -z\,\dfrac{\partial w}{\partial x}$<br>$u_z = w(x,t)$ |
| Strains: | $\varepsilon_x = z\,\dfrac{\partial \phi_x}{\partial x}$<br>$\gamma_{xz} = \dfrac{\partial w}{\partial x} + \phi_x$ | $\varepsilon_x = -z\,\dfrac{\partial^2 w}{\partial x^2}$ |
| Resultants: | $M_x = \displaystyle\int_{-h/2}^{h/2} \sigma_x\, z\, \mathrm{d}z$<br>$Q_x = \displaystyle\int_{-h/2}^{h/2} \tau_{xz}\, \mathrm{d}z$ | $M_x = \displaystyle\int_{-h/2}^{h/2} \sigma_x\, z\, \mathrm{d}z$ |
| Equilibrium: | $\dfrac{\partial Q_x}{\partial x} + q_z = \rho h \ddot{w}$<br>$\dfrac{\partial M_x}{\partial x} - Q_x + m_x = \dfrac{1}{12}\rho h^3 \ddot{\phi}_x$ | $Q_x = \dfrac{\partial M}{\partial x}$<br>$\dfrac{\partial^2 M_x}{\partial x^2} + q_z = \rho h \ddot{w}$ |
| Constitution: | $M_x = D\,\dfrac{\partial \phi_x}{\partial x}$<br>$Q_x = \kappa G h \left( \dfrac{\partial w}{\partial x} + \phi_x \right)$ | $M_x = -D\,\dfrac{\partial^2 w}{\partial x}$ |

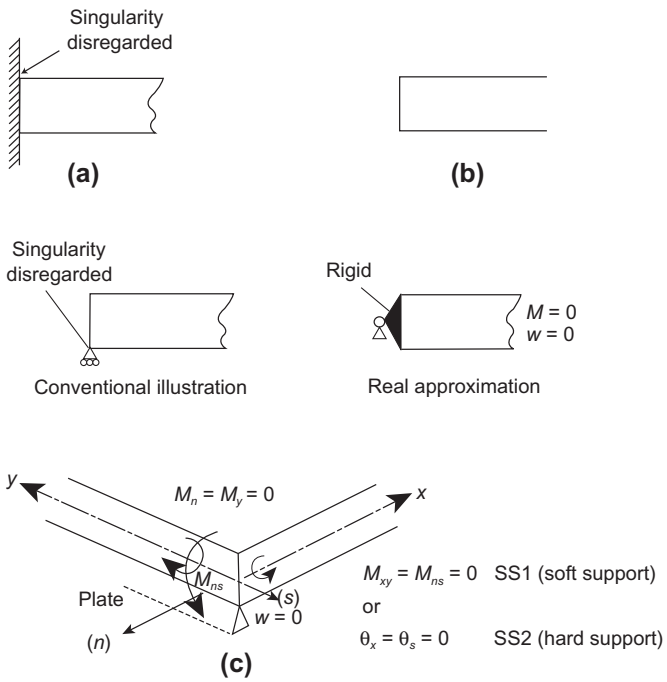



**FIGURE 13.2**

Support (end) conditions for a plate. Note: The conventionally illustrated simple support leads to infinite displacement—reality is different: (a) built-in support (clamped) with $u = v = w = 0$, $\theta = 0$; (b) free edge with $M = 0$, $S = 0 (P = 0)$; and (c) simply supported condition.

#### 13.2.1.4 *Boundary conditions*

In Fig. 13.2 we illustrate some of the boundary conditions imposed on plates and immediately note that the diagrammatic representations of simple support as a knife edge would lead to infinite displacements and stresses. Of course, if a rigid bracket is added in the manner shown this will alter the behavior to that which we shall generally assume.

### 13.2.2 Weak form for cylindrical bending

The weak form for plates may be deduced by integrating through the thickness the weak form for the elasticity equations. Accordingly, for the cylindrical bending problem we have

$$\mathrm{G} = \int_L \int_{-h/2}^{h/2} \left[ \delta u_x \, \rho \, \ddot{u}_x + \delta u_z \, \rho \, \ddot{u}_z + \delta \varepsilon_x \, \sigma_x + \delta \gamma_{xz} \, \tau_{xz} \right] \mathrm{d}z \, \mathrm{d}x$$
$$- \int_L \int_{-h/2}^{h/2} \left( \delta u_x \, b_x + \delta u_z \, b_z \right) \mathrm{d}z \, \mathrm{d}x - \int_{-h/2}^{h/2} \left( \delta u_x \, \bar{t}_x + \delta u_z \, \bar{t}_z \right) \mathrm{d}z \Big|_{\Gamma_L}$$

Inserting the variation of displacements and integrating over the thickness of the plate then gives the thick plate form

$$\begin{aligned} \mathrm{G} = & \int_L \left[ \delta\phi_x \, \tfrac{1}{12}\rho h^3 \ddot{\phi}_x + \delta w \, \rho h \, \ddot{w} + \frac{\partial \delta\phi_x}{\partial x} \, M_x + \left( \frac{\partial \delta w}{\partial x} + \delta\phi_x \right) Q_x \right] \mathrm{d}x \\ & - \int_L \left( \delta\phi_x \, m_x + \delta w \, q_z \right) \mathrm{d}x - \left( \delta\phi_x \, \bar{M}_x + \delta w \, \bar{Q}_x \right) \Big|_{\Gamma_L} \end{aligned}$$

The form for the thin plate results by replacing $\phi_x$ by $-\partial w/\partial x$ and ignoring the effects of $m_x$ and rotatory inertia to give

$$\begin{aligned} \mathrm{G} = & \int_L \left[ \delta w \, \rho h \, \ddot{w} - \frac{\partial^2 \delta w}{\partial x^2} \, M_x \right] \mathrm{d}x \\ & - \int_L \delta w \, q_z \, \mathrm{d}x - \left( - \frac{\partial \delta w}{\partial x} \, \bar{M}_x + \delta w \, \bar{Q}_x \right) \Big|_{\Gamma_L} \end{aligned}$$

#### 13.2.2.1 *Irreducible form*

The irreducible displacement form for each problem is obtained by inserting the constitutive equations from Table 13.1. For the thick plate this gives

$$\begin{aligned} \mathrm{G} = & \int_L \left[ \delta\phi_x \, \tfrac{1}{12}\rho h^3 \ddot{\phi}_x + \delta w \, \rho h \, \ddot{w} \right] \mathrm{d}x \\ & + \int_L \left[ \frac{\partial \delta\phi_x}{\partial x} \, D \, \frac{\partial \phi_x}{\partial x} + \left( \frac{\partial \delta w}{\partial x} + \delta\phi_x \right) \kappa G h \, \left( \frac{\partial w}{\partial x} + \phi_x \right) \right] \mathrm{d}x \\ & - \int_L \left( \delta\phi_x \, m_x + \delta w \, q_z \right) \mathrm{d}x - \left( \delta\phi_x \, \bar{M}_x + \delta w \, \bar{Q}_x \right) \Big|_{\Gamma_L} \end{aligned} \tag{13.6}$$

and for the thin plate the form

$$\begin{aligned} \mathrm{G} = & \int_L \left[ \delta w \, \rho h \, \ddot{w} + \frac{\partial^2 \delta w}{\partial x^2} \, D \, \frac{\partial^2 w}{\partial x^2} \right] \mathrm{d}x \\ & - \int_L \delta w \, q_z \, \mathrm{d}x - \left( - \frac{\partial \delta w}{\partial x} \, \bar{M}_x + \delta w \, \bar{Q}_x \right) \Big|_{\Gamma_L} \end{aligned} \tag{13.7}$$

in which both $m_x$ and rotatory inertia are omitted.

### 13.2.3 Finite element approximation

We first consider the thin plate problem where we notice that now second derivatives of the dependent variable appear. For this problem we require $\mathrm{C}_1$ compatibility in which $w$ and $\partial w/\partial x$ are continuous throughout the entire length of the plate $L$. For the one-dimensional problem this is easy to obtain using *Hermite interpolation*. The lowest order Hermite interpolation is cubic and may be expressed on a two-node element as

$$w(\xi, t) = \sum_{a=1}^{2} \left[ \, H_a^w(\xi) \;\; H_a^\theta(\xi) \, \right] \left\{ \begin{matrix} \tilde{w}_a(t) \\ \tilde{\theta}_a(t) \end{matrix} \right\} = \mathbf{N}_a \, \tilde{\mathbf{u}}_a \tag{13.8}$$

where $\tilde{w}_a$ and $\tilde{\theta}_a$ are nodal parameters of displacement and rotation, respectively, at the ends of the element and the shape functions are defined on the interval $-1 \le \xi \le 1$ as

$$\begin{aligned} H_1^w(\xi) &= \frac{1}{4}\left(2 - 3\,\xi + \xi^3\right), \quad H_1^\theta(\xi) = \frac{1}{8}L_e\left(\ \ 1 - \xi - \xi^2 + \xi^3\right) \\ H_2^w(\xi) &= \frac{1}{4}\left(2 + 3\,\xi - \xi^3\right), \quad H_2^\theta(\xi) = \frac{1}{8}L_e\left(-1 - \xi + \xi^2 + \xi^3\right) \end{aligned} \tag{13.9}$$

where $L_e$ is the element length. The mapping from $\xi$ to $x$ is given by the linear interpolation

$$x = \frac{1}{2}(1-\xi)\,x_1 + \frac{1}{2}(1+\xi)\,x_2$$

which gives $L_e = x_2 - x_1$. The Jacobian $j$ from the coordinate transformation is constant and equal to $L_e/2$. Similarly, since the coordinate mapping is linear,[1] the derivative for the chain rule is constant and given by

$$\frac{\partial x}{\partial \xi} = \frac{1}{2}L_e$$

From the above discussion we know that the displacement interpolation contains all global polynomials in $x$ up to cubic degree. This ensures that all rigid body modes and constant bending curvature can be obtained in each element. In addition we can compute a constant shearing force $Q_x$ in each element to yield a complete solution.

#### 13.2.3.1 *Element arrays: Thin plates*

Inserting the above approximation into the weak form defines the element mass, stiffness, and load matrices, which are expressed as

$$\begin{aligned} \mathbf{M}_{ab}^e &= \int_{-1}^{1} \begin{Bmatrix} H_a^w \\ H_a^\theta \end{Bmatrix} \rho h \begin{bmatrix} H_b^w & H_b^\theta \end{bmatrix} j\,\mathrm{d}\xi \\ \mathbf{K}_{ab}^e &= \int_{-1}^{1} \begin{Bmatrix} \dfrac{\partial^2 H_b^w}{\partial x^2} \\ \dfrac{\partial^2 H_a^\theta}{\partial x^2} \end{Bmatrix} D \begin{bmatrix} \dfrac{\partial^2 H_b^w}{\partial x^2} & \dfrac{\partial^2 H_b^\theta}{\partial x^2} \end{bmatrix} j\,\mathrm{d}\xi \\ \mathbf{f}_a^e &= \int_{-1}^{1} \begin{Bmatrix} H_a^w \\ H_a^\theta \end{Bmatrix} q_z\, j\,\mathrm{d}\xi + \begin{Bmatrix} H_a^w \\ H_a^\theta \end{Bmatrix} \bar{Q}_x\Big|_{\Gamma_L} - \begin{Bmatrix} \dfrac{\partial H_a^w}{\partial x} \\ \dfrac{\partial H_a^\theta}{\partial x} \end{Bmatrix} \bar{M}_x\Big|_{\Gamma_L} \end{aligned} \tag{13.10}$$

where $j = L_e/2$ and derivatives are given by

$$\frac{\partial^2 H_a^w}{\partial x^2} = \frac{\partial^2 H_a^w}{\partial \xi^2}\frac{4}{L_e^2}, \quad \frac{\partial^2 H_a^\theta}{\partial x^2} = \frac{\partial^2 H_a^\theta}{\partial \xi^2}\frac{4}{L_e^2}$$

[1] We note that the form for coordinate and displacement are not an isoparametric form. This has mapping implications when general plates are considered.

After assembly of all the individual element arrays we obtain the standard semi-discrete form

$$\mathbf{M}\,\ddot{\tilde{\mathbf{u}}} + \mathbf{K}\,\tilde{\mathbf{u}} = \mathbf{f}$$

where

$$\mathbf{M} = \sum_{e=1}^{M} \mathbf{M}^e, \quad \mathbf{K} = \sum_{e=1}^{M} \mathbf{K}^e, \quad \mathbf{f} = \sum_{e=1}^{M} \mathbf{f}^e$$

with $M$ denoting the total number of elements used.

#### 13.2.3.2 Simply supported, uniformly loaded thin plate

As an example using the above formulation, we consider the static cylindrical bending of a simply supported, uniformly loaded plate as shown in Fig. 13.3. The plate has a span of $L = 100$ units and depth of $h = 2$ units. Material properties are $E = 1.365 \times 10^6$, $\nu = 0.3, \kappa = 5/6$ with a load of $q_z = 10$ per unit length. The central displacement gives $w = 1.3020833$ which is the exact solution at this point. Indeed, we shall find that the nodal displacement and rotation at all the nodes are exact. This occurs for some approximations to one-dimensional solutions of *static* problems. The original proof was given by Tong [10] and a generalization is summarized in Appendix G.

#### 13.2.3.3 Element arrays: Thick plates

The element arrays for the thick plate case are obtained from (13.6) which we find only has a first derivative of the two dependent variables and thus we require only $C_0$ continuous interpolation for $w$ and $\phi_x$. This suggests we can introduce the isoparametric interpolation given by

$$x = \sum_a N_a(\xi)\, x_a \quad \text{and} \quad \begin{Bmatrix} w \\ \phi \end{Bmatrix} = \sum_a N_a(\xi) \begin{Bmatrix} \tilde{w}_a \\ \tilde{\phi}_a \end{Bmatrix} = \mathbf{N}_a \tilde{\mathbf{u}}_a \tag{13.11}$$

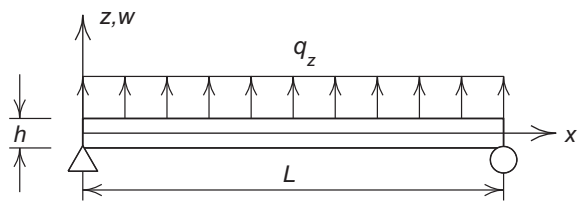



**FIGURE 13.3**
Cylindrical bending of simply supported, uniformly loaded plate.

Inserting into (13.6) yields the element arrays

$$\begin{aligned}
\mathbf{M}^e_{ab} &= \int_{-1}^{1} \begin{bmatrix} N_a & 0 \\ 0 & N_a \end{bmatrix} \begin{bmatrix} \rho h & 0 \\ 0 & \frac{1}{12}\rho h^3 \end{bmatrix} \begin{bmatrix} N_b & 0 \\ 0 & N_b \end{bmatrix} j \, \mathrm{d}\xi \\
\mathbf{K}^e_{ab} &= \int_{-1}^{1} \left[ \begin{Bmatrix} 0 \\ \dfrac{\partial N_a}{\partial x} \end{Bmatrix} [D] \begin{bmatrix} 0 & \dfrac{\partial N_b}{\partial x} \end{bmatrix} + \begin{Bmatrix} \dfrac{\partial N_a}{\partial x} \\ N_a \end{Bmatrix} [\kappa G h] \begin{bmatrix} \dfrac{\partial N_b}{\partial x} & N_b \end{bmatrix} \right] j \, \mathrm{d}\xi \\
\mathbf{f}^e_a &= \int_{-1}^{1} \begin{Bmatrix} N_a \, q_z \\ N_a \, m_x \end{Bmatrix} j \, \mathrm{d}\xi + N_a \begin{Bmatrix} \bar{Q}_x \\ \bar{M}_x \end{Bmatrix} \Bigg|_{\Gamma_L}
\end{aligned} \tag{13.12}$$

The simplest element has only two nodes and the derivative of the shape functions becomes

$$\frac{\partial N_1}{\partial x} = -\frac{1}{L_e} \text{ and } \frac{\partial N_2}{\partial x} = \frac{1}{L_e}$$

The integrals for other products are

$$\int_{-1}^{1} N_a \frac{\partial N_b}{\partial x} \, j \, \mathrm{d}\xi = \begin{cases} -\frac{1}{2}, & b = 1 \\ \frac{1}{2}, & b = 2 \end{cases}$$

$$\int_{-1}^{1} N_a N_b \, j \, \mathrm{d}\xi = \frac{1}{6} L_e \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

Thus, for constant properties over each element the stiffness matrix for the element is given by

$$\mathbf{K}^e = \frac{D}{L_e} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix} + \frac{\kappa G h}{L_e} \begin{bmatrix} 1 & -\frac{1}{2}L_e & -1 & -\frac{1}{2}L_e \\ -\frac{1}{2}L_e & \frac{1}{3}L_e^2 & \frac{1}{2}L_e & \frac{1}{6}L_e^2 \\ -1 & \frac{1}{2}L_e & 1 & \frac{1}{2}L_e \\ -\frac{1}{2}L_e & \frac{1}{6}L_e^2 & \frac{1}{2}L_e & \frac{1}{3}L_e^2 \end{bmatrix} \tag{13.13}$$

#### 13.2.3.4 *Cylindrical bending thick plate*

We again consider the static cylindrical bending behavior of a uniformly loaded simply supported plate as shown in Fig. 13.3. The properties are identical to the thin case except we first consider a case where the thick plate behavior is appropriate[2] and let $h = 20$ (this gives $L/h = 5$). The results are shown in Fig. 13.4(a) by the upper curve where we observe that shear deformation leads to an increase of about 10% in the central displacement. The analysis is repeated for the case where $h = 2$ and we expect to obtain answers where shear deformation is not important. This is indeed nearly obtained using 256 elements, however, the solution converges very slowly—exhibiting the same locking behavior as observed for the nearly incompressible problem using a displacement method with low order elements.

[2]Generally, when $L_c/h > 10$, where $L_c$ is a characteristic length, thin plate theory is applicable.

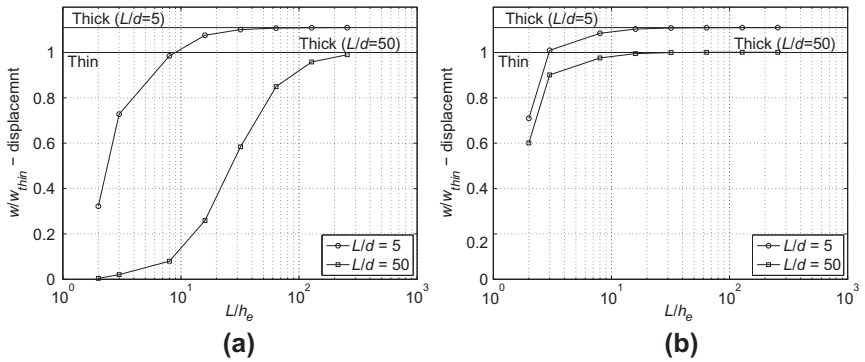



**FIGURE 13.4**

Convergence for cylindrical bending of simply supported, uniformly loaded thick and thin plates: (a) exact integration and (b) reduced quadrature.

This suggests using a reduced integration, which in the present case specifies only one point for the stiffness. This is sufficient as there are two rigid-body modes for an element and one point should add a rank 2 contribution. All terms in the stiffness are exactly integrated by a single point Gauss quadrature except for the product $N_a\, N_b$ where we find

$$\int_{-1}^{1} N_a N_b\, j\, \mathrm{d}\xi \approx \frac{1}{4} L_e \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

which is only rank 1. The analysis is repeated for both thickness cases with results shown in Fig 13.4(b) where we observe that the shear locking effect is no longer present.

Unfortunately, applying reduced quadrature to shear terms in general plates results in an element matrix which does not have proper rank [11] and, thus, we need more robust methods to obtain a method that may be applied to problems for which both thick and thin behavior can occur. For example, if we compute the frequencies and modes for the simply supported cylindrical bending problem for the example problem with $h = 2$ we find that the low modes have frequencies unaffected by shear. However, higher modes can lead to cases where effective $L/h$ is quite small leading to significant shear effects.

### 13.2.4 Exact nodal solution for thick plate

In the thin static solution for cylindrical bending we obtained an exact solution for displacement at every node. This is a result from the Hermite functions satisfying the Tong solution. Using the simple isoparametric solution did not result in an exact solution at nodes as shown in Fig. 13.4. Thus, we seek to find interpolations for $w$ and $\phi_x$ that are independent of material properties and lead to exact nodal solution. The requirement as given in Appendix G is that the interpolation functions satisfy the homogeneous adjoint differential equations of the weak form. However, the form given above in (13.6) for $G(w, \phi)$ leads to self-adjoint differential equations, hence

we can use the homogeneous differential equations to devise our interpolation. We assume constant properties for $D$ and $\kappa Gh$ and, thus, the differential equations are

$$\kappa Gh\left(\frac{\mathrm{d}^2 w}{\mathrm{d}x^2}+\frac{\mathrm{d}\phi_x}{\mathrm{d}x}\right)=0$$
$$D\frac{\mathrm{d}^2\phi_x}{\mathrm{d}x^2}-\kappa Gh\left(\frac{\mathrm{d}w}{\mathrm{d}x}+\phi_x\right)=0$$

A form independent of material parameters may be obtained by differentiating the second equation and adding to the first. This gives, after canceling material parameters

$$\frac{\mathrm{d}^2 w}{\mathrm{d}x^2}+\frac{\mathrm{d}\phi_x}{\mathrm{d}x}=0$$
$$\frac{\mathrm{d}^3 W}{\mathrm{d}x^3}=0$$

A solution to these may be given as

$$\begin{aligned} w(\xi,t) &= \frac{1}{2}(1-\xi)\tilde{w}_1+\frac{1}{2}(1+\xi)\tilde{w}_2 \\ &\quad -\frac{1}{8}L_e(1-\xi^2)\tilde{\phi}_{x1}+\frac{1}{8}L_e(1-\xi^2)\tilde{\phi}_{x2}-\frac{1}{6}L_e\xi(1-\xi^2)\Delta\tilde{\phi}_{x3} \\ \phi_x(\xi,t) &= \frac{1}{2}(1-\xi)\tilde{\phi}_{x1}+\frac{1}{2}(1+\xi)\tilde{\phi}_{x2}+(1-\xi^2)\Delta\tilde{\phi}_{x3} \end{aligned} \tag{13.14}$$

where we use a hierarchical representation for the rotation at an internal node with location $\xi = 0$. We note that the interpolation for $w$ now has parameters from $\phi_x$. We call such interpolations a *linked* form.

Using these functions to solve the example cylindrical bending problem yields, as expected, exact displacement and rotation at all nodes for both thicknesses. Furthermore it is exact when only two elements are used. If the hierarchic term is dropped, the resulting stiffness is identical to the reduced quadrature result for the two-node isoparametric element. However, the load and mass matrix will be different and thus lead to slightly different results than shown in the figure and, thus, are not exact at nodes.

We now turn our attention to the general plate problem. In devising robust elements for thick plate analysis we will employ linked interpolation forms of the type just developed along with some additional forms of interpolation.

## 13.3 General plate theory

The one-dimensional problem of plates and the introduction of thick and thin assumptions translate directly to the general theory of plates. In Fig. 13.5 we illustrate the extensions necessary and write, in place of Eq. (13.1) (assuming in-plane mid-surface displacements $u(x,\ y,\ t)$ and $v(x,\ y,\ t)$ to be zero),

$$u_x = z\,\phi_x(x,\ y,\ t)\quad u_y = z\,\phi_y(x,\ y,\ t)\quad u_z = w(x,\ y,\ t) \tag{13.15}$$

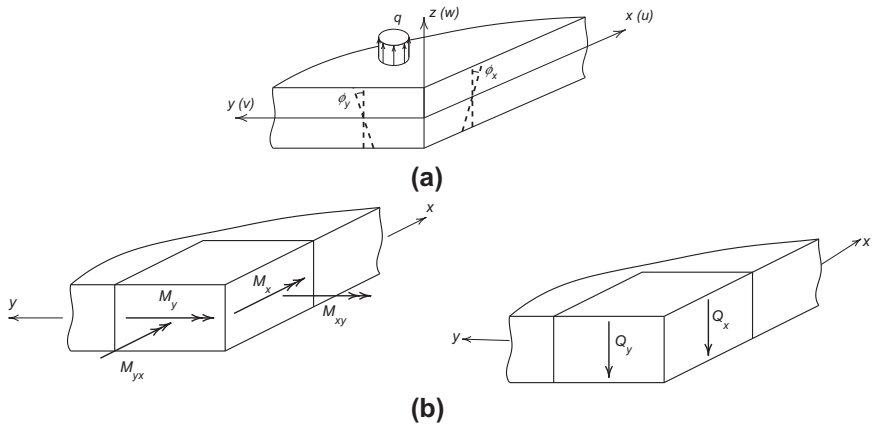



**FIGURE 13.5**
Definitions of variables for plate approximations: (a) displacement, rotation, and loads and (b) force resultants (on negative faces).

where we note that displacement parameters are now functions of $x$ and $y$. It is sometimes advantageous to replace $\phi_x$ and $\phi_y$ by rotations $\theta_x$ and $\theta_y$ about the $x$ and $y$ coordinates. This is accomplished using the relation

$$\begin{Bmatrix} \phi_x \\ \phi_y \end{Bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{Bmatrix} \theta_x \\ \theta_y \end{Bmatrix} \quad \text{or} \quad \begin{Bmatrix} \theta_x \\ \theta_y \end{Bmatrix} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{Bmatrix} \phi_x \\ \phi_y \end{Bmatrix} \qquad (13.16)$$
$$\boldsymbol{\phi} = \mathbf{T}\boldsymbol{\theta}$$

and may be substituted at any time in the development. We do not make this change here as it complicates the signs on expressions for the governing equations. We normally would make the substitution on nodal parameters after shape functions are formulated.

The strains may now be separated into bending and transverse shear groups and we have, in place of Eq. (13.2),

$$\boldsymbol{\epsilon} = \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} = z \begin{bmatrix} \dfrac{\partial}{\partial x} & 0 \\ 0 & \dfrac{\partial}{\partial y} \\ \dfrac{\partial}{\partial y} & \dfrac{\partial}{\partial x} \end{bmatrix} \begin{Bmatrix} \phi_x \\ \phi_y \end{Bmatrix} \equiv z\,\boldsymbol{\mathcal{S}}\boldsymbol{\phi} \qquad (13.17a)$$

and

$$\boldsymbol{\gamma} = \begin{Bmatrix} \gamma_{xz} \\ \gamma_{yz} \end{Bmatrix} = \begin{Bmatrix} \dfrac{\partial w}{\partial x} \\ \dfrac{\partial w}{\partial y} \end{Bmatrix} + \begin{Bmatrix} \phi_x \\ \phi_y \end{Bmatrix} = \nabla w + \boldsymbol{\phi} \qquad (13.17b)$$

The differential operator $\boldsymbol{\mathcal{S}}$ in (13.17a) is identical to the form introduced in Chapter 2 for the plane stress problem, however, here since only the plane stress form is used we remove the zero row for convenience.

We note that now in addition to normal bending moments $M_x$ and $M_y$, defined by expression (13.3a) for the $x$ and $y$ directions, respectively, a twisting moment arises defined by

$$M_{xy} = \int_{-h/2}^{h/2} \tau_{xy}\, Z \mathrm{d}z \tag{13.18}$$

Introducing appropriate constitutive relations, all moment components can be related to displacement derivatives. For isotropic elasticity we can thus write, in place of Eq. (13.3a),

$$\mathbf{M} = \begin{Bmatrix} M_x \\ M_y \\ M_{xy} \end{Bmatrix} = \mathbf{D}\,\mathcal{S}\boldsymbol{\phi} \tag{13.19a}$$

where, assuming plane stress behavior in each layer,

$$\mathbf{D} = D \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix} \tag{13.19b}$$

in which $\nu$ is Poisson's ratio and $D$ is defined by the second of Eqs. (13.3b). Further, the shear force resultants are

$$\mathbf{Q} = \begin{Bmatrix} Q_x \\ Q_y \end{Bmatrix} = \boldsymbol{\alpha}\,(\boldsymbol{\nabla} w + \boldsymbol{\phi}) \tag{13.19c}$$

For isotropic elasticity

$$\boldsymbol{\alpha} = \kappa\, G\, h\, \mathbf{I} \tag{13.19d}$$

where $\mathbf{I}$ is a $2 \times 2$ identity matrix (though here we deliberately have not related $G$ to $E$ and $\nu$ to allow for possibly different shear rigidities).

Of course, the constitutive relations can be simply generalized to anisotropic or inhomogeneous behavior such as can be manifested if several layers of materials are assembled symmetrically to form a *composite*. The only apparent difference is the structure of the $\mathbf{D}$ and $\boldsymbol{\alpha}$ matrices, which can always be found by simple integration.

The governing equations of thick and thin plate behavior are completed by writing the equilibrium relations. Again omitting the "in-plane" behavior we have, in place of Eq. (13.4b),

$$\begin{bmatrix} \dfrac{\partial}{\partial x} & \dfrac{\partial}{\partial y} \end{bmatrix} \begin{Bmatrix} Q_x \\ Q_y \end{Bmatrix} + q \equiv \boldsymbol{\nabla}^{\mathrm{T}}\mathbf{Q} + q = \rho\, h\, \ddot{w} \tag{13.20a}$$

and, in place of Eq. (13.4c),

$$\begin{bmatrix} \dfrac{\partial}{\partial x} & 0 & \dfrac{\partial}{\partial y} \\ 0 & \dfrac{\partial}{\partial y} & \dfrac{\partial}{\partial x} \end{bmatrix} \begin{Bmatrix} M_x \\ M_y \\ M_{xy} \end{Bmatrix} - \begin{Bmatrix} Q_x \\ Q_y \end{Bmatrix} = \frac{1}{12}\,\rho\, h^3 \begin{Bmatrix} \ddot{\phi}_x \\ \ddot{\phi}_y \end{Bmatrix} \tag{13.20b}$$

$$\mathcal{S}^{\mathrm{T}}\mathbf{M} - \mathbf{Q} = \frac{1}{12}\,\rho\, h^3\, \ddot{\boldsymbol{\phi}}$$

Equations (13.19a)–(13.20b) are the basis from which the solution of both thick and thin plates can start. For thick plates any (or all) of the independent variables can be

approximated independently, leading to a mixed formulation which we shall discuss in Section 13.7 of this chapter.

For thin plates in which the shear deformations are suppressed Eq. (13.19c) is rewritten as

$$\nabla w + \boldsymbol{\phi} = \mathbf{0} \quad \text{or} \quad \boldsymbol{\phi} = -\nabla w \tag{13.21}$$

and the strain-displacement relations (13.17a) become

$$\boldsymbol{\epsilon} = -z\,\mathcal{S}\nabla w = -z \begin{Bmatrix} \dfrac{\partial^2 w}{\partial x^2} \\ \dfrac{\partial^2 w}{\partial y^2} \\ 2\dfrac{\partial^2 w}{\partial x \partial y} \end{Bmatrix} = -z\,\mathcal{L} w = -z\,\boldsymbol{\chi} \tag{13.22}$$

where $\boldsymbol{\chi}$ is the matrix of *changes in curvature* of the plate and using Eq. (13.22),

$$\mathcal{L} = (\mathcal{S}\nabla) = \begin{bmatrix} \dfrac{\partial^2}{\partial x^2} & \dfrac{\partial^2}{\partial y^2} & 2\dfrac{\partial^2}{\partial x \partial y} \end{bmatrix}^{\mathrm{T}}$$

Using the above form for the thin plate, both irreducible and mixed forms can now be written. In particular, for the thin plate problem it is an easy matter to eliminate $\mathbf{M}$, $\mathbf{Q}$, and $\boldsymbol{\phi}$ and leave only $w$ as the variable.

Applying the operator $\nabla^{\mathrm{T}}$ to expression (13.20b), inserting Eqs. (13.19a) and (13.20a), ignoring rotatory inertia, and finally replacing $\boldsymbol{\phi}$ by the use of Eq. (13.21) gives a scalar equation

$$-\,\mathcal{L}^{\mathrm{T}}\mathbf{D}\,\mathcal{L} w + q = \rho h \ddot{w} \tag{13.23a}$$

In the case of isotropy with constant bending stiffness $D$ this becomes the well-known biharmonic equation of plate flexure

$$-\,D\left(\frac{\partial^4 w}{\partial x^4} + 2\frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4}\right) + q = -\,D\,\nabla^4 w + q = \rho h\,\ddot{w} \tag{13.23b}$$

### 13.3.1 The boundary conditions

The boundary conditions which have to be imposed on the problem (see Figs. 13.2 and 13.6) include the following classical conditions.

1. *Fixed boundary*, where displacements on restrained parts of the boundary are given specified values.[3] These conditions are expressed as

$$w = \bar{w}, \quad \phi_n = \bar{\phi}_n, \text{ and } \phi_s = \bar{\phi}_s$$

Here $n$ and $s$ are directions normal and tangential to the boundary curve of the middle surface. A *clamped edge* is a special case with zero values assigned.

[3]Note that in thin plates the specification of $w$ along $s$ automatically specifies $\phi_s$ by Eq. (13.21), but this is not the case in thick plates where the quantities are independently prescribed.

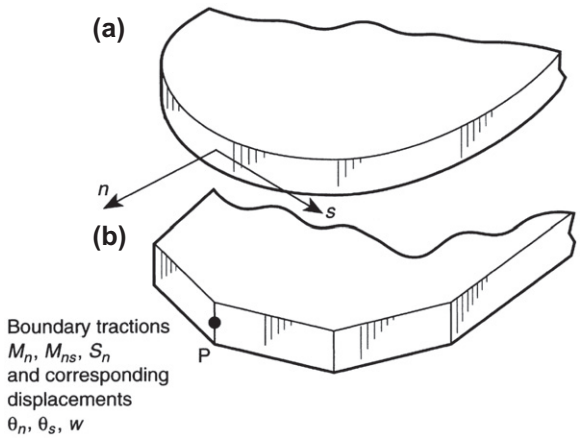



**FIGURE 13.6**

Boundary traction and conjugate displacement. Note: The simply supported condition requiring $M_n = 0$, $\phi_s = 0$, and $w = 0$ is identical at a corner node to specifying $\phi_n = \phi_s = 0$, that is, a clamped support. This leads to a paradox if a curved boundary (a) is modeled as a polygon (b).

2. *Traction boundary*, where stress resultants $M_n$, $M_{ns}$, and $Q_n$ (conjugate to the displacements $\phi_n$, $\phi_s$, and $w$) are given prescribed values.

$$M_n = \bar{M}_n, \quad M_{ns} = \bar{M}_{ns}, \quad \text{and} \quad Q_n = \bar{Q}_n$$

   A *free edge* is a special case with zero values assigned.

3. *"Mixed" boundary conditions*, where both traction and displacement components can be specified. Typical here is the *simply supported edge* (see Fig. 13.2). For this, clearly, $M_n = 0$ and $w = 0$, but it is less clear whether $M_{ns}$ or $\phi_s$ needs to be given. Specification of $M_{ns} = 0$ is *physically* a more acceptable condition. This should be adopted for thick plates. Thus options for simply supported edges are

| Type | Conditions | | |
|---|---|---|---|
| SS1 | $w = 0,$ | $M_n = 0,$ | $M_{ns} = 0$ |
| SS2 | $w = 0,$ | $M_n = 0,$ | $\phi_s = 0$ |

In thin plates $\phi_s$ is automatically specified from $w$ and we shall find certain difficulties, and indeed anomalies, associated with this assumption [12,13]. For instance, in Fig. 13.6 we see how a specification of $\phi_s = 0$ at corner nodes implicit in thin plates formally leads to the prescription of all boundary parameters, which is identical to boundary conditions of a clamped plate for this point. It is well known in thin plate theory that the limit of increasing the number of sides on a polygonal plate with simply supported edges does not yield the result for a simply supported circular plate [14]. In the finite element literature this same result is referred to as the "Babuška

paradox" [15]. Thus, for a curved simply supported edge a unique normal to each node must be specified and used to specify one of the mixed conditions given above.

### 13.3.2 The irreducible, thin plate approximation

The thin plate approximation when cast in terms of a single variable $w$ is clearly irreducible and is in fact typical of a displacement formulation. The Eqs. (13.20a) and (13.20b) can be written together, omitting rotatory inertia, as

$$\boldsymbol{\mathcal{L}}^{\mathrm{T}}\mathbf{M} + q = \rho\, h\, \ddot{w} \tag{13.24}$$

and the constitutive relation (13.19a) can be recast by using Eq. (13.21) as

$$\mathbf{M} = -\,\mathbf{D}\,\boldsymbol{\mathcal{L}} w \tag{13.25}$$

The derivation of the finite element equations can be obtained from a weak form of Eq. (13.24) as

$$\mathrm{G}(\delta w, w, \mathbf{M}) = \int_{\Omega} \delta w \left[\rho\, h\, \ddot{w} - \boldsymbol{\mathcal{L}}^{\mathrm{T}}\mathbf{M} - q\right] \mathrm{d}\Omega = 0 \tag{13.26}$$

where $\Omega$ denotes the area of the plate reference (middle) surface. After integration by parts and introducing traction boundary conditions the weak form becomes

$$\begin{aligned}\mathrm{G}(\delta w, w, \mathbf{M}) = &\int_{\Omega} \delta w \left[\rho\, h\, \ddot{w} - q\right] \mathrm{d}\Omega - \int_{\Omega} (\boldsymbol{\mathcal{L}}\delta w)^{T}\mathbf{M}\, \mathrm{d}\Omega \\ &- \int_{\Gamma_n} \delta\phi_n\, \bar{M}_n\, \mathrm{d}\Gamma - \int_{\Gamma_s} \delta\phi_s\, \bar{M}_{ns}\, \mathrm{d}\Gamma - \int_{\Gamma_q} \delta w\, \bar{Q}_n\, \mathrm{d}\Gamma = 0\end{aligned} \tag{13.27}$$

where $\bar{M}_n$, $\bar{M}_{ns}$, $\bar{Q}_n$ are specified values and $\Gamma_n$, $\Gamma_s$, and $\Gamma_q$ are parts of the boundary where each component is specified. For thin plates with straight edges Eq. (13.21) gives immediately $\phi_s = -\,\partial w/\partial s$ and thus the last two terms above may be combined ($\Gamma_s = \Gamma_q$) as

$$\int_{\Gamma_s} \delta\phi_s \bar{M}_{ns}\, \mathrm{d}\Gamma + \int_{\Gamma_q} \delta w\, \bar{Q}_n\, \mathrm{d}\Gamma = \int_{\Gamma_q} \delta w\, \bar{V}_n\, \mathrm{d}\Gamma + \sum_i \delta w_i\, R_i \tag{13.28}$$

where

$$\bar{V}_n = \bar{Q}_n + \frac{\partial \bar{M}_{ns}}{\partial s} \tag{13.29}$$

is an "ersatz boundary shear" and $R_i$ are concentrated forces arising at locations where corners exist (see Fig. 13.2) [2].

Introducing the constitutive Eq. (13.25) into (13.27) and using (13.28) gives the irreducible weak form in terms of $w$ as

$$\begin{aligned}\mathrm{G}(\delta w, w) = &\int_{\Omega} \delta w \left[\rho\, h\, \ddot{w} - q\right] \mathrm{d}\Omega + \int_{\Omega} (\boldsymbol{\mathcal{L}}\delta w)^{T}\mathbf{D}\boldsymbol{\mathcal{L}} w\, \mathrm{d}\Omega \\ &+ \int_{\Gamma_n} \frac{\partial \delta w}{\partial n}\, \bar{M}_n\, \mathrm{d}\Gamma - \int_{\Gamma_q} \delta w\, \bar{V}_n\, \mathrm{d}\Gamma - \sum_i \delta w_i\, R_i = 0\end{aligned} \tag{13.30}$$

For the static problem (13.7) is equivalent to minimization of the total potential energy

$$\Pi = \frac{1}{2}\int_\Omega (\mathcal{L}w)^{\mathrm{T}}\mathbf{D}\,\mathcal{L}w\,\mathrm{d}\Omega - \int_\Omega w\,q\,\mathrm{d}\Omega + \int_{\Gamma_n} \frac{\partial w}{\partial n}\,\bar{M}_n\,\mathrm{d}\Gamma - \int_{\Gamma_q} w\,\bar{V}_n\,\mathrm{d}\Gamma - \sum_i w_i R_i = \text{minimum} \tag{13.31}$$

as the reader can verify by taking the first variation.

### 13.3.3 Finite element approximation

Substituting into (13.30) the discretization

$$w = \mathbf{N}(x, y)\tilde{\mathbf{u}}(t) \quad \text{and} \quad \delta w = \mathbf{N}(x, y)\delta\tilde{\mathbf{u}} \tag{13.32}$$

where $\mathbf{N}$ are $\tilde{\mathbf{u}}$ are appropriate shape functions and parameters, respectively, we can obtain a standard displacement approximation expressed in semi-discrete form as

$$\mathbf{M}\ddot{\tilde{u}} + \mathbf{K}\tilde{\mathbf{u}} = \mathbf{f} \tag{13.33}$$

with the mass, stiffness, and load arrays given by

$$\begin{aligned} \mathbf{M} &= \int_\Omega \mathbf{N}^T\,\rho\,h\,\mathbf{N}\,\mathrm{d}\Omega \\ \mathbf{K} &= \int_\Omega \mathbf{B}^{\mathrm{T}}\mathbf{D}\mathbf{B}\,\mathrm{d}\Omega \\ \mathbf{f} &= \int_\Omega \mathbf{N}^{\mathrm{T}} q\,\mathrm{d}\Omega + \mathbf{f}_{\mathrm{b}} \end{aligned} \tag{13.34}$$

where $\mathbf{f}_{\mathrm{b}}$ is the boundary contribution

$$\mathbf{f}_{\mathrm{b}} = \int_{\Gamma_s} \mathbf{N}\,\bar{V}_n\,\mathrm{d}\Gamma - \int_{\Gamma_n} \frac{\partial \mathbf{N}}{\partial n}\,\bar{M}_n\,\mathrm{d}\Gamma + \sum_i \mathbf{N}(x_i, y_i)\,R_i \tag{13.35}$$

and

$$\mathbf{B} = -\mathcal{L}\mathbf{N} = -\begin{Bmatrix} \dfrac{\partial^2 \mathbf{N}}{\partial x^2} \\ \dfrac{\partial^2 \mathbf{N}}{\partial y^2} \\ 2\dfrac{\partial^2 \mathbf{N}}{\partial x\,\partial y} \end{Bmatrix} \tag{13.36}$$

The reader will recognize in the above the well-known ingredients of a displacement formulation and the procedures are almost automatic once $\mathbf{N}$ is chosen.

### 13.3.4 Continuity requirement for shape functions ($C_1$ continuity)

The weak form given above contains all the second derivatives of $w$, thus, it is clear that at least all the first derivative of $w$ must be continuous in $\Omega$. As noted above this class of functions is called $C_1$ continuous. If the first derivatives are not continuous the plate has "kinks" resulting in singular second derivatives (similar to a Kronnecker delta along a line) and, thus, the square of a singular function is not integrable in $\Omega$.

We have shown above that using Hermite interpolation it is not difficult to construct piecewise polynomials along a line that are $C_1$ continuous. We also noted that Hermite interpolation is not isoparametric and this causes difficulty when we attempt to extend the use of Hermite-based shape functions to general shaped elements for a thin plate formulation.

To ensure the continuity of both $w$ and its normal slope across an interface we must have both $w$ and $\partial w/\partial n$ uniquely defined by values of nodal parameters along such an interface. Consider Fig. 13.7 depicting the side 1–2 of a rectangular element. The normal direction $n$ is in fact that of $y$ and we desire $w$ and $\partial w/\partial y$ to be uniquely determined by values of $w$, $\partial w/\partial x$, $\partial w/\partial y$ at the nodes lying along this line.

To show the $C_1$ continuity along side 1–2, we write

$$w = A_1 + A_2 x + A_3 y + \cdots \tag{13.37}$$

and

$$\frac{\partial w}{\partial y} = B_1 + B_2 x + B_3 y + \cdots \tag{13.38}$$

with a number of constants in each expression just sufficient to determine a unique solution for the nodal parameters associated with the line.

Thus, for instance, if only two nodes are present a cubic variation of $w$ should be permissible noting that $\partial w/\partial x$ and $w$ are specified at each node. Similarly, for the two nodes only a linear, or two term, variation of $\partial w/\partial y$ would be permissible.

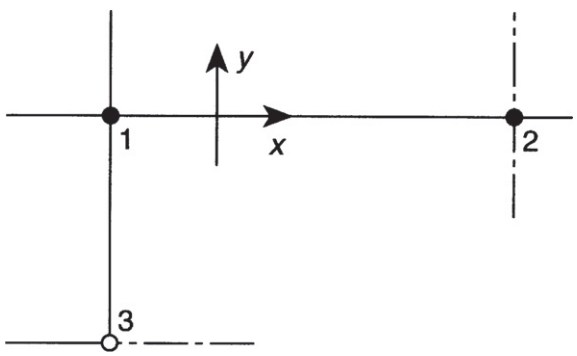



**FIGURE 13.7**
Continuity requirement for normal slopes.

Note, however, that a similar exercise could be performed along the side placed in the $y$ direction preserving continuity of $\partial w/\partial x$ along this. Along side 1–2 we thus have $\partial w/\partial y$, depending on nodal parameters of line 1–2 only, and along side 1–3 we have $\partial w/\partial x$, depending on nodal parameters of line 1–3 only. Differentiating the first with respect to $x$, on line 1–2 we have $\partial^2 w/\partial x \partial y$, depending on nodal parameters of line 1–2 only, and similarly, on line 1–3 we have $\partial^2 w/\partial y \partial x$, depending on nodal parameters of line 1–3 only.

At the common point, 1, an inconsistency arises immediately as we cannot automatically have there the necessary identity for continuous functions

$$\frac{\partial^2 w}{\partial x \partial y} \equiv \frac{\partial^2 w}{\partial y \partial x} \tag{13.39}$$

for arbitrary values of the parameters at nodes 2 and 3. *It is thus impossible to specify simple polynomial expressions for shape functions ensuring full compatibility when only* w *and its slopes are prescribed at corner nodes* [16].

Thus if any functions satisfying the compatibility are found with the three nodal variables, they must be such that at corner nodes these functions are not continuously differentiable and the cross-derivative is not unique. The above proof has been given for a rectangular element. Clearly, the arguments can be extended for any two arbitrary directions of interface at the corner node 1. Some elements for use in thin plates with corner discontinuous functions have been developed by Clough et al. [17,18], Fraeijs de Veubeke [19,20], Bazeley et al. [21], and Irons [22].

A way out of this difficulty appears to be obvious. We could specify the cross-derivative as one of the nodal parameters. For an assembly of rectangular elements, this is convenient and indeed permissible. Simple functions of this type have been developed by Bogner et al. [23] and used with success in problems where square elements are adequate. For the geometry of a rectangular element as shown in Fig. 13.8, the shape functions may be taken as combinations of the Hermite interpolations

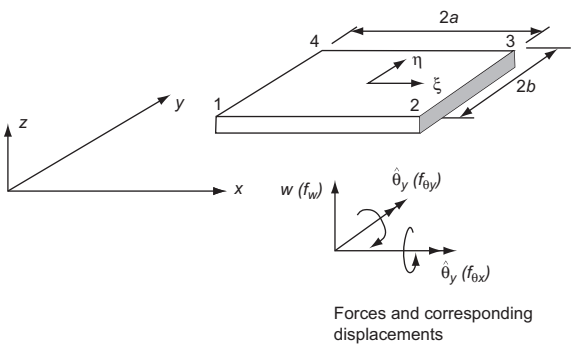



**FIGURE 13.8**

A rectangular plate element.

expressed as

$$\hat{w}(\xi,\eta)=\sum_{a=1}^{4}\mathbf{N}_a(\xi,\eta)\,\tilde{\mathbf{u}}_a \tag{13.40a}$$

where

$$\mathbf{N}_a=\left[\,H_a^w(\xi)H_a^w(\eta)\,,\;H_a^\theta(\xi)H_a^w(\eta)\,,\;H_a^w(\xi)H_a^\theta(\eta)\,,\;H_a^\theta(\xi)H_a^\theta(\eta)\,\right] \tag{13.40b}$$

and

$$\tilde{\mathbf{u}}_a=\left[\,\tilde{w}_a\,,\;\frac{\partial\tilde{w}_a}{\partial x}\;\;\frac{\partial\tilde{w}_a}{\partial y}\;\;\frac{\partial^2\tilde{w}_a}{\partial x\partial y}\,\right]^{\mathrm{T}} \tag{13.40c}$$

and the geometry is given by

$$\mathbf{x}=\sum_{a=1}^{4}N_a(\xi,\eta)\,\mathbf{x}_a \tag{13.41}$$

with node and function ordering as shown in Table 13.2.

Indeed this element is the most accurate rectangular element available for thin problems. A development of this type of element to include continuity of higher derivatives is simple using Hermite interpolation and is outlined in Ref. [24].

In their undistorted form the element is, as for all rectangles, of very limited applicability. Unfortunately, the extension to nodes at which a number of element interfaces meet with different angles (Fig. 13.9) is not, in general, permissible since the element is not of isoparametric type. Here, the continuity of cross-derivatives in several sets of orthogonal directions implies, in fact, a specification of *all second derivatives at a node*.

This, however, violates physical requirements if the plate stiffness varies abruptly from element to element, for then equality of moments normal to the interfaces cannot be maintained. This process has been used with some success in homogeneous plate situations [24–31] although Smith and Duncan [24] comment adversely on the effect of imposing such *excessive continuities* on several orders of higher derivatives. For cases where thickness or material changes occur, it is possible to transform the nodal parameters to be normal and tangential to the interface and relax continuity on the parameters $\partial^2 w_a/\partial n^2$.

The difficulties of finding compatible displacement functions have led to many attempts at ignoring the complete slope continuity while still continuing with the

**Table 13.2** Node and Function for Rectangular Element

| | Nodes | | | |
|---|---|---|---|---|
| ***a*** | **1** | **2** | **3** | **4** |
| | **Function** | | | |
| $H(\xi)$ | 1 | 2 | 2 | 1 |
| $H(\eta)$ | 1 | 1 | 2 | 2 |

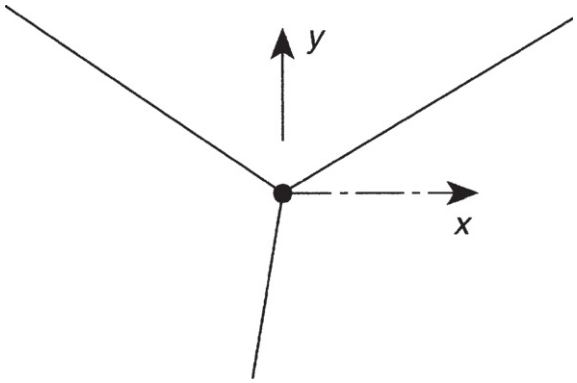



**FIGURE 13.9**

Nodes where elements meet in arbitrary directions.

other necessary criteria. Proceeding perhaps from a naive but intuitive idea that the imposition of slope continuity at nodes only must, in the limit, lead to a complete slope continuity, several successful, "nonconforming," elements have been developed [21,32–46].

The convergence of such elements is not obvious but can be proved either by application of the patch test or by comparison with finite difference algorithms. We have discussed the importance of the patch test extensively in Chapter 8, and for plate problems, its importance in both design and testing of elements is paramount and this test should never be omitted. Indeed, we shall show how a successful triangular element is developed via this analytical interpretation [47–52].

## 13.4 The patch test: An analytical requirement

The patch test in its different forms is generally applied numerically to test the final form of an element. However, the basic requirements for its satisfaction by shape functions that violate compatibility can be forecast accurately if certain conditions are satisfied in the choice of such functions. These conditions follow from the requirement that for constant strain states the work done by internal forces acting at the discontinuity must be zero. Thus if the tractions acting on an element interface of a plate are (see Fig. 13.6)

$$M_n,\ M_{ns},\ \text{and}\ Q_n \tag{13.42}$$

and if the corresponding mismatch of virtual displacements are

$$\Delta\phi_n \equiv \Delta\left(\frac{\partial w}{\partial n}\right),\ \Delta\phi_s \equiv \Delta\left(\frac{\partial w}{\partial s}\right),\ \text{and}\ \Delta w \tag{13.43}$$

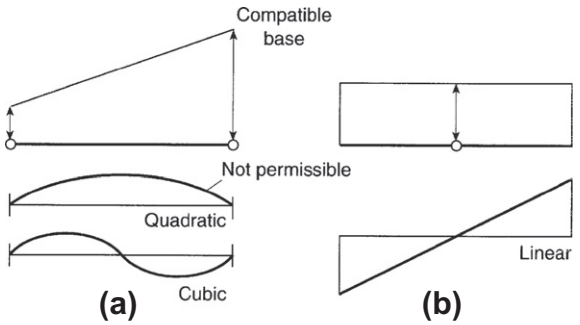



**FIGURE 13.10**

Continuity condition for satisfaction of patch test [$\int (\partial w/\partial n)\ \mathrm{d}s = 0$]; variation of $\partial w/\partial n$ along side: (a) definition by corner nodes (linear component compatible) and (b) definition by one central node (constant component compatible).

then ideally we would like the integral given below to be zero for all constant stress states:

$$\int_{\Gamma_e} M_n \Delta\phi_n \,\mathrm{d}\Gamma + \int_{\Gamma_e} M_{ns} \Delta\phi_s \,\mathrm{d}\Gamma + \int_{\Gamma_e} Q_n \Delta w \,\mathrm{d}\Gamma = 0 \tag{13.44}$$

The last term will always be zero identically for constant $M_x$, $M_y$, $M_{xy}$ fields as then $Q_x = Q_y = 0$ [in the absence of applied couples, see Eq. (13.20b) for the static problem] and we can ensure the satisfaction of the remaining conditions if

$$\int_{\Gamma_e} \Delta\phi_n \,\mathrm{d}\Gamma = 0 \quad \text{and} \quad \int_{\Gamma_e} \Delta\phi_s \,\mathrm{d}\Gamma = 0 \tag{13.45}$$

is satisfied for each straight side $\Gamma_e$ of the element.

For elements joining at vertices where $\partial w/\partial n$ is prescribed, these integrals will be identically zero if only anti-symmetric cubic terms arise in the departure from linearity and the quadratic variation of normal gradients is absent, as shown in Fig. 13.10(a). The satisfaction of the second condition of Eq. (13.45) is always ensured if the function $w$ is specified solely by parameters along the edge under consideration.

## 13.5 A nonconforming three-node triangular element

The use of area coordinates are a natural choice for triangles, see Fig. 13.11. There are alternate ways of specifying complete polynomials of any order. For instance,

$$\alpha_1 L_1 + \alpha_2 L_2 + \alpha_3 L_3 \tag{13.46}$$

gives the three terms of a *complete* linear polynomial and

$$\alpha_1 L_1^2 + \alpha_2 L_2^2 + \alpha_3 L_3^2 + \alpha_4 L_1 L_2 + \alpha_5 L_2 L_3 + \alpha_6 L_3 L_1 \tag{13.47}$$

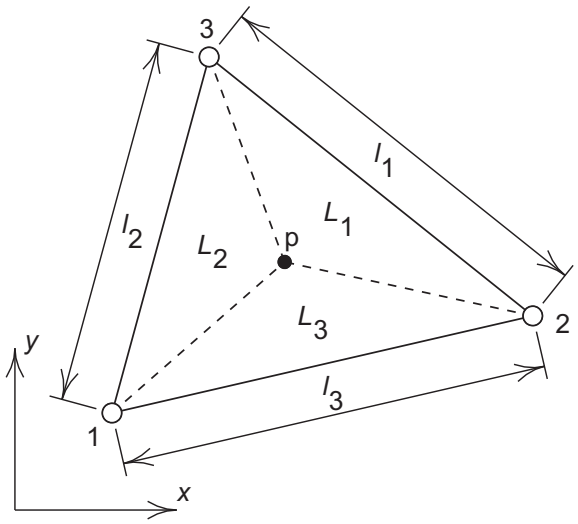



**FIGURE 13.11**
Area coordinates and side lengths for three-node triangle.

gives all six terms of a quadratic (containing within it the linear terms). However, it is also possible to write a complete quadratic as

$$\alpha_1 L_1 + \alpha_2 L_2 + \alpha_3 L_3 + \alpha_4 L_1 L_2 + \alpha_5 L_2 L_3 + \alpha_6 L_3 L_1$$

and so on, for higher orders. This has the advantage of explicitly stating all retained terms of polynomials of lower order. The 10 terms of a cubic expression are similarly formed by the products of all possible cubic combinations, that is,

$$L_1^3,\ L_2^3,\ L_3^3,\ L_1^2 L_2,\ L_1^2 L_3,\ L_2^2 L_3,\ L_2^2 L_1,\ L_3^2 L_1,\ L_3^2 L_2,\ L_1 L_2 L_3 \qquad (13.48)$$

For an element envisaged with nine degrees of freedom we must ensure that all six quadratic terms are present. The quadratic terms ensure that a constant curvature, necessary for patch test satisfaction, is possible. To obtain the additional three terms we add the following terms

$$L_1^2 L_2,\ L_2^2 L_3,\ L_3^2 L_1,\ L_1^2 L_2 L_3,\ L_1 L_2^2 L_3,\ L_1 L_2 L_3^2$$

in which we note the last three quartic terms add to give the missing cubic term. With these 12 terms collected in the vector $\hat{\mathbf{P}}$ we write the interpolation as

$$w = \hat{\mathbf{P}}(L_i)\,\hat{\boldsymbol{\alpha}} \qquad (13.49)$$

The values for $\hat{\boldsymbol{\alpha}}$ are obtained by evaluating (13.49) at the three nodes in terms of

$$\left[\,\tilde{w}_a,\ -\tilde{\phi}_{xa},\ -\tilde{\phi}_{ya}\,\right] = \left[\,\tilde{w}_a,\ \left.\frac{\partial \tilde{w}}{\partial x}\right|_a,\ \left.\frac{\partial \tilde{w}}{\partial y}\right|_a\,\right], \quad a = 1, 2, 3$$

and adding a constraint to remove the quadratic part of the normal derivative along the three edges. The normal derivative may be expressed by noting that

$$\frac{\partial w}{\partial n_a} = -\frac{1}{l_a}\begin{bmatrix} b_a & c_a \end{bmatrix}\begin{Bmatrix} \dfrac{\partial w}{\partial x} \\ \dfrac{\partial w}{\partial y} \end{Bmatrix} \tag{13.50}$$

where

$$\begin{Bmatrix} \dfrac{\partial w}{\partial x} \\ \dfrac{\partial w}{\partial y} \end{Bmatrix} = \begin{bmatrix} \dfrac{\partial L_1}{\partial x} & \dfrac{\partial L_2}{\partial x} & \dfrac{\partial L_3}{\partial x} \\ \dfrac{\partial L_1}{\partial y} & \dfrac{\partial L_2}{\partial y} & \dfrac{\partial L_3}{\partial y} \end{bmatrix}\begin{Bmatrix} \dfrac{\partial w}{\partial L_1} \\ \dfrac{\partial w}{\partial L_2} \\ \dfrac{\partial w}{\partial L_3} \end{Bmatrix} = \frac{1}{2\Delta}\begin{bmatrix} b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}\begin{Bmatrix} \dfrac{\partial w}{\partial L_1} \\ \dfrac{\partial w}{\partial L_2} \\ \dfrac{\partial w}{\partial L_3} \end{Bmatrix} \tag{13.51}$$

in which $b_a$, $c_a$, and $\Delta$ are the same as given in Section 5.1.2 and $l_a$ is the length of side $a$ (see Fig. 13.11). After solving for $\hat{\boldsymbol{\alpha}}$ we can write nine functions as

$$\begin{aligned} P_a &= L_a \\ P_{a+3} &= L_a L_b \\ P_{a+6} &= L_a^2 L_b + \frac{1}{2} L_1 L_2 L_3 \left[3(1+\mu_c)L_a - (1-3\mu_c)L_b + (1-\mu_c)L_c\right] \end{aligned} \tag{13.52}$$

where

$$\mu_a = \frac{l_b^2 - l_c^2}{l_a^2}$$

These polynomials are used to define the nine shape functions

$$\mathbf{N}_a^{\mathrm{T}} = \begin{Bmatrix} P_a - P_{a+3} + P_{c+3} + 2(P_{a+6} - P_{c+6}) \\ b_b(P_{c+6} - P_{c+3}) + b_c P_{a+6} \\ c_b(P_{c+6} - P_{c+3}) + c_c P_{a+6} \end{Bmatrix}, \quad a = 1,\ 2,\ 3 \tag{13.53}$$

in which $a, b, c$ are cyclic permutations.

The computation of stiffness and load matrices can again follow the standard patterns, and integration of (13.34) can be done exactly using the general integrals given in Fig. 13.11. However, numerical quadrature is generally used and proves equally efficient (see Section 6.8). The stiffness matrix requires computation of second derivatives of shape functions and these may be conveniently obtained from

$$\begin{bmatrix} \dfrac{\partial^2 N_a}{\partial x^2} & \dfrac{\partial^2 N_a}{\partial x \partial y} \\ \dfrac{\partial^2 N_a}{\partial y \partial x} & \dfrac{\partial^2 N_a}{\partial y^2} \end{bmatrix} = \frac{1}{4\Delta^2}\begin{bmatrix} b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}\begin{bmatrix} \dfrac{\partial^2 N_a}{\partial L_1^2} & \dfrac{\partial^2 N_a}{\partial L_1 \partial L_2} & \dfrac{\partial^2 N_a}{\partial L_1 \partial L_3} \\ \dfrac{\partial^2 N_a}{\partial L_2 \partial L_1} & \dfrac{\partial^2 N_a}{\partial L_2^2} & \dfrac{\partial^2 N_a}{\partial L_2 \partial L_3} \\ \dfrac{\partial^2 N_a}{\partial L_3 \partial L_1} & \dfrac{\partial^2 N_a}{\partial L_3 \partial L_2} & \dfrac{\partial^2 N_a}{\partial L_3^2} \end{bmatrix} \times \begin{bmatrix} b_1 & c_1 \\ b_2 & c_2 \\ b_3 & c_3 \end{bmatrix} \tag{13.54}$$

where $N_{a,xx} = \frac{\partial\, d^2 N_a}{\partial\, dx^2}$ and $N_{a,11} = \frac{\partial\, d^2 N_a}{\partial\, dL_1^2}$, etc. In which $N_a$ denotes any of the shape functions given in Eq. (13.53).

The element just derived is one first developed in Ref. [52]. It satisfies the constant strain criterion (being able to produce constant curvature states) and passes the patch test for arbitrary mesh configurations.

## 13.6 Numerical example for thin plates

The plate bending elements are used to solve two classical plate bending problems to demonstrate the results that can be attained using thin plate theory.

The first example is a square plate with simply supported boundaries and subjected to a uniform loading. The plate has side dimensions of 10 units, a thickness $h$ of 1 unit, elastic properties $E = 10.92$ and $\nu = 0.3$, and a uniform loading $q = 1$. A uniform mesh of $n \times n$ units is solved for both the rectangular four-node element and for the three-node nonconforming element. The error in energy

$$E_{error} = \frac{|E_{ex} - E_{fe}|}{E_{ex}}$$

is shown in Fig. 13.12(a).

For the second analysis we repeat the calculation using the same data but with clamped edges on all sides. The solution of a clamped plate subjected to uniform loading $q$ was a topic of considerable study during the early 1900s [53,54]. An accurate numerical solution in series form was given by Hencky [55,56] and recently evaluated using the form given by Wojtaszak [57] by Taylor and Govindjee [58] to obtain correct solution values to several significant figures. The error in energy is shown in Fig. 13.12(b).

Figure 13.13 shows the convergence for deflections and moments at the center of a square plate clamped along its edges and solved by the use of the rectangular element derived in Section 13.3 and a uniform mesh. Table 13.3 gives numerical results for a

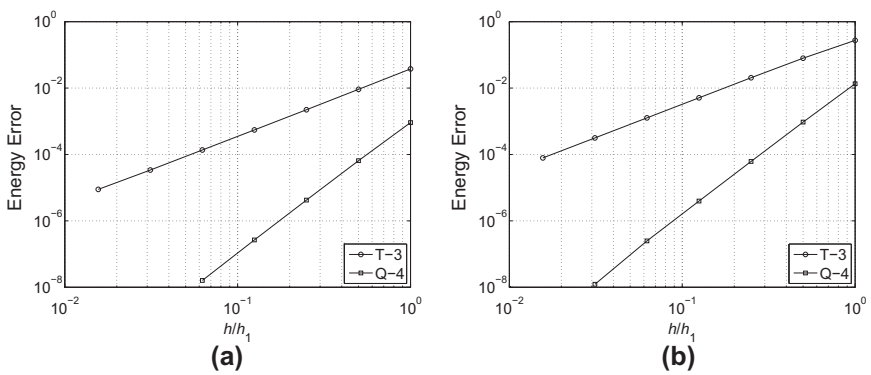



**FIGURE 13.12**
Uniformly loaded square plate: (a) simply supported edges and (b) clamped edges.

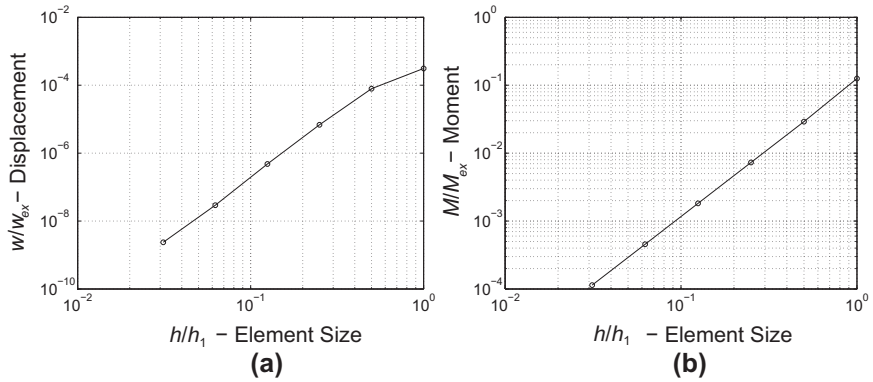



**FIGURE 13.13**

A square plate with clamped edges, uniform load $q$, and square elements: (a) center displacement and (b) center moment.

set of similar examples solved with the same element, and Table 13.4 presents another square plate with more complex boundary conditions [45]. Exact results are available here and comparisons are made [59,60].

## 13.7 Thick plates

We have already introduced in Section 13.3 the full theory of thick plates from which the thin plate, Kirchhoff, theory arose as the limiting case. In this section we shall show

**Table 13.3** Computed Central Deflection of a Square Plate for Several Meshes (Rectangular Elements)

| | | Simply Supported Plate | | Clamped Plate | |
|---|---|---|---|---|---|
| **Mesh** | **Number of Nodes** | $\alpha^a$ | $\beta^b$ | $\alpha^a$ | $\beta^b$ |
| $2 \times 2$ | 9 | 0.004065332 | 0.01147244 | 0.001264924 | 0.005485489 |
| $4 \times 4$ | 25 | 0.004062525 | 0.01156871 | 0.001265219 | 0.005579713 |
| $8 \times 8$ | 81 | 0.004062363 | 0.01159282 | 0.001265310 | 0.005603985 |
| $16 \times 16$ | 256 | 0.004062353 | 0.01159884 | 0.001265318 | 0.005610019 |
| $32 \times 32$ | 1089 | 0.004062353 | 0.01160034 | 0.001265319 | 0.005611523 |
| $64 \times 64$ | 4225 | 0.004062353 | 0.01160071 | 0.001265319 | 0.005611899 |
| Series | | 0.004062353 | 0.01160081 | 0.001265319 | 0.005601201 |

[a] $w_{max} = \alpha qL^4/D$ *for uniformly distributed load q.*
[b] $w_{max} = \beta PL^2/D$ *for central concentrated load P.*

Notes: Subdivision of quadrant of plate given for mesh.

**Table 13.4** Corner Supported Square Plate

| | | Point 1 | | Point 2 | |
|---|---|---|---|---|---|
| **Method** | **Mesh** | $w$ | $M_x$ | $w$ | $M_x$ |
| Finite element | $2 \times 2$ | 0.0126 | 0.139 | 0.0176 | 0.095 |
| | $4 \times 4$ | 0.0165 | 0.149 | 0.0232 | 0.108 |
| | $6 \times 6$ | 0.0173 | 0.150 | 0.0244 | 0.109 |
| Marcus [59] | | 0.0180 | 0.154 | 0.0281 | 0.110 |
| Ballesteros and Lee | | 0.0170 | 0.140 | 0.0265 | 0.109 |
| Multiplier | | $qL^4/D$ | $qL^2$ | $qL^4/D$ | $qL^2$ |

Notes: Point 1, center of side; point 2, center of plate.

how the numerical solution of thick plates can easily be achieved and how, in the limit, an alternative procedure for solving all thin plate problems appears. The development of approximations to the cylindrical bending form discussed in Section 13.2.1 plays an important role in the development of viable solutions of the thick, Reissner-Mindlin, theory that also work in "thin" plate applications.

To ensure continuity we repeat below the governing Eqs. (13.19a)–(13.20b). Referring to Fig. 13.5 and the text for definitions, we remark that all the equations could equally well be derived from full three-dimensional analysis of a flat and relatively thin portion of an elastic continuum illustrated in Fig. 13.14. All that it is now necessary to do is to assume that whatever the form of the approximating shape functions in the $xy$ plane, those in the $z$ direction are only linear. Further, it is assumed that the $\sigma_z$ stress is zero, thus eliminating the effect of vertical strain.[4] The first approximations of this type were introduced quite early [61,62] and the elements then derived are exactly of the Reissner-Mindlin theory.

The equations from which we shall start and on which we shall base all subsequent discussion are (a) the moment constitutive equation [see Eq. (13.19a)],

$$\mathbf{M} - \mathbf{D}\mathcal{S}\boldsymbol{\phi} = \mathbf{0} \tag{13.55a}$$

where $\mathbf{D}$ is the matrix of bending rigidities, (b) the shear constitutive equation [see Eqs. (13.19c)],

$$\boldsymbol{\alpha}^{-1}\mathbf{Q} - \boldsymbol{\phi} - \nabla w = \mathbf{0} \tag{13.55b}$$

where $\boldsymbol{\alpha} = \kappa G h \mathbf{I} = \alpha \mathbf{I}$ is the shear rigidity, (c) the moment equilibrium equation [see Eqs. (13.20b)],

$$\mathcal{S}^{\mathrm{T}}\mathbf{M} - \mathbf{Q} = \frac{1}{12}\rho h^3 \ddot{\boldsymbol{\phi}} \tag{13.55c}$$

and (d) the shear equilibrium equation [see Eq. (13.20a)],

$$\nabla^{\mathrm{T}}\mathbf{Q} + q = \rho h \ddot{w} \tag{13.55d}$$

[4] Reissner includes the effect of $\sigma_z$ in bending but, for simplicity, this is disregarded here.

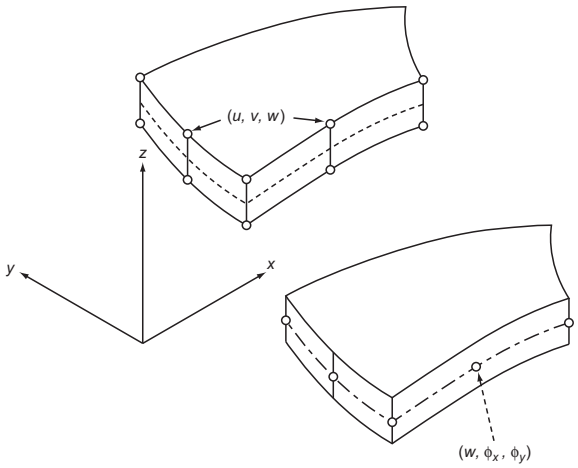



**FIGURE 13.14**
An isoparametric three-dimensional element with linear interpolation in the transverse (thickness) direction and the "thick" plate element.

In the above

$$\boldsymbol{\mathcal{S}} = \begin{bmatrix} \dfrac{\partial}{\partial x} & 0 \\ 0 & \dfrac{\partial}{\partial y} \\ \dfrac{\partial}{\partial y} & \dfrac{\partial}{\partial x} \end{bmatrix} \tag{13.56}$$

defines the strain-displacement operator on rotations $\boldsymbol{\phi}$, and its transpose the equilibrium operator on moments, $\mathbf{M}$. Boundary conditions are of course imposed on $w$ and $\boldsymbol{\phi}$ or the corresponding plate forces $Q_n,\ M_n,\ M_{ns}$ in the manner discussed in Section 13.3.1.

It is convenient to eliminate $\mathbf{M}$ from Eqs. (13.55a)–(13.55d) and write the system of three equations as

$$\begin{aligned} \boldsymbol{\nabla}^{\mathrm{T}}\mathbf{Q} + q &= \rho h \ddot{w} \\ \boldsymbol{\mathcal{S}}^{\mathrm{T}}\mathbf{D}\boldsymbol{\mathcal{S}}\boldsymbol{\phi} - \mathbf{Q} &= \frac{1}{12}\rho h^3 \ddot{\boldsymbol{\phi}} \\ \boldsymbol{\alpha}^{-1}\mathbf{Q} - \boldsymbol{\phi} - \boldsymbol{\nabla} w &= \mathbf{0} \end{aligned} \tag{13.57}$$

This equation system can serve as the basis on which a mixed discretization is built—or alternatively can be reduced further to yield an irreducible form. In Section 13.3.2 we have dealt with the irreducible form which is given by a fourth-order equation in terms of $w$ alone and which could only serve for solution of *thin plate* problems, that is, when $\alpha = \infty$. On the other hand, it is easy to derive an alternative irreducible form which is

valid only if $\alpha \neq \infty$. Thus, the shear forces can be eliminated yielding two equations:

$$\begin{aligned} \nabla^{\mathrm{T}} \left[\alpha \left(\nabla w + \boldsymbol{\phi}\right)\right] + q &= \rho h \ddot{w} \\ \boldsymbol{\mathcal{S}}^{\mathrm{T}} \mathbf{D} \boldsymbol{\mathcal{S}} \boldsymbol{\phi} - \boldsymbol{\alpha} \left(\nabla w + \boldsymbol{\phi}\right) &= \frac{1}{12} \rho h^3 \ddot{\boldsymbol{\phi}} \end{aligned} \tag{13.58}$$

This is an *irreducible* system which for the static problem corresponds to minimization of the total potential energy

$$\begin{aligned} \Pi = \frac{1}{2} \int_{\Omega} \left(\boldsymbol{\mathcal{S}} \boldsymbol{\phi}\right)^{\mathrm{T}} \mathbf{D} \boldsymbol{\mathcal{S}} \boldsymbol{\phi} \, \mathrm{d}\Omega + \frac{1}{2} \int_{\Omega} \left(\nabla w + \boldsymbol{\phi}\right)^{\mathrm{T}} \alpha \left(\nabla w + \boldsymbol{\phi}\right) \mathrm{d}\Omega \\ - \int_{\Omega} w \, q \, \mathrm{d}\Omega - \Pi_{bt} = \text{minimum} \end{aligned} \tag{13.59}$$

as can easily be verified. In the above the first term is simply the bending energy, the second the shear distortion energy, the third term the work of transverse loads, and the last the work of boundary moments and shears given as

$$\Pi_{bt} = \int_{\Gamma_n} \left[\phi_n \bar{M}_n \, \Gamma + \int_{\Gamma_s} \phi_s \bar{M}_{ns}\right] \mathrm{d}\Gamma + \int_{\Gamma_q} w \bar{Q}_n \, \mathrm{d}\Gamma \tag{13.60}$$

Clearly, this irreducible system is only possible when $\alpha \neq \infty$, but it can, obviously, be interpreted as a solution of the potential energy given by Eq. (13.31) for "thin" plates with the constraint of Eq. (13.21) being imposed in a *penalty manner* with $\alpha$ being now a penalty parameter. Thus, as indeed is physically evident, the thin plate formulation is simply a limiting case of such analysis.

We shall see that the penalty form can yield a satisfactory solution only when discretization of the corresponding mixed formulation satisfies the necessary convergence criteria.

The thick plate form now permits independent specification of three conditions at each point of the boundary. The options which exist are

$$\begin{array}{lll} w & \text{or} & Q_n \\ \phi_n & \text{or} & M_n \\ \phi_s & \text{or} & M_{ns} \end{array}$$

in which the subscript $n$ refers to a normal direction to the boundary and $s$ a tangential direction. Clearly, now there are many combinations of possible boundary conditions.

A "fixed" or "clamped" situation exists when all three conditions are given by displacement components, which are generally zero, as

$$w = \phi_n = \phi_s = 0$$

and a free boundary when all conditions are the "resultant" components

$$Q_n = M_n = M_{ns} = 0$$

When we discuss the so-called simply supported conditions (see Section 13.3.1), we shall usually refer to the specification

$$w = 0 \quad \text{and} \quad M_n = M_{ns} = 0$$

as a "soft" support, SS1 (and indeed the most realistic support), and to

$$w = 0, \quad M_n = 0 \quad \text{and} \quad \phi_s = 0$$

as a "hard" support, SS2. The latter in fact replicates the thin plate assumptions and, incidentally, leads to some of the difficulties associated with it.

Finally, there is an important difference between thin and thick plates when "point" loads are involved. In the thin plate case the displacement $w$ remains finite at locations where a point load is applied; however, for thick plates the presence of shearing deformation leads to an infinite displacement (as indeed three-dimensional elasticity theory also predicts). In finite element approximations one always predicts a finite displacement at point locations with the magnitude increasing without limit as a mesh is refined near the loads. Thus, it is meaningless to compare the deflections at point load locations for different element formulations and we will not do so for thick plates. It is, however, possible to compare the total strain energy for such situations and here we immediately observe that for cases in which a single-point load is involved the displacement provides a direct measure for this quantity.

## 13.8 Irreducible formulation: Reduced integration

The procedures for discretizing Eq. (13.58) appear to be straightforward. However, we will find that the process is very sensitive. First, we consider standard isoparametric interpolation in which the two displacement variables are approximated by shape functions and parameters as

$$\boldsymbol{\phi} = \mathbf{N}_\phi \tilde{\boldsymbol{\phi}} \quad \text{and} \quad w = \mathbf{N}_w \tilde{\mathbf{w}} \tag{13.61}$$

We recall that the rotation parameters $\boldsymbol{\phi}$ may be transformed into physical rotations about the coordinate axes, $\boldsymbol{\theta}$, using Eq. (13.16). The parameters $\boldsymbol{\theta}$ are often more convenient for calculations and are essential in shell developments. The approximation equations are now obtained directly by the use of the total potential energy principle [Eq. (13.59)], the Galerkin process on the weak form, or by the use of virtual work expressions. Here we note that the appropriate generalized strain components, corresponding to the moments $\mathbf{M}$ and shear forces $\mathbf{Q}$, are

$$\boldsymbol{\epsilon}_m = \mathcal{S}\boldsymbol{\phi} = (\mathcal{S}\mathbf{N}_\phi)\tilde{\boldsymbol{\phi}} \tag{13.62a}$$

and

$$\boldsymbol{\epsilon}_s = \nabla w + \boldsymbol{\phi} = \nabla \mathbf{N}_w \tilde{\mathbf{w}} + \mathbf{N}_\phi \tilde{\boldsymbol{\phi}} \tag{13.62b}$$

We thus obtain the discretized problem

$$\left(\int_\Omega (\mathcal{S}\mathbf{N}_\phi)^{\mathrm{T}}\mathbf{D}\mathcal{S}\mathbf{N}_\phi \,\mathrm{d}\Omega + \int_\Omega \mathbf{N}_\phi^{\mathrm{T}}\alpha\mathbf{N}_\phi \,\mathrm{d}\Omega\right)\tilde{\boldsymbol{\phi}} + \left(\int_\Omega \mathbf{N}_\phi^{\mathrm{T}}\alpha\nabla\mathbf{N}_w \,\mathrm{d}\Omega\right)\tilde{\mathbf{w}} = \mathbf{f}_\phi$$

and

$$\left(\int_\Omega (\nabla\mathbf{N}_w)^{\mathrm{T}}\alpha\mathbf{N}_\phi \,\mathrm{d}\Omega\right)\tilde{\boldsymbol{\phi}} + \left(\int_\Omega (\nabla\mathbf{N}_w)^{\mathrm{T}}\alpha\nabla\mathbf{N}_w \,\mathrm{d}\Omega\right)\tilde{\mathbf{w}} = \mathbf{f}_w$$

or simply

$$\begin{bmatrix}\mathbf{K}_{ww} & \mathbf{K}_{w\phi}\\ \mathbf{K}_{\phi w} & \mathbf{K}_{\phi\phi}\end{bmatrix}\begin{Bmatrix}\tilde{\mathbf{w}}\\ \tilde{\boldsymbol{\phi}}\end{Bmatrix} = \mathbf{K}\tilde{\mathbf{u}} = \left(\mathbf{K}_b + \mathbf{K}_s\right)\tilde{\mathbf{u}} = \begin{Bmatrix}\mathbf{f}_w\\ \mathbf{f}_\phi\end{Bmatrix} = \mathbf{f} \tag{13.63a}$$

with

$$\begin{aligned}\tilde{\mathbf{u}} &= \begin{Bmatrix}\tilde{\mathbf{w}}\\ \tilde{\boldsymbol{\phi}}\end{Bmatrix} \quad \tilde{\boldsymbol{\phi}} = \begin{Bmatrix}\tilde{\boldsymbol{\phi}}_x\\ \tilde{\boldsymbol{\phi}}_y,\end{Bmatrix}\\ \mathbf{K}_b &= \begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{K}^b_{\phi\phi}\end{bmatrix} \quad \mathbf{K}_s = \begin{bmatrix}\mathbf{K}^s_{ww} & \mathbf{K}^s_{w\phi}\\ \mathbf{K}^s_{\phi w} & \mathbf{K}^s_{\phi\phi}\end{bmatrix}\end{aligned} \tag{13.63b}$$

where the arrays are defined by

$$\begin{aligned}\mathbf{K}^b_{\phi\phi} &= \int_\Omega (\mathcal{S}\mathbf{N}_\phi)^{\mathrm{T}}\mathbf{D}\mathcal{S}\mathbf{N}_\phi \,\mathrm{d}\Omega \quad \mathbf{K}^s_{ww} = \int_\Omega (\nabla\mathbf{N}_w)^{\mathrm{T}}\alpha\nabla\mathbf{N}_w \,\mathrm{d}\Omega\\ \mathbf{K}^s_{\phi\phi} &= \int_\Omega \mathbf{N}_\phi^{\mathrm{T}}\alpha\mathbf{N}_\phi \,\mathrm{d}\Omega \qquad\qquad \mathbf{K}^s_{\phi w} = \int_\Omega \mathbf{N}_\phi^{\mathrm{T}}\alpha\nabla\mathbf{N}_w \,\mathrm{d}\Omega = (\mathbf{K}^s_{w\phi})^{\mathrm{T}}\end{aligned} \tag{13.63c}$$

and forces are given by

$$\begin{aligned}\mathbf{f}_w &= \int_\Omega \mathbf{N}_w^{\mathrm{T}}\, q \,\mathrm{d}\Omega + \int_{\Gamma_s} \mathbf{N}_w^T\, \bar{Q}_n \,\mathrm{d}\Gamma\\ \mathbf{f}_\phi &= \int_{\Gamma_m} \mathbf{N}_\phi^T \bar{\mathbf{M}} \,\mathrm{d}\Gamma\end{aligned} \tag{13.63d}$$

where $\bar{Q}_n$ is the prescribed shear on boundary $\Gamma_s$, and $\bar{\mathbf{M}}$ $(\bar{M}_n,\ \bar{M}_{ns})$ is the prescribed moment on boundary $\Gamma_m$ (where for simplicity we combine $\Gamma_n$ and $\Gamma_s$).

The formulation is straightforward and there is little to be said about it *a priori*. Since the form contains only first derivatives apparently any $C_0$ shape functions of a two-dimensional kind can be used to interpolate the two rotations and the lateral displacement. Figure 13.15 shows some rectangular (or with isoparametric distortion, quadrilateral) elements used in the early work [61–63]. All should, in principle, be convergent as $C_0$ continuity exists and constant strain states are available. In Fig. 13.16 we show what in fact happens with a fairly fine subdivision of quadratic serendipity and Lagrangian rectangles as the ratio of span to thickness, $L/h$, varies. Here $L$ is a characteristic length of the plate and may be a side length, a loading length or a normal mode characteristic. We note that the magnitude of the coefficient $\alpha$ is best measured by the ratio of the bending to shear rigidities and we could assess its value in a nondimensional form. Thus, for an isotropic material with $\alpha = \kappa G h$ this ratio

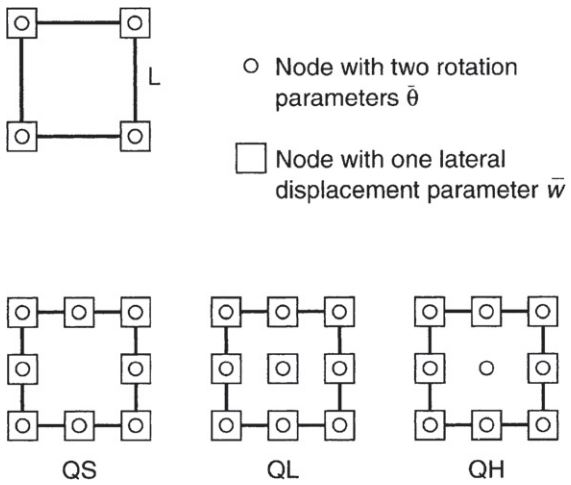



**FIGURE 13.15**
Some early thick plate elements.

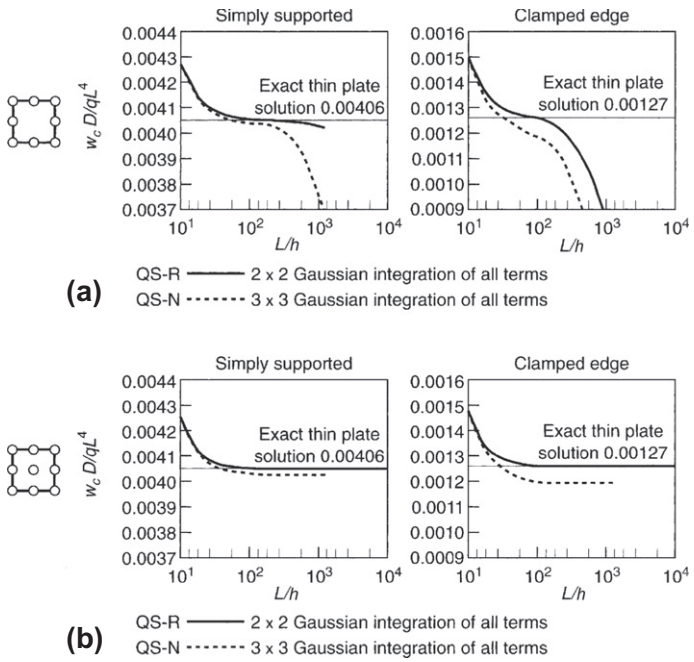



**FIGURE 13.16**
Performance of (a) quadratic serendipity (QS) and (b) Lagrangian (QL) elements with varying span-to-thickness $L/h$ ratios, uniform load on a square plate with $4 \times 4$ normal subdivisions in a quarter. R is reduced $2 \times 2$ quadrature and N is normal $3 \times 3$ quadrature.

becomes

$$\frac{12(1-\nu^2)\kappa G h L^2}{E h^3} \propto \left(\frac{L}{h}\right)^2 \tag{13.64}$$

Obviously, "thick" and "thin" behavior therefore depends on the $L/h$ ratio.

It is immediately evident from Fig. 13.16 that, while the answers are quite good for smaller $L/h$ ratios, the serendipity quadratic fully integrated elements (QS) rapidly depart from the thin plate solution, and in fact tend to zero results (locking) when this ratio becomes large. For Lagrangian quadratics (QL) the answers are better, but again as the plate tends to be thin they are on the small side.

The reason for this "locking" performance is similar to those considered for the nearly incompressible problem in Chapter 10 and the cylindrical bending of thin plate problem (viz. Section 13.2.1). In the case of plates the shear constraint implied by the third of Eq. (13.57), and used to eliminate the shear resultant, is too strong if the terms in which this is involved are fully integrated. Indeed, we see that the effect is more pronounced in the serendipity element than in the Lagrangian one. In early work the problem was thus mitigated by using a *reduced* quadrature, either on all terms, which we label $R$ in the figure [64,65], or only on the offending shear terms selectively [66,67] (labeled $S$). The dramatic improvement in results is immediately noted.

The same improvement in results is observed for linear quadrilaterals in which the full (exact) integration gives results that are totally unacceptable (as shown in Fig. 13.17), but where a reduced integration on the shear terms (single point) gives excellent performance [11], although a careful assessment of the element stiffness shows it to be rank deficient in an "hourglass" mode in transverse displacements. (Reduced integration on all terms gives additional matrix singularity.) The use of reduced quadrature to develop a four-node plate element has been revisited by Gruttmann and Wagner using a stabilized form [68].

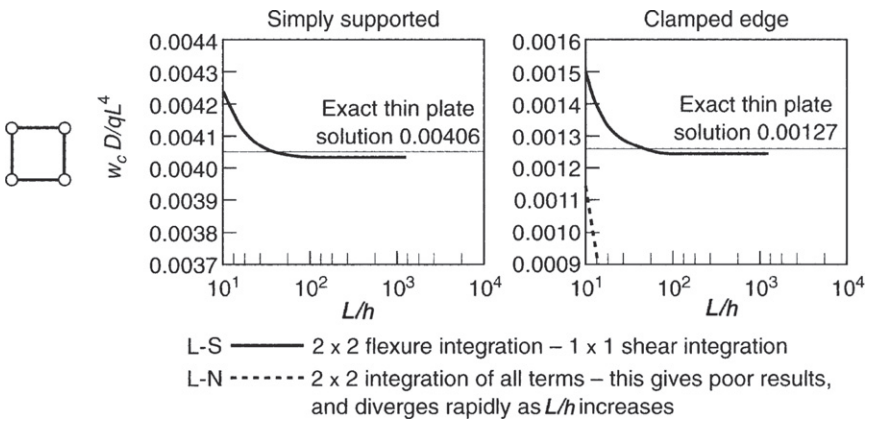



**FIGURE 13.17**

Performance of bilinear elements with varying span-to-thickness, $L/h$, values.

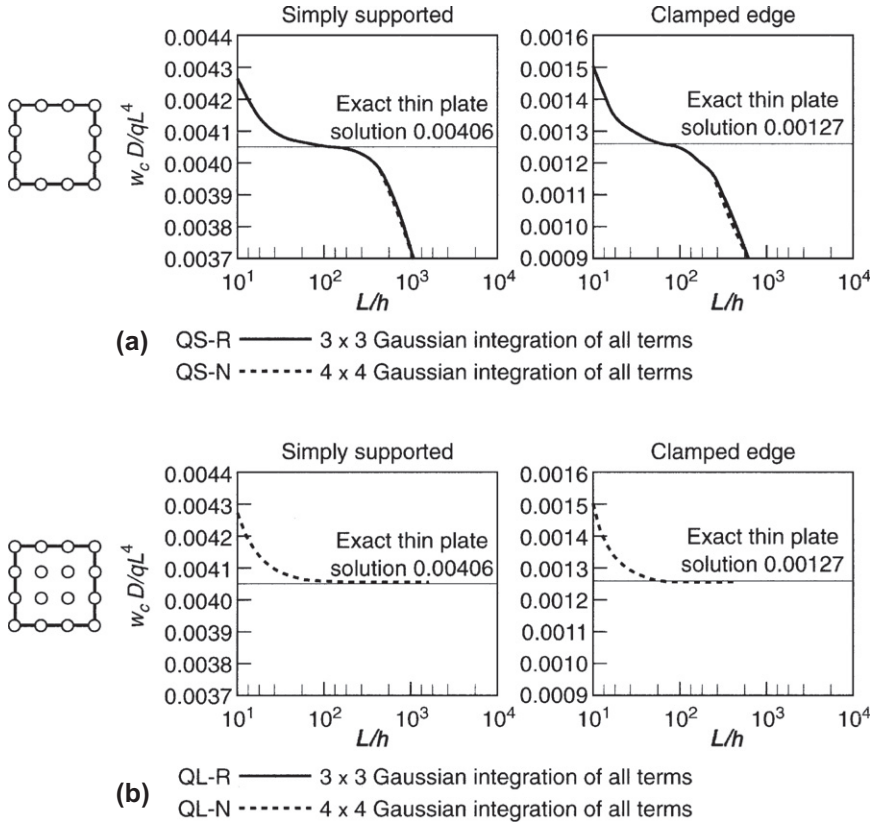



**FIGURE 13.18**
Performance of cubic quadrilaterals: (a) serendipity (QS) and (b) Lagrangian (QL) with varying span-to-thickness, $L/h$, values.

A remedy thus has been found; however, it is not universal. We note in Fig. 13.16 that even without reduction of integration order, Lagrangian elements perform better in the quadratic expansion since in a sub-parametric mapping they retain all global coordinate quadratic terms. In cubic elements (Fig. 13.18), however, we note that (a) almost no change occurs when integration is "reduced" and (b) again, Lagrangian-type elements perform very much better.

In the late 1970s many heuristic arguments were advanced for devising better elements [69–72], all making use of reduced integration concepts. Some of these perform quite well, for example the so-called "heterosis" element of Hughes and Cohen [69] (in which the serendipity-type interpolation is used on $w$ and a Lagrangian one on $\boldsymbol{\phi}$), but all of the elements developed in that era fail on some occasions, either locking or exhibiting singular behavior. Thus such elements are not "robust" and should not be used universally.

A better explanation of their failure is needed and hence an understanding of how such elements can be designed. In the next section we shall address this problem by considering a mixed formulation. The reader will recognize here arguments used in the nearly incompressible problem in Chapter 10 which led to a better understanding of the failure of some straightforward elasticity elements as incompressible behavior was approached. The situation is completely parallel here.

## 13.9 Mixed formulation for thick plates

### 13.9.1 The approximation

The problem of thick plates can, of course, be solved as a mixed one starting from Eq. (13.57) and approximating directly each of the variables $w$, $\boldsymbol{\phi}$, and $\mathbf{Q}$ independently. In the sequel we consider only the development of appropriate stiffness behavior since this is the source of locking and singularities. The inertia effects may be added later. Using Eq. (13.57), we construct a static weak form as

$$\begin{aligned} \int_\Omega \delta w \left[\nabla^{\mathrm{T}}\mathbf{Q} + q\right] \mathrm{d}\Omega = 0 \\ \int_\Omega \left[\left(\mathcal{S}\delta\boldsymbol{\phi}\right) \mathcal{D}\mathcal{S}\boldsymbol{\phi} + \delta\boldsymbol{\phi}^T\mathbf{Q}\right] \mathrm{d}\Omega = 0 \\ \int_\Omega \delta\mathbf{Q}^{\mathrm{T}} \left[-\frac{1}{\alpha}\mathbf{Q} + \boldsymbol{\phi} + \nabla w\right] \mathrm{d}\Omega = 0 \end{aligned} \tag{13.65}$$

We now write the independent approximations, using the standard Galerkin procedure, as

$$\begin{aligned} \mathbf{w} = \mathbf{N}_w\tilde{\mathbf{w}}, \quad \boldsymbol{\phi} = \mathbf{N}_\phi\tilde{\boldsymbol{\phi}} \quad \text{and} \quad \mathbf{Q} = \mathbf{N}_q\tilde{\mathbf{Q}} \\ \delta\mathbf{w} = \mathbf{N}_w\delta\tilde{\mathbf{w}}, \quad \delta\boldsymbol{\phi} = \mathbf{N}_\phi\delta\tilde{\boldsymbol{\phi}} \quad \text{and} \quad \delta\mathbf{Q} = \mathbf{N}_q\delta\tilde{\mathbf{Q}} \end{aligned} \tag{13.66}$$

though, of course, other interpolation forms can be used, as we shall note later.

After appropriate integrations by parts of the first two of (13.65), we obtain the discrete symmetric equation system

$$\begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{E}^T \\ \mathbf{0} & \mathbf{K}_b & \mathbf{C}^T \\ \mathbf{E} & \mathbf{C} & \mathbf{H} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{w}} \\ \tilde{\boldsymbol{\phi}} \\ \tilde{\mathbf{Q}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_w \\ \mathbf{f}_\phi \\ \mathbf{0} \end{Bmatrix} \tag{13.67}$$

where

$$\begin{aligned} \mathbf{K}_b &= \int_\Omega \left(\mathcal{S}\mathbf{N}_\phi\right)^T \mathbf{D} \left(\mathcal{S}\mathbf{N}_\phi\right) \mathrm{d}\Omega \\ \mathbf{E} &= \int_\Omega \mathbf{N}_q^T \nabla\mathbf{N}_w \mathrm{d}\Omega \\ \mathbf{C} &= \int_\Omega \mathbf{N}_q^T \mathbf{N}_\phi \mathrm{d}\Omega \\ \mathbf{H} &= -\int_\Omega \mathbf{N}_q^T \frac{1}{\alpha}\mathbf{N}_q \mathrm{d}\Omega \end{aligned} \tag{13.68}$$

and where $\mathbf{f}_w$ and $\mathbf{f}_\phi$ are as defined in Eq. (13.63d).

The above represents a typical three-field mixed problem of the type discussed in Section 9.5, which has to satisfy certain criteria for stability of approximation as the thin plate limit (which can now be solved exactly) is approached. For this limit we have

$$\alpha = \infty \quad \text{and} \quad \mathbf{H} = \mathbf{0} \tag{13.69}$$

In this limiting case it can readily be shown that *necessary* criteria of solution stability for any element assembly and boundary conditions are that

$$n_\phi + n_w \geq n_Q \quad \text{or} \quad \alpha_{\mathrm{P}} \equiv \frac{n_\phi + n_w}{n_Q} \geq 1 \tag{13.70a}$$

and

$$n_Q \geq n_w \quad \text{or} \quad \beta_{\mathrm{P}} \equiv \frac{n_Q}{n_w} \geq 1 \tag{13.70b}$$

where $n_\phi$, $n_Q$, and $n_w$ are the number of $\tilde{\boldsymbol{\phi}}$, $\tilde{\mathbf{Q}}$, and $\tilde{\mathbf{w}}$ parameters in Eqs. (13.67).

When the count condition is not satisfied then the *equation system either will be singular or will lock*. Equations (13.70a) and (13.70b) must be satisfied for the whole system but, in addition, they need to be satisfied for element patches if local instabilities and oscillations are to be avoided [73–75]. We remind the reader that Eqs. (13.70a) are (13.70b) are *necessary conditions*; however, they are not *sufficient conditions*. It is always necessary to conduct consistency and stability tests to ensure the proposed element passes the complete mixed patch test.

The above criteria will, as we shall see later, help us to design suitable thick plate elements which show convergence to correct thin plate solutions.

### 13.9.2 Continuity requirements

The approximation of the form given in Eqs. (13.67) and (13.68) implies certain continuities. It is immediately evident that $C_0$ continuity is needed for rotation shape functions $\mathbf{N}_\phi$ (as products of first derivatives are present in the approximation), but that either $\mathbf{N}_w$ or $\mathbf{N}_q$ can be discontinuous. In the form given in Eq. (13.68) a $C_0$ approximation for $w$ is implied; however, after integration by parts a form for $C_0$ approximation of $\mathbf{Q}$ results. Of course, physically only the component of $\mathbf{Q}$ normal to boundaries should be continuous.

In all the early approximations discussed in the previous section, $C_0$ continuity was assumed for both $\boldsymbol{\phi}$ and $w$ variables, this being very easy to impose. We note that such continuity cannot be described as *excessive* (as no physical conditions are violated), but we shall show later that successful elements also can be generated with discontinuous $w$ interpolation (which is indeed not motivated by physical considerations).

For $\mathbf{Q}$ it is obviously more convenient to use a completely discontinuous interpolation as then the shear can be eliminated at the element level and the final stiffness matrices written simply in standard $\tilde{\boldsymbol{\phi}}$, $\tilde{\mathbf{w}}$ terms for element boundary nodes. We shall show later that some formulations permit a limited case where $\alpha^{-1}$ is identically zero while others require it to be nonzero.

The continuous interpolation of the normal component of $\mathbf{Q}$ is, as stated above, physically correct in the absence of line or point loads. However, with such interpolation, elimination of $\tilde{\mathbf{Q}}$ is usually not possible and the retention of such additional system variables is usually too costly to be used in practice and has so far not been adopted. However, we should note that an iterative solution process applicable to mixed forms can reduce substantially the cost of such additional variables [76].

### 13.9.3 Equivalence of mixed forms with discontinuous Q interpolation and reduced (selective) integration

An equivalence of penalized mixed forms with discontinuous interpolation of the constraint variable and of the corresponding irreducible forms with the same penalty variable may be demonstrated following work of Malkus and Hughes for incompressible problems [77]. Indeed, an exactly analogous proof can be used for the present case. Thus, for instance, we consider a serendipity quadrilateral, shown in Fig. 13.19, in which integration of shear terms (involving $\alpha$) is made at four Gauss points (i.e., $2 \times 2$ reduced quadrature) in an irreducible formulation [see Eqs. (13.63a)–(13.63d)]. We find that the answers are *identical* to a mixed form in which the $\mathbf{Q}$ variables are given by a bilinear interpolation from nodes placed at the same Gauss points.

This result can also be argued from the limitation principle. This states that if the mixed form in which the stress is independently interpolated is precisely capable of reproducing the stress variation which is given in a corresponding irreducible form then the analysis results will be identical. It is clear that the four Gauss points at which the shear stress is sampled can only define a bilinear variation and thus the identity applies here.

The equivalence of reduced integration with the mixed discontinuous interpolation of $\mathbf{Q}$ will be useful in our discussion to point out reasons why many elements mentioned in the previous section failed. However, in practice, it will be found equally convenient (and often more effective) to use the mixed interpolation explicitly and eliminate the $\mathbf{Q}$ variables by element-level condensation rather than to use special integration rules. Moreover, in problems where the material properties lead to coupling between bending and shear response use of selective reduced integration is not convenient. It must also be pointed out that the equivalence fails if $\alpha$ varies within an element or indeed if the isoparametric mapping implies different interpolations.

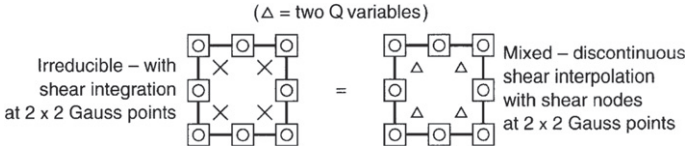



**FIGURE 13.19**
Equivalence of mixed form and reduced shear integration in quadratic serendipity rectangle.

### 13.9.4 Why elements fail: Patch test for thick plates

The nature and application of the patch test have changed considerably since its early introduction. As shown in Chapters 9 and 10 and in Refs. [73–75,78–82] this test can prove, in addition to *consistency* requirements (which were initially the only item tested), the *stability* of the approximation by requiring that for a patch consisting of an assembly of one or more elements the stiffness matrices are nonsingular whatever the boundary conditions imposed.

To be absolutely sure of such nonsingularity the test must, at the final stage, be performed numerically. However, we find that the "count" conditions given in Eqs. (13.70a) and (13.70b) are *necessary* for avoiding such nonsingularity. Frequently, they also prove *sufficient* and make the numerical test only a final confirmation [74,75]. We shall demonstrate how the simple application of such counts immediately indicates *which elements fail* and which have a chance of success. Indeed, it is easy to show why the original quadratic serendipity element with reduced integration (QS-R) is not robust.

In Fig. 13.20 we consider this element in a single-element and four-element patch subjected to so-called *constrained* boundary conditions, in which all displacements on the external boundary of the patch are prescribed and a *relaxed* boundary condition in which only three displacements (conveniently two $\phi$'s and one $w$) eliminate the rigid body modes. To ease the presentation of this figure, as well as in subsequent tests, we shall simply quote the values of $\alpha_P$ and $\beta_P$ parameters as defined in Eqs. (13.70a)

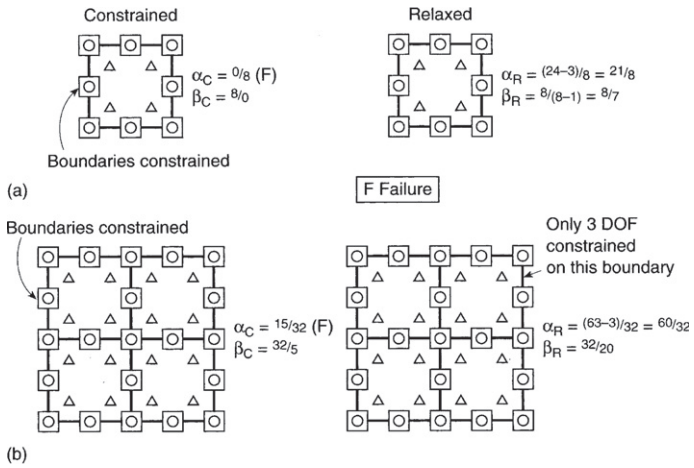



**FIGURE 13.20**

"Constrained" and "relaxed" patch test/count for serendipity (quadrilateral). (In the *C* test all boundary displacements are fixed. In the *R* test only three boundary displacements are fixed, eliminating rigid body modes.) (a) Single-element test and (b) four-element test.

and (13.70b) with subscript replaced by C or R to denote the constrained or relaxed tests, respectively. The symbol F will be given to any failure to satisfy the *necessary* condition. In the tests of Fig. 13.20 we note that both patch tests fail with the parameter $\alpha_C$ being less than 1, and hence the elements will lock under certain circumstances (or show a singularity in the evaluation of $\mathbf{Q}$). A failure in the relaxed tests generally predicts a singularity in the final stiffness matrix of the assembly, and this is also where frequently computational failures have been observed.

As the mixed and reduced integration elements are identical in this case we see immediately why the element fails in the problem of Fig. 13.16 (more severely under clamped conditions). Indeed, it is clear why in general the performance of Lagrangian-type elements is better as it adds further degrees of freedom to increase $n_\phi$ (and $n_w$ unless heterosis-type interpolation is used) [69].

In Table 13.5 we show a list of the $\alpha_P$ and $\beta_P$ values for single- and four-element patches of various rectangles, and again we note that none of these satisfies completely the *necessary* requirements, and therefore none can be considered robust. However, it is interesting to note that the elements closest to satisfaction of the count perform best, and this explains why the heterosis elements [83] are quite successful and indeed why the Lagrangian cubic is nearly robust and often is used with success [84].

Of course, similar approximation and counts can be made for various triangular elements. We list some typical and obvious ones, together with patch test counts, in the first part of Table 13.6. Again, none perform adequately and all will result in locking and spurious modes in finite element applications.

We should note again that the failure of the patch test (with regard to stability) means that under some circumstances the element will fail. However, in many problems a reasonable performance can still be obtained and nonsingularity observed in its performance, providing consistency is, of course, also satisfied.

#### 13.9.4.1 *Numerical patch test*

While the "count" condition of Eqs. (13.70a) and (13.70b) is necessary for stability of patches, on occasion singularity (and hence instability) can still arise even with its satisfaction. This is caused by a linear dependence in the equations of the stiffness matrix. For this reason numerical tests should always be conducted ascertaining the rank sufficiency of the stiffness matrices and testing the consistency condition.

In Chapter 8 we discussed in detail the consistency test for irreducible forms in which a single variable set $\mathbf{u}$ occurred. It was found that with a second-order operator the discrete equations should satisfy *at least* the solution corresponding to a linear field $\mathbf{u}$ exactly, thus giving constant strains (or first derivatives) and stresses. For the mixed equation set [Eq. (13.57)] again the lowest-order exact solution that has to be satisfied corresponds to

1. Constant values of moments or $\mathcal{S}\boldsymbol{\phi}$ and hence a linear $\boldsymbol{\phi}$ field
2. Linear $w$ field
3. Constant $\mathbf{Q}$ field

**Table 13.5** Quadrilateral Mixed Elements: Patch Count

| | | Single-Element Patch | | | | Four-Element Patch | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Element** | **Reference** | $\alpha_C$ | $\beta_C$ | $\alpha_R$ | $\beta_R$ | $\alpha_C$ | $\beta_C$ | $\alpha_R$ | $\beta_R$ |
| I | | | | | | | | | |
|  | Q4S1 [11,64,67] | $\frac{0}{2}$ (F) | $\frac{2}{0}$ | $\frac{9}{2}$ | $\frac{2}{3}$ (F) | $\frac{3}{8}$ (F) | $\frac{8}{1}$ | $\frac{24}{8}$ | $\frac{8}{8}$ |
|  | Q8S4 [64] | $\frac{0}{8}$ (F) | $\frac{8}{0}$ | $\frac{21}{8}$ | $\frac{8}{7}$ | $\frac{15}{32}$ (F) | $\frac{32}{5}$ | $\frac{60}{32}$ | $\frac{32}{20}$ |
|  | Q9S4 [66,67] | $\frac{3}{8}$ (F) | $\frac{8}{1}$ | $\frac{16}{8}$ | $\frac{8}{8}$ | $\frac{27}{32}$ (F) | $\frac{32}{9}$ | $\frac{72}{32}$ | $\frac{32}{20}$ |
|  | Q9HS4 [69] | $\frac{2}{8}$ (F) | $\frac{8}{0}$ | $\frac{15}{8}$ | $\frac{8}{7}$ | $\frac{23}{32}$ (F) | $\frac{32}{5}$ | $\frac{68}{32}$ | $\frac{32}{20}$ |
|  | Q12S9 [64] | $\frac{0}{18}$ (F) | $\frac{18}{0}$ | $\frac{23}{18}$ | $\frac{18}{11}$ | $\frac{27}{72}$ (F) | $\frac{72}{9}$ | $\frac{96}{72}$ | $\frac{72}{32}$ |
|  | Q16S9 [84] | $\frac{12}{18}$ (F) | $\frac{18}{4}$ | $\frac{45}{18}$ | $\frac{18}{15}$ | $\frac{75}{72}$ | $\frac{72}{25}$ | $\frac{150}{72}$ | $\frac{72}{50}$ |
| II | | | | | | | | | |
|  | Q4S1B1 [85]<br>Q4S1B1L [86,87] | $\frac{2}{2}$ | $\frac{2}{0}$ | $\frac{11}{2}$ | $\frac{2}{3}$ (F) | $\frac{11}{8}$ | $\frac{8}{1}$ | $\frac{32}{8}$ | $\frac{8}{8}$ |
|  | Q4S2B2L [88] | $\frac{4}{4}$ | $\frac{4}{0}$ | $\frac{13}{4}$ | $\frac{4}{4}$ | $\frac{19}{16}$ | $\frac{16}{1}$ | $\frac{40}{16}$ | $\frac{16}{8}$ |

[F] Failure to satisfy necessary conditions.

The exact solutions for which plate elements commonly are tested and where full satisfaction of nodal equations is required consist of

1. Arbitrary constant $\mathbf{M}$ fields and arbitrary linear $\boldsymbol{\phi}$ fields with zero shear forces ($\mathbf{Q} = \mathbf{0}$); here a quadratic $w$ form is assumed still yielding an exact finite element solution.

**Table 13.6** Triangular Mixed Elements: Patch Count

| | | Single-Element Patch | | | | Six-Element Patch | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Element** | **Reference** | $\alpha_C$ | $\beta_C$ | $\alpha_R$ | $\beta_R$ | $\alpha_C$ | $\beta_C$ | $\alpha_R$ | $\beta_R$ |
| I | | | | | | | | | |
|  | T3S1 | $\frac{0}{2}$ (F) | $\frac{2}{0}$ | $\frac{6}{2}$ | $\frac{2}{2}$ | $\frac{3}{12}$ (F) | $\frac{12}{1}$ | $\frac{18}{12}$ | $\frac{12}{6}$ |
|  | T6S3 | $\frac{0}{6}$ (F) | $\frac{6}{0}$ | $\frac{15}{6}$ | $\frac{6}{5}$ | $\frac{21}{36}$ (F) | $\frac{36}{7}$ | $\frac{54}{36}$ | $\frac{36}{18}$ |
|  | T10S3 | $\frac{3}{6}$ (F) | $\frac{6}{6}$ | $\frac{27}{6}$ | $\frac{6}{9}$ (F) | $\frac{57}{36}$ | $\frac{36}{19}$ | $\frac{108}{36}$ | $\frac{36}{36}$ |
| II | | | | | | | | | |
|  | T3S1B1L [89,90] | $\frac{2}{2}$ | $\frac{2}{0}$ | $\frac{8}{2}$ | $\frac{2}{2}$ | $\frac{20}{12}$ | $\frac{12}{6}$ | $\frac{35}{12}$ | $\frac{12}{11}$ |
|  | T3S1B1A [91] | $\frac{2}{2}$ | $\frac{2}{0}$ | $\frac{8}{2}$ | $\frac{2}{2}$ | $\frac{15}{12}$ | $\frac{12}{1}$ | $\frac{30}{12}$ | $\frac{12}{6}$ |
|  | T6S1B1 | $\frac{2}{2}$ | $\frac{2}{0}$ | $\frac{17}{2}$ | $\frac{2}{5}$ (F) | $\frac{33}{12}$ | $\frac{12}{7}$ | $\frac{66}{12}$ | $\frac{12}{18}$ (F) |
|  | T6S3B3 [75] | $\frac{6}{6}$ | $\frac{6}{0}$ | $\frac{21}{6}$ | $\frac{6}{5}$ | $\frac{75}{36}$ | $\frac{36}{7}$ | $\frac{108}{36}$ | $\frac{36}{18}$ |

[F] Failure to satisfy necessary conditions.

2. Constant $\mathbf{Q}$ and linear $w$ fields yielding a constant $\boldsymbol{\phi}$ field. The solution requires a distributed couple on the right-hand side of the second of Eq. (13.57) and this was not included in the original formulation. A simple procedure is to disregard the satisfaction of the moment equilibrium in this test. This may be done simply by inserting a very large value of the bending rigidity $\mathbf{D}$.

### 13.9.5 Design of some useful elements

The simple patch count test indicates how elements could be designed to pass it, and thus avoid the singularity (instability). Equation (13.70b) is always trivial to satisfy for elements in which $\mathbf{Q}$ is interpolated independently in each element. In a single-element test it will be necessary to restrain at least one $\tilde{\mathbf{w}}$ degree of freedom to prevent rigid body translations. Thus, the *minimum* number of terms which can be included in $\mathbf{Q}$ for each element is always one less than the number of $\tilde{\mathbf{w}}$ parameters in each

element. As patches with more than one element are constructed the number of $w$ parameters will increase proportionally with the number of nodes and the number of shear constraints increase by the number of elements. For both quadrilateral and triangular elements the requirement that $n_Q \geq n_w - 1$ *for no boundary restraints* ensures that Eq. (13.70b) is satisfied on all patches for both constrained and relaxed boundary conditions. Failure to satisfy this simple requirement explains clearly why certain of the elements in Tables 13.5 and 13.6 failed the single-element patch test for the relaxed boundary condition case.

Thus, a successful satisfaction of the count condition requires now only the consideration of Eq. (13.70a). In the remainder of this chapter we will discuss an approach which can successfully satisfy Eq. (13.70a). The basic approach is to introduce bubble or enhanced modes for the rotation parameters in the interior of elements. Here, for convenience, we refer to both as a "bubble mode" approach. The inclusion of at least as many bubble modes as shear modes will automatically satisfy Eq. (13.70a). This latter approach is similar to that used in Chapter 10 to stabilize elements for solving the (nearly) incompressible problem and is a clear violation of "intuition" since for the thin plate problem the rotations appear as derivatives of $w$. Its use in this case is justified by patch counts and performance. Subsequently, we introduce linking to improve the accuracy of the elements.

## 13.10 Elements with rotational bubble or enhanced modes

As a starting point for this class of elements we may consider a standard functional of Reissner type given by

$$\begin{aligned}\Pi = \frac{1}{2}\int_\Omega \left(\mathcal{S}\boldsymbol{\phi}\right)^{\mathrm{T}} \mathbf{D}\mathcal{S}\boldsymbol{\phi}\,\mathrm{d}\Omega + \int_\Omega \mathbf{Q}^{\mathrm{T}}\left(\nabla w + \boldsymbol{\phi}\right)\,\mathrm{d}\Omega \\ -\frac{1}{2}\int_\Omega \mathbf{Q}^{\mathrm{T}}\alpha^{-1}\mathbf{Q}\,\mathrm{d}\Omega - \int_\Omega w\,q\,\mathrm{d}\Omega - \Pi_{bt} = \text{stationary}\end{aligned} \tag{13.71}$$

in which approximations for $w$, $\boldsymbol{\phi}$, and $\mathbf{Q}$ are required.

Two triangular elements designed by introducing "bubble modes" for rotation parameters are found to be robust, and at the same time excellent performers. None of these elements is "obvious," and they all use an interpolation of rotations that is of higher or equal order than that of $w$. Figure 13.21 shows the degree of freedom assignments for these triangular elements and the second part of Table 13.6 shows again their performance in patches.

The quadratic element (T6S3B3) was devised by Zienkiewicz and Lefebvre [75] starting from a quadratic interpolation for $w$ and $\boldsymbol{\phi}$. The shear $\mathbf{Q}$ is interpolated by a complete linear polynomial in each element, giving here six parameters, $\tilde{\mathbf{Q}}$. Three hierarchical quartic bubbles are added to the rotations giving the interpolation

$$\boldsymbol{\phi} = \sum_{a=1}^{6} N_a(L_k)\tilde{\boldsymbol{\phi}}_a + \sum_{b=1}^{3} \Delta N_b(L_k)\Delta\tilde{\boldsymbol{\phi}}_b \tag{13.72}$$

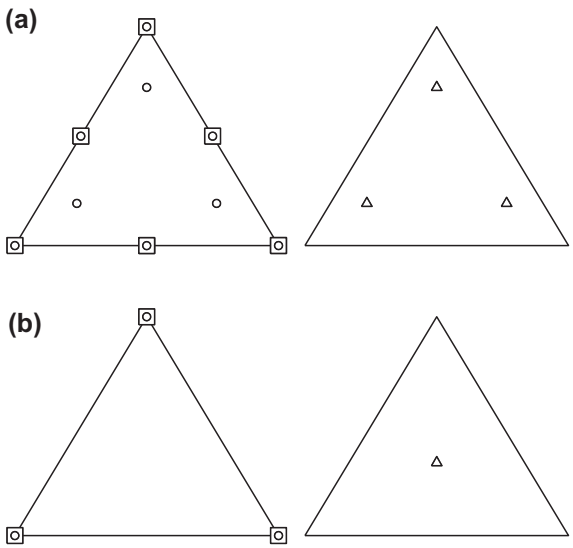



**FIGURE 13.21**
Robust triangular elements: (a) the T6S3B3 element of Zienkiewicz and Lefebvre [75] and (b) the T3S1B1 element of Xu [89].

where $N_a(L_k)$ are conventional quadratic isoparametric interpolations on the six-node triangle and shape functions for the quartic bubble modes are given as

$$\Delta N_b(L_k) = L_b(L_1 L_2 L_3)$$

Thus, we have introduced six additional rotation parameters but have left the number of $w$ parameters unaltered from those given by a quadratic interpolation. This element has very desirable properties and good performance when the integral for $\mathbf{K}_b$ in Eq. (13.67) is computed by using seven-point quadrature and the other integrals are computed by using four points. We shall show later that optimal performance can be attained using "linked" interpolation on this element.

Since the T6S3B3 type elements use a complete quadratic to describe the displacement and rotation field, an isoparametric mapping may be used to produce curved-sided elements. Furthermore, by design this type passes the count test and by numerical testing is proved to be quite robust when used to analyze both thick and thin plate problems [75,92]. Since the $w$ displacement interpolation is a standard quadratic interpolation the element may be joined compatibly to tetrahedral or prism solid elements which have six-node faces. We shall show later that optimal performance can be attained for this element by using "linked" interpolation for $w$ with additional enhanced strain modes—however, using this form the compatibility between any attached tetrahedral elements is lost.

A linear triangular element [T3S1B1—Fig. 13.21(b)] with a total of nine nodal degrees of freedom adds a single cubic bubble to the linear rotational interpolation

and uses linear interpolation for $w$ with constant discontinuous shear. This element satisfies all count conditions for solution (see Table 13.6); however, without further enhancements it locks as the thin plate limit is approached [89]. As we have stated previously the count condition is necessary but not sufficient to define successful elements and numerical testing is always needed. In a later section we discuss a "linked interpolation" modification which also makes this element robust.

The introduction of successful bubble modes in quadrilaterals is more difficult. The first condition examined was the linear quadrilateral with a single bubble mode (Q4S1B1). For this element the patch count test fails when only a single element is considered but for assemblies above four elements it is passed and much hope was placed on this condition [86,87]. Unfortunately, one singular mode with a zero eigenvalue persists in all assemblies when the completely relaxed support conditions are considered. Despite this singularity the element does not lock and usually gives an excellent performance [87].

To avoid, however, any singularity it is necessary to have at least three shear stress components and a similar number of rotation components of bubble form. No simple way of achieving a three-term interpolation exists but a successful four-component form was obtained by Auricchio and Taylor [88]. This four-term interpolation for shear is given by

$$\mathbf{Q} = \begin{bmatrix} J^0_{11} & J^0_{21} & J^0_{11}\eta & J^0_{21}\xi \\ J^0_{12} & J^0_{22} & J^0_{12}\eta & J^0_{22}\xi \end{bmatrix} \begin{Bmatrix} \widetilde{Q}_1 \\ \widetilde{Q}_2 \\ \widetilde{Q}_3 \\ \widetilde{Q}_4 \end{Bmatrix} \tag{13.73}$$

The Jacobian transformation $J^0_{ij}$ is identical to that introduced when describing the Pian-Sumihara element in Chapter 9 and is computed as

$$\begin{aligned} J^0_{11} &= x_{,\xi}\big|_{\xi=\eta=0}, \quad J^0_{12} = x_{,\eta}\big|_{\xi=\eta=0} \\ J^0_{21} &= y_{,\xi}\big|_{\xi=\eta=0}, \quad J^0_{22} = y_{,\eta}\big|_{\xi=\eta=0} \end{aligned} \tag{13.74}$$

To satisfy Eq. (13.70a) it is necessary to construct a set of four bubble modes. An appropriate form is found to be

$$\Delta\mathbf{N}_h = \frac{1}{j} N_h(\xi,\ \eta) \begin{bmatrix} J^0_{22} & -J^0_{12} & J^0_{22}\eta & -J^0_{12}\xi \\ -J^0_{21} & J^0_{11} & -J^0_{21}\eta & J^0_{11}\xi \end{bmatrix} \tag{13.75}$$

in which $j$ is the determinant of the Jacobian transformation $\mathbf{J}$ (i.e., not the determinant of $\mathbf{J}^0$) and $N_h = (1-\xi^2)(1-\eta^2)$ is a hierarchical bubble mode. Thus, the rotation parameters are interpolated by using

$$\boldsymbol{\phi} = \sum_{a=1}^{4} N_a \tilde{\boldsymbol{\phi}}_a + \Delta\mathbf{N}_h\ \Delta\tilde{\boldsymbol{\phi}}_h \tag{13.76}$$

where $N_a$ are the standard bilinear interpolations for the four-node quadrilateral.

The element so achieved (Q4S2B2) is stable and quite accurate. However, improvement in accuracy can be achieved for all elements by introducing "linked" interpolation.

## 13.11 Linked interpolation: An improvement of accuracy

In the previous section we outlined various procedures which are effective in ensuring the necessary count conditions and which are, therefore, essential to make the elements "work" without locking or singularity. In this section we shall try to improve the interpolation used to increase the accuracy without involving additional parameters.

The reader will here observe that in the primary interpolation we have used equal-order polynomials to interpolate both the displacement ($w$) and the rotations ($\boldsymbol{\phi}$). Clearly, if we consider the limit of thin plate theory

$$\boldsymbol{\phi} = \begin{Bmatrix} \phi_x \\ \phi_y \end{Bmatrix} = -\nabla w \tag{13.77}$$

and hence one order lower interpolation for $\boldsymbol{\phi}$ appears necessary. To achieve this we introduce here the concept of *linked interpolation* in which the primary expression is written as

$$\boldsymbol{\phi} = \mathbf{N}_\phi \tilde{\boldsymbol{\phi}} \tag{13.78a}$$

and

$$w = \mathbf{N}_w \tilde{\mathbf{w}} + \mathbf{N}_{w\phi} \tilde{\boldsymbol{\phi}} \tag{13.78b}$$

This is precisely the form we obtained when seeking interpolations for the cylindrical bending of thick plates that gave exact interelement nodal results and forms the basis from which we now develop an interpolation for the general Reissner-Mindlin plate theory. Such an expression ensures that a higher order polynomial can be introduced for the representation of $w$ without adding additional element parameters. This procedure can, of course, be applied to any of the elements we have listed in Tables 13.5 and 13.6 to improve the accuracy attainable. We shall here develop such linking for two types of elements in which the essential interpolations are linear on each edge and one element where they are quadratic.

We thus improve the triangular T3S1B1 and by linking $L$ to its formulation we arrive at T3S1B1L. The same procedure can, of course, be applied to the quadrilaterals Q4S1B1 and Q4S2B2 of which only the second is unconditionally stable.

In the context of thick plates linked interpolation on a three-node triangle was introduced by Lynn and co-workers [93,94] and first extended to permit also thin plate analysis by Xu [89,95]. Similar interpolations have also been used by Tessler and Hughes [96,97] and termed "anisoparametric." Additional presentations dealing with the simple triangular element with nine degrees of freedom in its reduced form [see Eq. (13.102)] have been given by Taylor and co-workers [90,98,99].

Quadrilateral elements employing linked interpolation have been developed by Crisfield [100], Zienkiewicz and co-workers [85,87], and Auricchio and Taylor [88].

### 13.11.1 Linking function for linear triangles and quadrilaterals

Based on the linked form for cylindrical bending given in Section 13.2.4 the function $\mathbf{N}_{w\phi}$ for the linear triangle T3 and the linear quadrilaterals may be expressed as

$$w = N_a \tilde{w}_a - \frac{1}{8}\sum_{a=1}^{n_{\text{el}}} (N_{w\phi})_a l_{bc}(\phi_{sb} - \phi_{sc}) \tag{13.79}$$

where $n_{\text{el}}$ is the number of vertex nodes on the element (i.e., 3 or 4), $l_{bc}$ is the length of the $b$–$c$ side, $\phi_{sb}$ is the rotation at node $b$ in the tangent direction of the $a$th side, and $(N_{w\phi})_a$ are shape functions defining the quadratic $w$ along the side but still maintaining zero $w$ at corner nodes. For the triangular element these are the shape functions identical to those arising in the plane six-node element at mid-side nodes and are given by

$$\mathbf{N}_{w\phi} = 4\left[\, L_1L_2,\ L_2L_3,\ L_3L_1 \,\right] \tag{13.80}$$

and for the quadrilateral element these are the shape functions for the eight-node serendipity functions given by

$$\mathbf{N}_{w\phi} = \frac{1}{2}\left[\, (1-\xi^2)(1-\eta),\ (1+\xi)(1-\eta^2),\ (1-\xi^2)(1+\eta),\ (1-\xi)(1-\eta^2) \,\right] \tag{13.81}$$

**Example 13.1.** Derivation of linked function for three-node triangle

The development of a shape function for the three-node triangular element with a total of nine degrees of freedom is developed using the fact that both shear and moment in the equivalent cylindrical bending form were constant. A process to develop a linked interpolation for the plate transverse displacement, $w$, can start from a full quadratic expansion written in hierarchical form. Thus, for a triangle we have the interpolation in area coordinates

$$w = L_1\tilde{w}_1 + L_2\tilde{w}_2 + L_3\tilde{w}_3 + 4L_1L_2\tilde{\alpha}_1 + 4L_2L_3\tilde{\alpha}_2 + 4L_3L_1\tilde{\alpha}_3 \tag{13.82}$$

where $\tilde{w}_a$ are the nodal displacements and $\tilde{\alpha}_a$ are hierarchical parameters. The hierarchical parameters are then expressed in terms of rotational parameters. Along any edge, say the 1–2 edge (where $L_3 = 0$), the displacement is given by

$$w = L_1\tilde{w}_1 + L_2\tilde{w}_2 + 4L_1L_2\alpha_1 \tag{13.83}$$

The expression used to eliminate $\alpha_1$ is deduced by constraining the transverse edge shear strain to be a constant. Along the edge the transverse shear strain is given by

$$\gamma_{12} = \frac{\partial w}{\partial s} + \phi_s \tag{13.84}$$

where $s$ is the coordinate tangential to the edge and $\phi_s$ is the component of the rotation along the edge. The derivative of Eq. (13.83) along the edge is given by

$$\frac{\partial w}{\partial s} = \frac{\tilde{w}_2 - \tilde{w}_1}{l_{12}} + 4\frac{L_2 - L_1}{l_{12}}\tilde{\alpha}_1 \tag{13.85}$$

where $l_{12}$ is the length of the 1–2 side. Assuming a linear interpolation for $\phi_s$ along the edge we have

$$\phi_s = L_1\tilde{\phi}_{s1} + L_2\tilde{\phi}_{s2} \tag{13.86}$$

which, after noting that $L_1 + L_2 = 1$ along the edge, may also be expressed as

$$\phi_s = \frac{1}{2}(\tilde{\phi}_{s1} + \tilde{\phi}_{s2}) - \frac{1}{2}(\tilde{\phi}_{s1} - \tilde{\phi}_{s2})(L_2 - L_1) \tag{13.87}$$

The transverse shear may now be given as

$$\gamma_{12} = \frac{\tilde{w}_2 - \tilde{w}_1}{l_{12}} + \frac{1}{2}(\tilde{\phi}_{s1} + \tilde{\phi}_{s2}) + \left[\frac{4}{l_{12}}\tilde{\alpha}_1 - \frac{1}{2}(\tilde{\phi}_{s1} - \tilde{\phi}_{s2})\right](L_2 - L_1) \tag{13.88}$$

Constraining the strain to be constant gives

$$\tilde{\alpha}_1 = \frac{l_{12}}{8}(\tilde{\phi}_{s1} - \tilde{\phi}_{s2}) \tag{13.89}$$

yielding the "linked" edge interpolation

$$w = L_1\tilde{w}_1 + L_2\tilde{w}_2 + \frac{1}{2}l_{12}L_1L_2(\tilde{\phi}_{s1} - \tilde{\phi}_{s2}) \tag{13.90}$$

The normal rotations may now be expressed in terms of the nodal Cartesian components by using

$$\phi_s = \cos\psi_{12}\phi_x + \sin\psi_{12}\phi_y \tag{13.91}$$

where $\psi_{12}$ is the angle that the normal to the edge makes with the $x$ axis. Repeating this process for the other two edges gives the final interpolation for the transverse displacement.

A similar process can be followed to develop the linked interpolations for the quadrilateral element. The reader can verify that the use of the constant $1/8$ ensures that constant shear strain on the element side occurs. Further, a rigid body rotation with $\tilde{\boldsymbol{\phi}}_a = \tilde{\boldsymbol{\phi}}_b$ in the element causes no straining. Finally, with rotation $\tilde{\boldsymbol{\phi}}_a$ being the same for adjacent elements $C_0$ continuity is ensured.

### 13.11.2 Linked interpolation for quadratic elements

The interpolation for the quadratic element shown in Fig. 13.22 proceeds in a similar way to the linear form just presented. The development of the quadratic element, which is similar to the Zienkiewicz-Lefebvre element described above in Section 13.10, was presented by Auricchio and Lovadina [101]. The development starts with quadratic interpolation for the displacement $w$ and rotation $\boldsymbol{\phi}$. The transverse displacement is then increased to a complete cubic using the linked concept in which $w$ is written as

$$\begin{aligned} w = &\sum_{a=1}^{6} N_a(L_k)\tilde{w}_a + N_h\tilde{w}_h \\ &+ \frac{1}{3}\sum_{a=1}^{3} l_{ab}L_aL_b(L_b - L_a)\mathbf{t}_{ab}^T(\tilde{\boldsymbol{\phi}}_a + \tilde{\boldsymbol{\phi}}_b - 2\,\tilde{\boldsymbol{\phi}}_{a+3});\ a = 1, 2, 3;\ b = 2, 3, 1 \end{aligned} \tag{13.92}$$

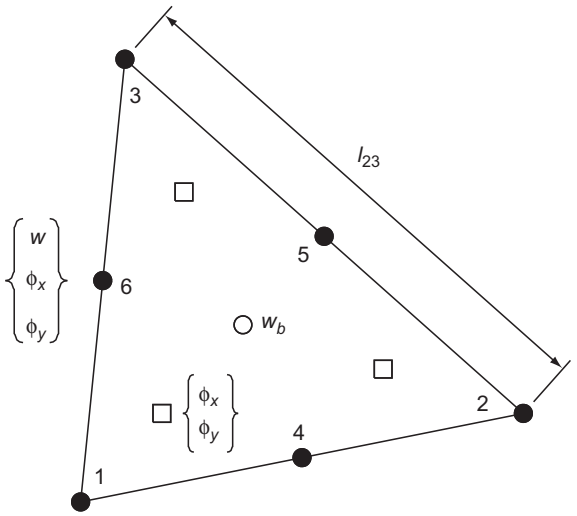



**FIGURE 13.22**
Quadratic linked element.

with

$$N_h = L_1 L_2 L_3 \tag{13.93}$$

Here $w_h$ is an hierarchical parameter. We note that the bubble mode can be retained as a displacement degree of freedom or its gradient added as an enhanced mode to shearing strains.[5] As noted above the rotation interpolation is given by the quadratic form

$$\boldsymbol{\phi} = \sum_{a=1}^{6} N_a(L_k)\tilde{\boldsymbol{\phi}}_a \tag{13.94}$$

To satisfy the mixed patch test the strains for the linked element need additional enhanced terms which are computed from the derivatives and values of additional displacement functions. Accordingly, we consider the enhanced curvature given by

$$\boldsymbol{\chi} = \begin{Bmatrix} \chi_x \\ \chi_y \\ \chi_{xy} \end{Bmatrix} = \sum_{a=1}^{6} \begin{bmatrix} \dfrac{\partial N_a}{\partial x} & 0 \\ 0 & \dfrac{\partial N_a}{\partial y} \\ \dfrac{\partial N_a}{\partial y} & \dfrac{\partial N_a}{\partial x} \end{bmatrix} \begin{Bmatrix} \tilde{\phi}_x \\ \tilde{\phi}_y \end{Bmatrix} + \begin{Bmatrix} \dfrac{\partial \phi_x^{en}}{\partial x} \\ \dfrac{\partial \phi_y^{en}}{\partial y} \\ \dfrac{\partial \phi_x^{en}}{\partial y} + \dfrac{\partial \phi_y^{en}}{\partial x} \end{Bmatrix} \tag{13.95}$$

where

$$\boldsymbol{\phi}^{en} = \sum_{b=1}^{3} \left( L_b - \frac{1}{3} \right) N_h \, \tilde{\boldsymbol{\beta}} + (\nabla N_h) N_h \, \tilde{\phi}_h \tag{13.96}$$

in which $\tilde{\boldsymbol{\beta}}$ and $\tilde{\phi}_h$ add seven parameters.

[5] Hierarchical form is preferred for dynamic problems since, then, it does not affect the inertial terms.

The enhanced transverse shear strain is given by

$$\boldsymbol{\gamma} = \left\{ \begin{array}{c} \gamma_x \\ \gamma_y \end{array} \right\} = \left\{ \begin{array}{c} \dfrac{\partial w}{\partial x} + \phi_x \\ \dfrac{\partial w}{\partial y} + \phi_y \end{array} \right\} + \boldsymbol{\phi}^{en} \tag{13.97}$$

where we assume that $w_h$ is treated as a displacement parameter.

Finally, the shear interpolation is given by

$$\mathbf{Q} = \sum_{a=1}^{3} L_a \tilde{\mathbf{Q}}_a + \boldsymbol{\nabla} N_h \widetilde{Q}_h \tag{13.98}$$

For the form given above each node has three degrees of freedom ($\tilde{w}_a$, $\phi_{xa}$, and $\phi_{ya}$), one internal $n_w$, seven internal $n_\phi$, and seven internal $n_Q$. The element satisfies the count condition of the mixed patch test for all element configurations including a single element with all external degrees of freedom restrained.

**Example 13.2.** Clamped plate by linked elements

For a clamped plate a solution based on the Reissner-Mindlin plate theory has been given by Chinosi and Lovadina [102]. Using an inverse method we may write the solution as

$$\begin{aligned} w &= w_b + w_s \quad \text{with} \quad \boldsymbol{\phi} = -\boldsymbol{\nabla} w_b \\ w_b &= \frac{1}{3} x^3 (x-1)^3 y^3 (y-1)^3 \\ w_s &= -\frac{h^2}{6\kappa(1-\nu)} \nabla^2 w_b \end{aligned}$$

which gives the load

$$\begin{aligned} q = D \; \Big[ & 12y(y-1)(5x(x-1)+1)(2y^2(y-1)^2 + x(x-1)(5y(y-1)+1) \\ & + 12x(x-1)(5y(y-1)+1)(2x^2(x-1)^2 + y(y-1)(5x(x-1)+1) \Big] \end{aligned}$$

This solution is used to illustrate the convergence of the energy error for a clamped plate as shown in Fig. 13.23.

## 13.12 Discrete "exact" thin plate limit

Discretization of Eq. (13.71) using interpolations of the form

$$w = \mathbf{N}_w \tilde{\mathbf{w}} + \mathbf{N}_{w\phi} \tilde{\boldsymbol{\phi}} \,, \qquad \boldsymbol{\phi} = \mathbf{N}_\phi \tilde{\boldsymbol{\phi}} + \mathbf{N}_h \Delta \tilde{\boldsymbol{\phi}}_h \,, \qquad \mathbf{Q} = \mathbf{N}_q \tilde{\mathbf{Q}} \tag{13.99}$$

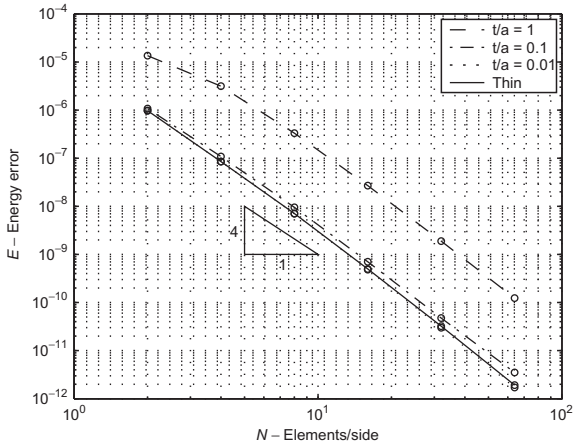



**FIGURE 13.23**
Energy error for quadratic linked element: clamped plate.

leads to the algebraic system of equations

$$\begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{K}_{qw}^{\mathrm{T}} \\ \mathbf{0} & \mathbf{K}_{\phi\phi} & \mathbf{K}_{h\phi}^{\mathrm{T}} & \mathbf{K}_{q\phi}^{\mathrm{T}} \\ \mathbf{0} & \mathbf{K}_{h\phi} & \mathbf{K}_{hh} & \mathbf{K}_{qh}^{\mathrm{T}} \\ \mathbf{K}_{qw} & \mathbf{K}_{q\phi} & \mathbf{K}_{qh} & \mathbf{K}_{qq} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{w}} \\ \tilde{\boldsymbol{\phi}} \\ \Delta\tilde{\boldsymbol{\phi}}_h \\ \tilde{\mathbf{Q}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_w \\ \mathbf{f}_\phi \\ \mathbf{0} \\ \mathbf{0} \end{Bmatrix} \tag{13.100}$$

where, for simplicity, only the forces $\mathbf{f}_w$ and $\mathbf{f}_\phi$ due to transverse load $q$ and boundary conditions are included [see Eq. (13.63d)]. The arrays appearing in Eq. (13.100) are given by[6]

$$\begin{aligned} \mathbf{K}_{\phi\phi} &= \int_\Omega (\mathcal{S}\mathbf{N}_\phi)^{\mathrm{T}}\mathbf{D}(\mathcal{S}\mathbf{N}_\phi)\,\mathrm{d}\Omega, \quad \mathbf{K}_{qq} = -\int_\Omega \mathbf{N}_q\boldsymbol{\alpha}^{-1}\mathbf{N}_q\,\mathrm{d}\Omega, \\ \mathbf{K}_{h\phi} &= \int_\Omega (\mathcal{S}\mathbf{N}_h)^{\mathrm{T}}\mathbf{D}(\mathcal{S}\mathbf{N}_\phi)\,\mathrm{d}\Omega, \quad \mathbf{K}_{q\phi} = \int_\Omega \mathbf{N}_q^{\mathrm{T}}\left[\nabla\mathbf{N}_{w\phi} - \mathbf{N}_\phi\right]\mathrm{d}\Omega, \\ \mathbf{K}_{hh} &= \int_\Omega (\mathcal{S}\mathbf{N}_h)^{\mathrm{T}}\mathbf{D}(\mathcal{S}\mathbf{N}_h)\,\mathrm{d}\Omega, \quad \mathbf{K}_{qh} = -\int_\Omega \mathbf{N}_q^{\mathrm{T}}\mathbf{N}_h\,\mathrm{d}\Omega, \\ \mathbf{K}_{qw} &= \int_\Omega \mathbf{N}_q^{\mathrm{T}}\nabla\mathbf{N}_w\,\mathrm{d}\Omega \end{aligned} \tag{13.101}$$

Adopting a static condensation process at the element level [103] in which the internal rotational parameters are eliminated first, followed by the shear parameters,

[6] Some enhanced terms also exist in some of the matrices but do not affect the basic structure shown.

yields the element stiffness matrix in terms of the element $\tilde{\mathbf{w}}$ and $\tilde{\boldsymbol{\phi}}$ parameters given by

$$\begin{bmatrix} \mathbf{K}_{qw}^{\mathrm{T}}\mathbf{A}_{qq}^{-1}\mathbf{K}_{qw} & -\mathbf{K}_{qw}^{\mathrm{T}}\mathbf{A}_{qq}^{-1}\mathbf{A}_{q\phi} \\ -\mathbf{A}_{q\phi}^{\mathrm{T}}\mathbf{A}_{qq}^{-1}\mathbf{K}_{qw} & -\mathbf{A}_{\phi\phi}+\mathbf{A}_{q\phi}^{\mathrm{T}}\mathbf{A}_{qq}^{-1}\mathbf{A}_{q\phi} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{w}} \\ \tilde{\boldsymbol{\phi}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_w \\ \mathbf{f}_\phi \end{Bmatrix} \tag{13.102}$$

in which

$$\begin{aligned} &\mathbf{A}_{qq} = \mathbf{K}_{qb}\mathbf{K}_{bb}^{-1}\mathbf{K}_{qb}^{\mathrm{T}} - \mathbf{K}_{qq},\ \mathbf{A}_{q\phi} = \mathbf{K}_{qb}\mathbf{K}_{hh}^{-1}\mathbf{K}_{h\phi} - \mathbf{K}_{q\phi} \\ &\mathbf{A}_{\phi\phi} = \mathbf{K}_{h\phi}^{\mathrm{T}}\mathbf{K}_{hh}^{-1}\mathbf{K}_{h\phi} - \mathbf{K}_{\phi\phi} \end{aligned} \tag{13.103}$$

This solution strategy requires the inverse of $\mathbf{K}_{hh}$ and $\mathbf{A}_{qq}$ only. In particular, we note that the inverse of $\mathbf{A}_{qq}$ can be performed even if $\mathbf{K}_{qq}$ is zero (provided the other term is nonsingular). The vanishing of $\mathbf{K}_{qq}$ defines the *thin plate limit*. Thus, the above strategy leads to a solution process in which the thin plate limit is defined without recourse to a penalty method. Indeed, all terms in the process generally are not subject to ill-conditioning due to differences in large and small numbers. In the context of thick and thin plate analysis this solution strategy has been exploited with success in Ref. [88,90].

## 13.13 Limitations of plate theory

All "robust" elements of the thick plate kind can be easily mapped isoparametrically and their performance remains excellent and convergent. Of course, when the span-to-thickness ratio decreases and thus shear deformation importance increases, the thick plate elements are capable of yielding results not obtainable with thin plate theory. In Table 13.7 we show some results for a simply supported, uniformly loaded plate for two $L/h$ ratios and in Table 13.8 we show results for the clamped uniformly loaded plate for the same ratios. In this example we show also the effect of the *hard* and *soft* simple support conditions. In the hard support we assume just as in thin plates that the rotation along the support ($\phi_s$) is zero. In the soft support case we take, more rationally, a zero twisting moment along the support (see Section 13.3.1).

**Table 13.7** Center Displacement of a Simply Supported Plate under Uniform Load for Two $L/h$ Ratios: $E = 10.92$, $\nu = 0.3$, $L = 10$, $q = 1$

| | $L/h = 10$, $w \times 10^{-1}$ | | $L/h = 1000$, $w \times 10^{-7}$ | |
|---|---|---|---|---|
| **Mesh, *M*** | **Hard Support** | **Soft Support** | **Hard Support** | **Soft Support** |
| 2 | 4.2626 | 4.6085 | 4.0389 | 4.2397 |
| 4 | 4.2720 | 4.5629 | 4.0607 | 4.1297 |
| 8 | 4.2727 | 4.5883 | 4.0637 | 4.0928 |
| 16 | 4.2728 | 4.6077 | 4.0643 | 4.0773 |
| 32 | 4.2728 | 4.6144 | 4.0644 | 4.0700 |
| Series | 4.2728 | | 4.0624 | |

**Table 13.8** Center Displacement of a Clamped Plate under uniform Load for Two $L/h$ Ratios: $E = 10.92$, $\nu = 0.3$, $L = 10$, $q = 1$

| Mesh, $M$ | $L/h = 10$, $w \times 10^{-1}$ | $L/h = 1000$, $w \times 10^{-7}$ |
|---|---|---|
| 2 | 1.4211 | 1.1469 |
| 4 | 1.4858 | 1.2362 |
| 8 | 1.4997 | 1.2583 |
| 16 | 1.5034 | 1.2637 |
| 32 | 1.5043 | 1.2646 |
| Series | 1.499 [86] | 1.2653 |

It is immediately evident that

1. The thick plate ($L/h = 10$) shows deflections converging to very different values depending on the support conditions, both being considerably larger than those given by thin plate theory.
2. For the thin plate ($L/h = 1000$) the deflections converge uniformly to the thin (Kirchhoff) plate results for *hard* support conditions, but for *soft* support conditions give answers about 0.2% higher in center deflection.

It is perhaps an insignificant difference that occurs in this example between the support conditions but this can be more pronounced in different plate configurations.

## 13.14 Concluding remarks

The simplicity of deriving and using elements in which independent interpolation of rotations and displacements is postulated and shear deformations are included assures popularity of the approach. The final degrees of freedom used are exactly the same type as those used in the direct approach to thin plate forms and at no additional complexity when shear deformability is included in the analysis. The ease of element distortion will make elements of the type discussed here the first choice for curved element solutions.

If care is used to ensure robustness, thick plate elements of the type discussed in this chapter are generally applicable and indeed could be used with similar restrictions to other finite element approximations requiring $C_1$ continuity in the limit.

## 13.15 Problems

**13.1** Using the interpolations given in (13.9) determine the element load vector $\mathbf{f}^e$ for a uniform loading $q_z$.

**13.2** Using the interpolations given in (13.9) determine the element mass matrix $\mathbf{M}^e$.

**13.3** Using the interpolations given in (13.11) determine the element load vector $\mathbf{f}^e$ for a uniform loading $q_z$ for a two-node element.

**13.4** Using the interpolations given in (13.9) determine the element mass matrix $\mathbf{M}^e$ for a two-node element.

**13.5** Using the interpolations given in (13.11) determine the element load vector $\mathbf{f}^e$ for a uniform loading $q_z$ for a three-node element.

**13.6** Using the interpolations given in (13.11) determine the element mass matrix $\mathbf{M}^e$ for a three-node element.

**13.7** Derive the interpolations that give an exact solution for cylindrical bending of thick plates under static loads. Express all parameters in total not hierarchic form.

**13.8** Using the interpolations given in (13.14) determine the expressions for the element curvature and shearing strain.

**13.9** Using the interpolations given in (13.14) determine the element load vector $\mathbf{f}^e$ for a uniform loading $q_z$.

**13.10** Using the interpolations given in (13.14) determine the element mass matrix $\mathbf{M}^e$. Specialize for the case where the hierarchic term is ignored.

**13.11** Given an axisymmetric displacement field expressed by

$$u_r(r,t) = z\,\phi_r(r,t),\ u_z = w(r,t),\ \text{and}\ u_\theta = 0$$

and the equations for axisymmetry from Chapter 2, repeat the steps described in Section 13.2.1.

- **(a)** Define the the strain-displacement relations for a thick plate.
- **(b)** Define the plate forces and compute equilibrium equations.
- **(c)** Define the plate constitutive relations in terms of $E$, $\nu$, $G$, and $\kappa$ and write a weak form for the problem.
- **(d)** Integrate the weak form to deduce the governing differential equations.

**13.12** Use the displacement field defined in Problem 13.11 and strain-displacement relations from Chapter 2 to write an irreducible weak form for plate bending of axisymmetric plates subjected to axisymmetric loading. Assume elastic properties are isotropic and expressed in terms of $E$, $\nu$, and $G$ with a shear correction factor $\kappa$.

**13.13** Write a MATLAB or GNU Octave program to solve axisymmetric plate bending defined by Problem 13.12. Assume static behavior and use elements with shape functions described by (13.14).
Solve a problem with $a = 0$, $b = 10$, $h = 2$ for a concentrated load $Q = 1$ at $a = 0$ and fixed support at $r = b$. Let $E = 200$, $\nu = 0.3$, and $\kappa = 5/6$. Use meshes of 4, 8, 16, 32, and 64 elements. Plot the central displacement.
Repeat the above problem for a uniformly distributed loading $q = 1$.

## References

[1] G. Kirchhoff, Über das Gleichqewicht und die Bewegung einer elastichen Scheibe, J. Reine Angew. Math. 40 (1850) 51–88.

[2] S.P. Timoshenko, S. Woinowski-Krieger, Theory of Plates and Shells, second ed., McGraw-Hill, New York, 1959.
[3] L. Bucciarelly, N. Dworsky, Sophie Germain, An Essay on the History of Elasticity, Reidel, New York, 1980.
[4] E. Reissner, Reflections on the theory of elastic plates, Appl. Mech. Rev. 38 (1985) 1453–1464.
[5] E. Reissner, The effect of transverse shear deformation on the bending of elastic plates, J. Appl. Mech. 12 (1945) 69–76.
[6] R.D. Mindlin, Influence of rotatory inertia and shear in flexural motions of isotropic elastic plates, J. Appl. Mech. 18 (1951) 31–38.
[7] I.H. Shames, F.A. Cozzarelli, Elastic and Inelastic Stress Analysis, Taylor & Francis, Washington, D.C., 1997 (Revised printing).
[8] E.P. Popov and T.A. Balan. Engineering Mechanics of Solids, second ed., Prentice Hall, Upper Saddle River, N.J., 1999.
[9] F.P. Beer Jr., E.R. Johnson, and W.E. Clausen. Mechanics of Materials, third ed., McGraw-Hill, Boston, 2002.
[10] P. Tong, Exact solution of certain problems by the finite element method, J. AIAA 7 (1969) 179–180.
[11] T.J.R. Hughes, R.L. Taylor, W. Kanoknukulchai, A simple and efficient finite element for plate bending, Int. J. Num. Methods Eng. 11 (1977) 1529–1543.
[12] I. Babuška, T. Scapolla, Benchmark computation and performance evaluation for a rhombic plate bending problem, Int. J. Num. Methods Eng. 28 (1989) 155–180.
[13] I. Babuška. The stability of domains and the question of formulation of plate problems. Appl. Math. (1962) 463–467.
[14] H.L. Langhaar, Note on energy of bending of plates, J. Appl. Mech., ASME 19(2) (1952) 228.
[15] I. Babuška, J. Pitkäranta, The plate paradox for hard and soft simple support, SIAM J. Num. Anal 21 (1990) 551–576.
[16] B.M. Irons, J.K. Draper, Inadequacy of nodal connections in a stiffness solution for plate bending, J. AIAA 3 (1965) 5.
[17] R.W. Clough, J.L. Tocher, Finite element stiffness matrices for analysis of plate bending, in: Procedings of the First Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-66-80, Wright Patterson Air Force Base, Ohio, October 1966, pp. 515–545.
[18] R.W. Clough and C.A. Felippa, A refined quadrilateral element for analysis of plate bending. in: Proceedings of the Second Conference Matrix Methods in Structural Mechanics, vol. AFFDL-TR-68-150, Wright Patterson Air Force Base, Ohio, October 1968, pp. 399–440.
[19] B. Fraeijs de Veubeke, Bending and stretching of plates. Special models for upper and lower bounds, in: Procedings of the First Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-66-80, Wright Patterson Air Force Base, Ohio, October 1966, pp. 863–886.

[20] B. Fraeijs de Veubeke, A conforming finite element for plate bending, Int. J. Solids Struct. 4 (1968) 95–108.
[21] G.P. Bazeley, Y.K. Cheung, B.M. Irons, and O.C. Zienkiewicz, Triangular elements in bending—conforming and non-conforming solutions, in: Proceedings of the First Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-66-80, Wright Patterson Air Force Base, Ohio, October 1966, pp. 547–576.
[22] B.M. Irons, A conforming quartic triangular element for plate bending, Int. J. Num. Methods Eng. 1 (1969) 29–46.
[23] F.K. Bogner, R.L. Fox, and L.A. Schmit, The generation of interelement-compatible stiffness and mass matrices by the use of interpolation formulae, in: Procedings of the First Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-66-80, Wright Patterson Air Force Base, Ohio, October 1966, pp. 397–443.
[24] I.M. Smith, W. Duncan, The effectiveness of nodal continuities in finite element analysis of thin rectangular and skew plates in bending, Int. J. Num. Methods Eng. 2 (1970) 253–258.
[25] K. Bell, A refined triangular plate bending element, Int. J. Num. Methods Eng. 1 (1969) 101–122.
[26] G.A. Butlin and R. Ford. A Compatible Plate Bending Element, Technical Report 68–15, University of Leicester Engineering Department, 1968.
[27] G.R. Cowper, E. Kosko, G.M. Lindberg, and M.D. Olson, Formulation of a new triangular plate bending element, Trans. Canad. Aero-Space Inst. 1 (1968) 86–90 (see also NRC Aero, Report LR514, 1968).
[28] W. Bosshard, Ein neues vollverträgliches endliches Element für Plattenbiegung, Mt. Ass. Bridge Struct. Eng. Bull. 28 (1968) 27–40.
[29] J.H. Argyris, I. Fried, and D.W. Scharpf, The TUBA family of plate elements for the matrix displacement method, Aeronaut. J. Roy. Aeronaut. Soc. 72 (1968) 701–709.
[30] W. Visser, The finite element method in deformation and heat conduction problems, Dr. Wiss. dissertation, Technische Hochschule, Delft, 1968.
[31] B.M. Irons, J.G. Ergatoudis, and O.C. Zienkiewicz, Comments on 'complete polynomial displacement fields for finite element method' (by P.C. Dunne), Trans. Roy. Aeronaut. Soc. 72 (1968) 709.
[32] O.C. Zienkiewicz, Y.K. Cheung, The finite element method for analysis of elastic isotropic and orthotropic slabs, Proc. Inst. Civ. Eng. 28 (1964) 471–488.
[33] R.W. Clough, The finite element method in structural mechanics, in: O.C. Zienkiewicz, G.S. Holister (Eds.), Stress Analysis, John Wiley & Sons, Chichester, 1965, Chapter 7.
[34] D.J. Dawe, Parallelogram element in the solution of rhombic cantilever plate problems, J. Strain Anal. 1 (1966) 223–230.

[35] J.H. Argyris, Continua and discontinua, in: Procedings of the First Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-66-80, Wright Patterson Air Force Base, Ohio, October 1966, pp. 11–189.
[36] L.S.D. Morley, On the constant moment plate bending element, J. Strain Anal. 6 (1971) 20–24.
[37] R.D. Wood, A shape function routine for the constant moment triangular plate bending element, Eng. Comput. 1 (1984) 189–198.
[38] R. Narayanaswami, New triangular plate bending element with transverse shear flexibility, J. AIAA 12 (1974) 1761–1763.
[39] A. Razzaque, Program for triangular bending element with derivative smoothing, Int. J. Num. Methods Eng. 5 (1973) 588–589.
[40] B.M. Irons, A. Razzaque, Shape function formulation for elements other than displacement models. in: C.A. Brebbia, H. Tottenham, (Eds.), Proceedings of the International Conference on Variational Methods in Engineering, vol. II, Southampton University Press, 1973, pp. 4/59–4/72.
[41] J.E. Walz, R.E. Fulton, N.J. Cyrus, Accuracy and convergence of finite element approximations. in: Proceedings of the Second Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-68-150, Wright Patterson Air Force Base, Ohio, October 1968, pp. 995–1027.
[42] R.J. Melosh, Structural analysis of solids, J. Struct. Eng., ASCE, 4 (1963) 205–223.
[43] A. Adini and R.W. Clough, Analysis of Plate Bending by the Finite Element Method, Technical Report G-7337, Report to National Science Foundation USA, 1961.
[44] Y.K. Cheung, I.P. King, O.C. Zienkiewicz, Slab bridges with arbitrary shape and support conditions, Proc. Inst. Civ. Eng. 40 (1968) 9–36.
[45] J.L. Tocher and K.K. Kapur, Comment on basis of derivation of matrices for direct stiffness method (by R. Melosh), J. AIAA 3 (1965) 1215–1216.
[46] R.D. Henshell, D. Walters, G.B. Warburton, A new family of curvilinear plate bending elements for vibration and stability, J. Sound Vibr. 20 (1972) 327–343.
[47] P.G. Bergen, L. Hanssen, A new approach for deriving 'good' element stiffness matrices, in: J.R. Whiteman (Ed.), The Mathematics of Finite Elements and Applications, Academic Press, London, 1977, pp. 483–497.
[48] R.V. Southwell, Relaxation Methods in Theoretical Physics, Clarendon Press, Oxford, 1946.
[49] P.G. Bergan, M.K. Nygard, Finite elements with increased freedom in choosing shape functions, Int. J. Num. Methods Eng. 20 (1984) 643–663.
[50] C.A. Felippa, P.G. Bergan, A triangular plate bending element based on energy orthogonal free formulation, Comput. Methods Appl. Mech. Eng. 61 (1987) 129–160.
[51] A. Samuelsson, The global constant strain condition and the patch test, in: R. Glowinski, E.Y. Rodin, O.C. Zienkiewicz (Eds.), Energy Methods in Finite Element Analysis, John Wiley & Sons, Chichester, 1979, pp. 49–68, Chapter 3.

[52] B. Specht, Modified shape functions for the three node plate bending element passing the patch test, Int. J. Num. Methods Eng. 26 (1988) 705–715.
[53] W. Ritz, Über eine neue Methode zur Lösung gewisser variationsproblem der mathematischen physik, J. die reine Angew. Math. 135 (1908) 1–61.
[54] B.G. Galerkin, Series solution of some problems in elastic equilibrium of rods and plates, Vestn. Inzh. Tech. 19 (1915) 897–908.
[55] H. Hencky, Der Spannungszustand in Rechteckigen Platten, Technical Report VI, 94 S, München, 1913.
[56] H. Hencky, Der Spannungszustand in rechteckigen Platten. Ph.D thesis, Darmstadt, Published by R. Oldenbourg, Munich and Berlin, Germany, 1913.
[57] I.A. Wojtaszak, The calculation of maximum deflection, moment, and shear for uniformly loaded rectangular plate with clamped edges, J. Appl. Mech., ASME, 59 (1937) A173–A176.
[58] R.L. Taylor, S. Govindjee, Solution of clamped rectangular plate problems, Commun. Num. Methods Eng. 20 (2004) 757–765.
[59] H. Marcus, Die Theorie elastisher Geweve und ihre Anwendung auf die Berechnung biegsamer Platten, Springer, Berlin, 1932.
[60] P. Ballesteros, S.L. Lee, Uniformly loaded rectangular plate supported at the corners, Int. J. Mech. Sci. 2 (1960) 206–211.
[61] S. Ahmad, B.M. Irons, and O.C. Zienkiewicz, Curved thick shell and membrane elements with particular reference to axi-symmetric problems. in: Proceedings of the Second Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-68-150, Wright Patterson Air Force Base, Ohio, October 1968, pp. 539–572.
[62] S. Ahmad, B.M. Irons, O.C. Zienkiewicz, Analysis of thick and thin shell structures by curved finite elements, Int. J. Num. Methods Eng. 2 (1970) 419–451.
[63] S. Ahmad. Curved finite elements in the analysis of solids, shells and plate structures. Ph.D thesis, Department of Civil Engineering, University of Wales, Swansea, 1969.
[64] O.C. Zienkiewicz, J. Too, R.L. Taylor, Reduced integration technique in general analysis of plates and shells, Int. J. Num. Methods Eng. 3 (1971) 275–290.
[65] S.F. Pawsey, R.W. Clough, Improved numerical integration of thick slab finite elements, Int. J. Num. Methods Eng. 3 (1971) 575–586.
[66] O.C. Zienkiewicz, E. Hinton, Reduced integration, function smoothing and non-conformity in finite element analysis, J. Franklin Inst. 302 (1976) 443–461.
[67] E.D.L. Pugh, E. Hinton, O.C. Zienkiewicz, A study of quadrilateral plate bending elements with reduced integration, Int. J. Num. Methods Eng. 12 (1978) 1059–1079.
[68] F. Gruttmann, W. Wagner, A stabilized one-point integrated quadrilateral Reissner-Mindlin plate element, Int. J. Num. Methods Eng. 61 (2004) 2273–2295.

[69] T.J.R. Hughes, M. Cohen, The heterosis finite element for plate bending, Comput. Struct. 9 (1978) 445–450.
[70] T.J.R. Hughes, M. Cohen, M. Harou, Reduced and selective integration techniques in the finite element analysis of plates, Nucl. Engineering and Design 46 (1978) 203–222.
[71] E. Hinton, N. Bićanić, A comparison of Lagrangian and serendipity Mindlin plate elements for free vibration analysis, Computers and Structures 10 (1979) 483–493.
[72] R.D. Cook, Concepts and Applications of Finite Element Analysis, John Wiley & Sons, Chichester, 1982.
[73] R.L. Taylor, O.C. Zienkiewicz, J.C. Simo, A.H.C. Chan, The patch test—a condition for assessing FEM convergence, Int. J. Num. Methods Eng. 22 (1986) 39–62.
[74] O.C. Zienkiewicz, S. Qu, R.L. Taylor, S. Nakazawa, The patch test for mixed formulations, Int. J. Num. Methods Eng. 23 (1986) 1873–1883.
[75] O.C. Zienkiewicz, D. Lefebvre, A robust triangular plate bending element of the Reissner-Mindlin type, Int. J. Num. Methods Eng. 26 (1988) 1169–1184.
[76] O.C. Zienkiewicz, J.P. Vilotte, S. Toyoshima, S. Nakazawa, Iterative method for constrained and mixed approximation. An inexpensive improvement of FEM performance, Comput. Methods Appl. Mech. Eng., 51 (1985) 3–29.
[77] D.S. Malkus, T.J.R. Hughes, Mixed finite element methods—Reduced and selective integration techniques: A unification of concepts, Comput. Methods Appl. Mech. Eng. 15 (1978) 63–81.
[78] K.-J. Bathe, Finite Element Procedures, Prentice Hall, Englewood Cliffs, N.J., 1996.
[79] W.X. Zhong, FEM patch test and its convergence, Technical Report Report 97-3001, Research Institute Engineering Mechanics, Dalian University of Technology, 1997 (in Chinese).
[80] W.X. Zhong, Convergence of FEM and the conditions of patch test, Technical Report Report 97-3002, Research Institute Engineering Mechanics, Dalian University of Technology, 1997. (in Chinese).
[81] I. Babuška, R. Narasimhan, The Babuška-Brezzi condition and the patch test: An example, Comput. Methods Appl. Mech. Eng. 140 (1997) 183–199.
[82] O.C. Zienkiewicz, R.L. Taylor, The finite element patch test revisited: A computer test for convergence, validation and error estimates, Comput. Methods Appl. Mech. Eng. 149 (1997) 523–544.
[83] T.J.R. Hughes, W.K. Liu, Implicit-explicit finite elements in transient analyses. Part I and Part II, J. Appl. Mech. 45 (1978) 371–378.
[84] K.-J. Bathe and L.W. Ho, Some results in the analysis of thin shell structures, in: W. Wunderlich et al., (Eds.), Nonlinear Finite Element Analysis in Structural Mechanics, Springer-Verlag, Berlin, 1981, pp. 122–156.
[85] O.C. Zienkiewicz, M. Huang, J. Wu, S. Wu, A new algorithm for the coupled soil-pore fluid problem, Shock Vibration 1 (1993) 3–14.

[86] O.C. Zienkiewicz, Z. Xu, L.F. Zeng, A. Samuelsson, N.-E. Wiberg, Linked interpolation for Reissner-Mindlin plate elements. Part I—a simple quadrilateral element, Int. J. Num. Methods Eng. 36 (1993) 3043–3056.
[87] Z. Xu, O.C. Zienkiewicz, L.F. Zeng, Linked interpolation for Reissner-Mindlin plate elements. Part III—an alternative quadrilateral, Int. J. Num. Methods Eng., 36 (1993) 3043–3056.
[88] F. Auricchio, R.L. Taylor, A shear deformable plate element with an exact thin limit, Comput. Methods Appl. Mech. Eng. 118 (1994) 393–412.
[89] Z. Xu, A simple and efficient triangular finite element for plate bending, Acta Mech. Sinica 2 (1986) 185–192.
[90] F. Auricchio, R.L. Taylor, A triangular thick plate finite element with an exact thin limit, Finite Elemen. Anal. Des. 19 (1995) 57–68.
[91] D.N. Arnold, R.S. Falk, A uniformly accurate finite element method for Mindlin-Reissner plate, Technical Report IMA Preprint Series No. 307, Institute for Mathematics and its Application, University of Maryland, 1987.
[92] T. Tu, Performance of Reissner-Mindlin elements, Ph.D thesis, Rutgers University, Department of Mathematics, 1998.
[93] L.F. Greimann, P.P. Lynn, Finite element analysis of plate bending with transverse shear deformation, Nucl. Eng. Des. 14 (1970) 223–230.
[94] P.P. Lynn, B.S. Dhillon, Triangular thick plate bending elements, in: Proceedings the First International Conference on Structural Mechanics in Reactor Technology, Berlin, 1971, p. M 6/5.
[95] Z. Xu, A thick-thin triangular plate element, Int. J. Num. Methods Eng. 33 (1992) 963–973.
[96] A. Tessler, T.J.R. Hughes, A three node Mindlin plate element with improved transverse shear, Comput. Methods Appl. Mech. Eng. 50 (1985) 71–101.
[97] A. Tessler, A $C^0$ anisoparametric three node shallow shell element, Comput. Methods Appl. Mech. Eng. 78, 89–103.
[98] P. Papadopoulos, R.L. Taylor, A triangular element based on Reissner-Mindlin plate theory, Int. J. Num. Methods Eng. 30 (1990) 1029–1049.
[99] R.L. Taylor, F. Auricchio, Linked interpolation for Reissner-Mindlin plate elements: Part II—a simple triangle, Int. J. Num. Methods Eng. 36 (1993) 3057–3066.
[100] M.A. Crisfield, Non-linear Finite Element Analysis of Solids and Structures, vol. 1, John Wiley & Sons, Chichester, 1991.
[101] F. Auricchio, C. Lovadina, Analysis of kinematic linked interpolation methods for Reissner-Mindlin plate problems, Comput. Methods Appl. Mech. Eng. 190 (2001) 2465–2482.
[102] C. Chinosi, C. Lovadina, Numerical analysis of some mixed finite element methods for Reissner-Mindlin plates, Comput. Mech. 16 (1995) 36–44.
[103] E.L. Wilson, The static condensation algorithm, Int. J. Num. Methods Eng. 8 (1974) 199–203.

CHAPTER

# 14 Shells as a Special Case of Three-Dimensional Analysis

## 14.1 Introduction

A shell is, in essence, a structure that can be derived from a plate by initially forming the middle surface as a singly or doubly curved surface. The same assumptions as used in thin plates regarding the transverse distribution of strains and stresses are again valid. However, the way in which the shell supports external loads is quite different from that of a flat plate. The stress resultants acting on the middle surface of the shell now have both tangential and normal components which carry a major part of the load, a fact that explains the economy of shells as load-carrying structures and their well-deserved popularity.

The derivation of detailed governing equations for a curved shell problem presents many difficulties and, in fact, leads to many alternative formulations, each depending on the approximations introduced. For details of classical shell treatment the reader is referred to standard texts on the subject, for example the well-known treatise by Flügge [1], the classical book by Timoshenko and Woinowski-Krieger [2], or detailed treatment in our companion volume [3].

In the early finite element treatment of shell problems the difficulties referred to above are eliminated, at the expense of introducing a further approximation. This approximation is of a physical, rather than mathematical, nature. In this it is assumed that the behavior of a continuously curved surface can be adequately represented by the behavior of a surface built up of small flat elements. Intuitively, as the size of the subdivision decreases it would seem that convergence must occur and indeed experience indicates such a convergence.

In the division of an arbitrary shell into flat elements only triangular elements can be used for doubly curved surfaces. Although the concept of the use of such elements in the analysis was suggested as early as 1961 by Greene et al. [4] the success of such analysis was hampered by the lack of a good stiffness matrix for triangular plate elements in bending [5–8]. The developments described in Chapter 13 open the way to adequate models for representing the behavior of shells with such a faceted division.

As an alternative curved shell elements can be used. Here curvilinear coordinates are essential. The physical approximation involved in flat elements is now avoided at the expense of reintroducing an arbitrariness of various shell theories. Several approaches using a direct displacement approximation are given in references [9–29] and the use of "mixed variational principles" is given in references [30–33].

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00014-9

Curved, general, thin shell forms will not be discussed here and instead a general formulation of thick curved shells (based directly on three-dimensional elasticity theory and avoiding the shell equation ambiguities) will be presented. The development of curved elements for general shell theories can also be effected in a direct manner. The interested reader is referred to references [3,34–36] for additional discussion on this approach. In many respects the differences in the two approaches are quite similar, as shown by Bischoff and Ramm [37].

In the analysis of solids the use of isoparametric, curved, two- and three-dimensional elements is particularly effective, as illustrated in Chapter 7. It seems obvious that use of such elements in the analysis of curved shells could be made directly simply by reducing their dimension in the thickness direction as shown in Fig. 14.1. Indeed, in an axisymmetric situation such an application is illustrated in the example of Fig. 7.17. With a straightforward use of the three-dimensional concept, however, certain difficulties will be encountered.

In the first place the retention of three displacement degrees of freedom at each node leads to large stiffness coefficients from strains in the shell thickness direction. This often presents numerical problems and may lead to ill-conditioned equations when the shell thickness becomes small compared with other dimensions of the element.

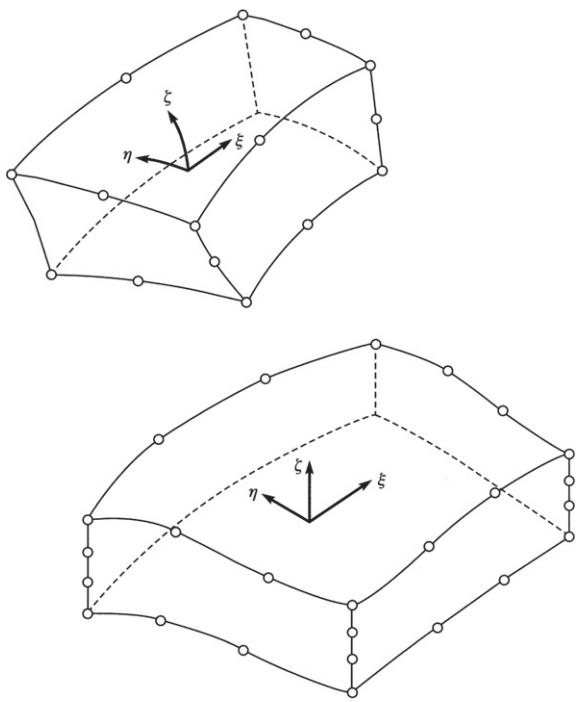



**FIGURE 14.1**
Curved, isoparametric hexahedra in a direct approximation to a curved shell.

The second factor is that of economy. The use of several nodes across the shell thickness ignores the well-known fact that even for thick shells the "normals" to the mid-surface remain practically straight after deformation. Thus an unnecessarily high number of degrees of freedom has to be carried, involving penalties on compute time.

In this chapter we present specialized formulations which overcome both of these difficulties. The constraint of straight "normals" is introduced to improve economy and the strain energy corresponding to the stress perpendicular to the mid-surface is ignored to improve numerical conditioning [19,38,39]. With these modifications an efficient tool for analyzing curved thick shells becomes available. The accuracy and wide range of applicability of the approach is demonstrated in several examples.

## 14.2 Shell element with displacement and rotation parameters

The reader will note that the two constraints introduced correspond precisely to the so-called Reissner-Mindlin assumptions already discussed in Chapter 13 to describe the behavior of thick plates. The omission of the third constraint associated with the thin plate theory (normals remaining normal to the mid-surface after deformation) permits the shell to experience transverse shear deformations—an important feature of thick shell situations.

The formulation presented here leads to additional complications compared with the straightforward use of a three-dimensional element. The elements developed here are in essence an alternative to the processes discussed in Chapter 13, for which an independent interpolation of slopes and displacement is used. The use of reduced integration again is useful if thin shells are to be dealt with—and, indeed, it was in this context that this procedure was first discovered [40–43]. Again the same restrictions for robust behavior as those discussed in Chapter 13 become applicable and generally elements that perform well in plate situations will do well in shells.

### 14.2.1 Geometric definition of an element

Consider a typical shell element illustrated in Fig. 14.2. The external faces of the element are curved, while the sections across the thickness are generated by straight lines. Pairs of nodal points, $\tilde{\mathbf{x}}_{a_{\text{top}}}$ and $\tilde{\mathbf{x}}_{a_{\text{bottom}}}$, each with given Cartesian coordinates, prescribe the shape of the element.

Let $\xi, \eta$ be the two curvilinear coordinates in the mid-surface of the shell and let $\zeta$ be a linear coordinate in the thickness direction. If, further, we assume that $\xi$, $\eta$, $\zeta$ vary between $-1$ and 1 on the respective faces of the element we can write a relationship between the Cartesian coordinates of any point of the shell and the curvilinear coordinates in the form

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \sum N_a(\xi, \eta) \left( \frac{1+\zeta}{2} \begin{Bmatrix} \tilde{x}_a \\ \tilde{y}_a \\ \tilde{z}_a \end{Bmatrix}_{\text{top}} + \frac{1-\zeta}{2} \begin{Bmatrix} \tilde{x}_a \\ \tilde{y}_a \\ \tilde{z}_a \end{Bmatrix}_{\text{bottom}} \right) \tag{14.1}$$

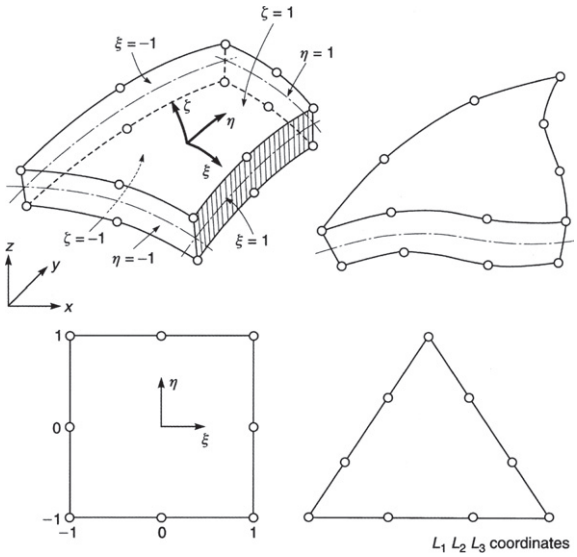



**FIGURE 14.2**
Curved thick shell elements of various types.

Here $N_a(\xi, \eta)$ is a standard two-dimensional shape function taking a value of unity at the top and bottom nodes $a$ and zero at all other nodes. If the basic functions $N_a$ are derived as "shape functions" of a "parent", two-dimensional element, square or triangular[1] in plan, and are so "designed" that compatibility is achieved at interfaces, then the curved space elements will fit into each other. Arbitrary curved shapes of the element can be achieved by using shape functions of higher order than linear. Indeed, any of the two-dimensional shape functions of Chapter 6 can be used here.

The relation between the Cartesian and curvilinear coordinates is now established and it will be found desirable to operate with the curvilinear coordinates as the basis. It should be noted that often the coordinate direction $\zeta$ is *only approximately normal* to the mid-surface.

It is convenient to rewrite the relationship, Eq. (14.1), in a form specified by the "vector" connecting the upper and lower points (i.e., a vector of length equal to the shell thickness $t$) and the mid-surface coordinates. Thus we can rewrite Eq. (14.1) as (Fig. 14.3)

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \sum N_a(\xi,\ \eta) \left( \begin{Bmatrix} \tilde{x}_a \\ \tilde{y}_a \\ \tilde{z}_a \end{Bmatrix} + \frac{1}{2}\zeta\, \mathbf{V}_{3a} \right) \tag{14.2}$$

[1] Area coordinates $L_k$ would be used in this case in place of $\xi, \eta$.

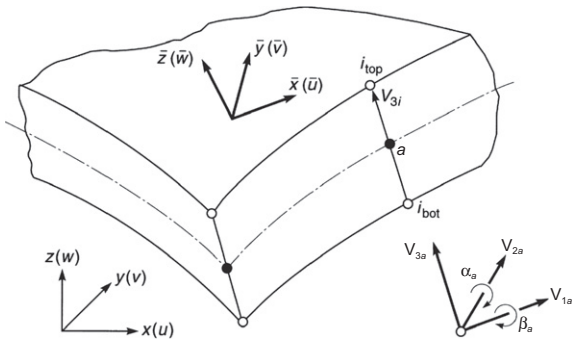



**FIGURE 14.3**
Local and global coordinates.

where

$$\begin{Bmatrix} \tilde{x}_a \\ \tilde{y}_a \\ \tilde{z}_a \end{Bmatrix} = \frac{1}{2}\left( \begin{Bmatrix} \tilde{x}_a \\ \tilde{y}_a \\ \tilde{z}_a \end{Bmatrix}_{\text{top}} + \begin{Bmatrix} \tilde{x}_a \\ \tilde{y}_a \\ \tilde{z}_a \end{Bmatrix}_{\text{bottom}} \right) \quad \text{and} \quad \mathbf{V}_{3a} = \begin{Bmatrix} \tilde{x}_a \\ \tilde{y}_a \\ \tilde{z}_a \end{Bmatrix}_{\text{top}} - \begin{Bmatrix} \tilde{x}_a \\ \tilde{y}_a \\ \tilde{z}_a \end{Bmatrix}_{\text{bottom}} \tag{14.3}$$

with $\mathbf{x}_a$ now defining the *mid-surface* nodal coordinates and $\mathbf{V}_{3a}$ a vector whose length represents the *shell director* and has length of shell thickness.

For relatively thin shells, it is convenient to replace the vector $\mathbf{V}_{3a}$ by a unit vector $\mathbf{v}_{3a}$ in the direction normal to the mid-surface. Now Eq. (14.2) is written simply as

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \sum_a N_a(\xi, \eta) \left( \begin{Bmatrix} \tilde{x}_a \\ \tilde{y}_a \\ \tilde{z}_a \end{Bmatrix} + \frac{1}{2}\zeta t_a \mathbf{v}_{3a} \right)$$

where $t_a$ is the shell thickness at the node $a$. Construction of a vector normal to the mid-surface is a simple process (see Eq. 7.26).

### 14.2.2 Displacement field

The displacement field is now specified for the element. As the strains in the direction normal to the mid-surface will be assumed to be negligible, the displacement throughout the element will be taken to be uniquely defined by the *three Cartesian components* of the mid-surface node displacement and *two rotations* about two orthogonal directions normal to the nodal vector $\mathbf{V}_{3a}$ (or alternatively, $\mathbf{v}_{3a}$). If these two orthogonal directions are denoted by unit vectors $\mathbf{v}_{1a}$ and $\mathbf{v}_{2a}$ with corresponding rotations $\tilde{\alpha}_a$ and $\tilde{\beta}_a$ (see Fig. 14.3), we can write, similar to Eq. (14.2),

$$\begin{Bmatrix} u \\ v \\ w \end{Bmatrix} = \sum_a N_a(\xi, \eta) \left( \begin{Bmatrix} \tilde{u}_a \\ \tilde{v}_a \\ \tilde{w}_a \end{Bmatrix} + \frac{1}{2}\zeta t_a [\, \mathbf{v}_{1a} \quad -\mathbf{v}_{2a} \,] \begin{Bmatrix} \tilde{\alpha}_a \\ \tilde{\beta}_a \end{Bmatrix} \right) \tag{14.4}$$

from which the usual form is readily obtained as

$$\begin{Bmatrix} u \\ v \\ w \end{Bmatrix} = \mathbf{N}\tilde{\mathbf{u}}^e, \quad \tilde{\mathbf{u}}^e = \begin{Bmatrix} \tilde{\mathbf{u}}_1^e \\ \vdots \\ \tilde{\mathbf{u}}_m^e \end{Bmatrix} \quad \text{with} \quad \tilde{\mathbf{u}}_a^e = \begin{Bmatrix} \tilde{u}_a \\ \tilde{v}_a \\ \tilde{w}_a \\ \tilde{\alpha}_a \\ \tilde{\beta}_a \end{Bmatrix} \tag{14.5}$$

where $u$, $v$, and $w$ are displacements in the directions of the global $x$, $y$, and $z$ axes, and

$$\mathbf{N}_a(\xi, \eta, \zeta) = N_a(\xi, \eta) \, [\mathbf{I} \quad \tfrac{1}{2}\zeta t_a \mathbf{v}_{1a} \quad -\tfrac{1}{2}\zeta t_a \mathbf{v}_{2a}]$$

As an infinity of vector directions normal to a given direction can be generated, a particular scheme has to be devised to ensure a *unique* definition. One unique method is given here but other possibilities are open.

Let $\mathbf{V}_{3a}$ be the vector to which a normal direction is to be constructed. A coordinate vector in a Cartesian system may be defined by

$$\mathbf{x} = x\mathbf{e}_x + y\mathbf{e}_y + z\mathbf{e}_z \tag{14.6}$$

in which $\mathbf{e}_x$, $\mathbf{e}_y$, and $\mathbf{e}_z$ are three (orthogonal) base vectors. To construct the first normal vector we find the minimum component of $\mathbf{V}_{3a}$ and compute the vector cross-product with the unit vector in this direction to define a vector $\mathbf{V}_{1a}$. For example, if the $x$ component of $\mathbf{V}_{3a}$ is the smallest one we construct

$$\mathbf{V}_{1a} = \mathbf{e}_x \times \mathbf{V}_{3a} \tag{14.7}$$

where

$$\mathbf{e}_x = [\,1 \quad 0 \quad 0\,]^T$$

is the form of the unit vector in the $x$ direction. Now

$$\mathbf{v}_{1a} = \frac{\mathbf{V}_{1a}}{|\mathbf{V}_{1a}|} \quad \text{where} \quad |\mathbf{V}_{1a}| = \sqrt{\mathbf{V}_{1a}^T \mathbf{V}_{1a}} \tag{14.8}$$

defines the first unit vector.

The second normal vector may now be computed from

$$\mathbf{V}_{2a} = \mathbf{V}_{3a} \times \mathbf{V}_{1a} \tag{14.9}$$

and normalized using the form in Eq. (14.8). We have thus three local, orthogonal axes defined by unit vectors

$$\mathbf{v}_{1a}, \mathbf{v}_{2a}, \text{ and } \mathbf{v}_{3a} \tag{14.10}$$

Once again if $N_a$ are $C_0$ functions then using (14.4) displacement compatibility is maintained between adjacent elements.

The element coordinate definition is now given by the Eq. (14.2) and has more degrees of freedom than the definition of the displacements. The element is therefore

of the "superparametric" kind and the constant strain criteria are not automatically satisfied. Nevertheless, it will be seen from the definition of strain components involved that both rigid body motions and constant strain conditions are available.

Physically it has been assumed in the definition of Eq. (14.4) that no strains occur in the "thickness" direction $\zeta$. While this direction is not always exactly normal to the mid-surface it still represents a good approximation of one of the usual shell assumptions.

At each mid-surface node $a$ of Fig. 14.3 we now have the five basic degrees of freedom, and the connection of elements will follow precisely the standard pattern for element assembly.

### 14.2.3 Definition of strains and stresses

To derive the properties of a finite element the essential strains and stresses need first to be defined. The components in directions of *orthogonal axes* related to the surface $\zeta$ (constant) are essential if account is to be taken of the basic shell assumptions. Thus, if at any point in this surface we erect a normal $\bar{z}$ with two other orthogonal axes $\bar{x}$ and $\bar{y}$ tangential to it (Fig. 14.3), the strain components of interest are given simply by the three-dimensional relationships in Chapter 2:

$$\bar{\boldsymbol{\varepsilon}} = \begin{Bmatrix} \varepsilon_{\bar{x}} \\ \varepsilon_{\bar{y}} \\ \gamma_{\bar{x}\bar{y}} \\ \gamma_{\bar{y}\bar{z}} \\ \gamma_{\bar{z}\bar{x}} \end{Bmatrix} = \begin{Bmatrix} \bar{u}_{,\bar{x}} \\ \bar{v}_{,\bar{y}} \\ \bar{u}_{,\bar{y}} + \bar{v}_{,\bar{x}} \\ \bar{v}_{,\bar{z}} + \bar{w}_{,\bar{y}} \\ \bar{w}_{,\bar{x}} + \bar{u}_{,\bar{z}} \end{Bmatrix} \tag{14.11}$$

with the strain in direction $\bar{z}$ neglected so as to be consistent with the usual shell assumptions. It must be noted that in general none of these directions coincide with those of the curvilinear coordinates $\xi, \eta, \zeta$, although $\bar{x}, \bar{y}$ are in the $\xi\eta$ plane ($\zeta$ = constant).[2]

The stresses corresponding to these strains are defined by a matrix $\bar{\boldsymbol{\sigma}}$ and for elastic behavior are related to the usual elasticity matrix $\bar{\mathbf{D}}$. Thus

$$\bar{\boldsymbol{\sigma}} = \begin{Bmatrix} \sigma_{\bar{x}} \\ \sigma_{\bar{y}} \\ \tau_{\bar{x}\bar{y}} \\ \tau_{\bar{y}\bar{z}} \\ \tau_{\bar{z}\bar{x}} \end{Bmatrix} = \bar{\mathbf{D}}\left(\bar{\boldsymbol{\varepsilon}} - \bar{\boldsymbol{\varepsilon}}_0\right) + \bar{\boldsymbol{\sigma}}_0 \tag{14.12}$$

where $\bar{\boldsymbol{\varepsilon}}_0$ and $\bar{\boldsymbol{\sigma}}_0$ represent any "initial" strains and stresses, respectively.

The $5 \times 5$ matrix $\bar{\mathbf{D}}$ can now include any anisotropic properties and indeed may be prescribed as a function of $\zeta$ if sandwich or laminated construction is used.

[2]Indeed, these directions will only approximately agree with the nodal directions $\mathbf{v}_{1a}$, $\mathbf{v}_{2a}$ previously derived, as in general the vector $\mathbf{v}_{3a}$ is only approximately normal to the mid-surface.

For the present we shall define it only for an isotropic material. Here

$$\bar{\mathbf{D}} = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & \nu & 0 & 0 & 0 \\ \nu & 1 & 0 & 0 & 0 \\ 0 & 0 & (1-\nu)/2 & 0 & 0 \\ 0 & 0 & 0 & \kappa(1-\nu)/2 & 0 \\ 0 & 0 & 0 & 0 & \kappa(1-\nu)/2 \end{bmatrix} \tag{14.13}$$

in which $E$ and $\nu$ are Young's modulus and Poisson's ratio, respectively. The factor $\kappa$ included in the last two shear terms is taken as $5/6$ and its purpose is to improve the shear displacement approximations (see Chapter 13). From the displacement definition it will be seen that the shear distribution is approximately constant through the thickness, whereas in reality the shear distribution for elastic behavior is approximately parabolic. The value $\kappa = 5/6$ is the ratio of relevant strain energies.

It is important to note that this matrix is *not* defined by deleting appropriate terms from the equivalent three-dimensional stress matrix. It must be derived by substituting $\sigma_{\bar{z}} = 0$ into the three-dimensional constitutive equations and performing suitable elimination so that this important shell assumption is satisfied. This is similar to the procedure for deriving plane stress behavior in two-dimensional analyses.

### 14.2.4 Element properties and necessary transformations

The stiffness matrix—and indeed all other "element" property matrices—involves integrals over the volume of the element, which are quite generally of the form

$$\int_{\Omega^e} \mathbf{H}\, \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z \tag{14.14}$$

where the matrix $\mathbf{H}$ is a function of the coordinates. For instance, in the stiffness matrix

$$\mathbf{H} = \bar{\mathbf{B}}^T \bar{\mathbf{D}} \bar{\mathbf{B}} \tag{14.15}$$

and with the usual definition

$$\bar{\boldsymbol{\varepsilon}} = \bar{\mathbf{B}} \tilde{\mathbf{u}}^e \tag{14.16}$$

we have $\bar{\mathbf{B}}$ defined in terms of the displacement derivatives with respect to the local Cartesian coordinates $\bar{x}, \bar{y}, \bar{z}$ by Eq. (14.11). Now, therefore, *two sets of transformations* are necessary before the element can be integrated with respect to the curvilinear coordinates $\xi, \eta, \zeta$.

First, by identically the same process as that used for isoparametric elements, the derivatives with respect to the $x, y, z$ directions are obtained. As Eq. (14.4) relates the

global displacements $u, v, w$ to the curvilinear coordinates, the derivatives of these displacements with respect to the global $x, y, z$ coordinates are given by a matrix relation:

$$\begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial v}{\partial x} & \frac{\partial w}{\partial x} \\ \frac{\partial u}{\partial y} & \frac{\partial v}{\partial y} & \frac{\partial w}{\partial y} \\ \frac{\partial u}{\partial z} & \frac{\partial v}{\partial z} & \frac{\partial w}{\partial z} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \frac{\partial u}{\partial \xi} & \frac{\partial v}{\partial \xi} & \frac{\partial w}{\partial \xi} \\ \frac{\partial u}{\partial \eta} & \frac{\partial v}{\partial \eta} & \frac{\partial w}{\partial \eta} \\ \frac{\partial u}{\partial \zeta} & \frac{\partial v}{\partial \zeta} & \frac{\partial w}{\partial \zeta} \end{bmatrix} \tag{14.17}$$

In this, the Jacobian matrix is defined as

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \\ \frac{\partial x}{\partial \zeta} & \frac{\partial y}{\partial \zeta} & \frac{\partial z}{\partial \zeta} \end{bmatrix} \tag{14.18}$$

and calculated from the coordinate definitions of Eq. (14.2). Now, for every set of curvilinear coordinates the global displacement derivatives can be obtained numerically.

A second transformation to the local displacements $\bar{x}, \bar{y}, \bar{z}$ will allow the strains, and hence the $\bar{\mathbf{B}}$ matrix, to be evaluated. The directions of the local axes can be established from a vector normal to the $\xi\eta$ mid-surface ($\zeta = 0$). This vector can be found from two vectors $\partial\mathbf{x}/\partial\xi$ and $\partial\mathbf{x}/\partial\eta$ that are tangential to the mid-surface. Thus

$$\mathbf{V}_3 = \begin{bmatrix} \frac{\partial x}{\partial \xi} \\ \frac{\partial y}{\partial \xi} \\ \frac{\partial z}{\partial \xi} \end{bmatrix} \times \begin{bmatrix} \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \eta} \\ \frac{\partial z}{\partial \eta} \end{bmatrix} = \begin{bmatrix} \frac{\partial y}{\partial \xi}\frac{\partial z}{\partial \eta} - \frac{\partial y}{\partial \eta}\frac{\partial z}{\partial \xi} \\ \frac{\partial z}{\partial \xi}\frac{\partial x}{\partial \eta} - \frac{\partial z}{\partial \eta}\frac{\partial x}{\partial \xi} \\ \frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta}\frac{\partial y}{\partial \xi} \end{bmatrix} \tag{14.19}$$

We can now construct two perpendicular vectors $\mathbf{V}_1$ and $\mathbf{V}_2$ following the process given previously to describe the $\bar{x}$ and $\bar{y}$ directions, respectively. The three orthogonal vectors can be reduced to unit magnitudes to obtain a matrix of vectors in the $\bar{x}, \bar{y}, \bar{z}$ directions (which is in fact the direction cosine matrix) given as

$$\boldsymbol{\theta} = [\,\mathbf{v}_1, \quad \mathbf{v}_2, \quad \mathbf{v}_3\,] \tag{14.20}$$

The global derivatives of displacement $u$, $v$, and $w$ are now transformed to the local derivatives of the local orthogonal displacements by a standard operation

$$\begin{bmatrix} \frac{\partial \bar{u}}{\partial \bar{x}} & \frac{\partial \bar{v}}{\partial \bar{x}} & \frac{\partial \bar{w}}{\partial \bar{x}} \\ \frac{\partial \bar{u}}{\partial \bar{y}} & \frac{\partial \bar{v}}{\partial \bar{y}} & \frac{\partial \bar{w}}{\partial \bar{y}} \\ \frac{\partial \bar{u}}{\partial \bar{z}} & \frac{\partial \bar{v}}{\partial \bar{z}} & \frac{\partial \bar{w}}{\partial \bar{z}} \end{bmatrix} = \boldsymbol{\theta}^T \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial v}{\partial x} & \frac{\partial w}{\partial x} \\ \frac{\partial u}{\partial y} & \frac{\partial v}{\partial y} & \frac{\partial w}{\partial y} \\ \frac{\partial u}{\partial z} & \frac{\partial v}{\partial z} & \frac{\partial w}{\partial z} \end{bmatrix} \boldsymbol{\theta} \tag{14.21}$$

From this the components of the $\bar{\mathbf{B}}$ matrix can now be found explicitly, noting that five degrees of freedom exist at each node:

$$\bar{\boldsymbol{\varepsilon}} = \bar{\mathbf{B}}\tilde{\mathbf{u}}^e \tag{14.22}$$

where the form of $\tilde{\mathbf{u}}^e$ is given in Eq. (14.5).

The infinitesimal volume is given in terms of the curvilinear coordinates as

$$\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = j\,\mathrm{d}\xi\,\mathrm{d}\eta\,\mathrm{d}\zeta \tag{14.23}$$

where $j = \det\ \mathbf{J}$. This standard expression completes the basic formulation.

Numerical integration within the appropriate limits is carried out in exactly the same way as for three-dimensional elements using Gaussian quadrature formulae. An identical process serves to define all the other relevant element matrices arising from body and surface loading, inertia matrices, etc.

As the variation of the strain quantities in the thickness, or $\zeta$ direction, is linear, two Gauss points in that direction are sufficient for homogeneous elastic sections, while two to four points are needed in the $\xi, \eta$ directions for parabolic and cubic shape functions $N_a(\xi, \eta)$.

### 14.2.5 Some remarks on stress representation

The element properties are now defined, and the assembly and solution are in standard form. It remains to discuss the presentation of the stresses, and this problem is of some consequence. The strains being defined in local direction, $\bar{\boldsymbol{\sigma}}$, are readily available. Such components are indeed directly of interest but as the directions of local axes are not easily visualized (and indeed may not be continuously defined between adjacent elements) it is sometimes convenient to transfer the components to the global system using the standard transformation

$$\begin{bmatrix} \sigma_x & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_y & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_z \end{bmatrix} = \boldsymbol{\theta} \begin{bmatrix} \sigma_{\bar{x}} & \tau_{\bar{x}\bar{y}} & \tau_{\bar{x}\bar{z}} \\ \tau_{\bar{y}\bar{x}} & \sigma_{\bar{y}} & \tau_{\bar{y}\bar{z}} \\ \tau_{\bar{z}\bar{x}} & \tau_{\bar{z}\bar{y}} & \sigma_{\bar{z}} \end{bmatrix} \boldsymbol{\theta}^T \tag{14.24}$$

This transformation should be performed only for elements which belong to the approximation for the same smooth surface and/or same material.

In a general shell structure, the stresses in a global system do not, however, give a clear picture of shell surface stresses. It is thus convenient always to compute the principal stresses (or invariants of stress) by a suitable transformation. Regarding the shell stresses more rationally, one may note that the shear components $\tau_{\bar{x}\bar{z}}$ and $\tau_{\bar{y}\bar{z}}$ are often zero on the top and bottom surfaces and this may be noted when making the transformation of Eq. (14.24) before converting to global components to ensure in this case that the principal stresses lie on the surface of the shell. The values obtained directly for these shear components are the average values across the section. The maximum transverse shear on a solid cross-section occurs on the mid-surface and, assuming near parabolic variation over the thickness, is equal to about 1.5 times the average value.

## 14.3 Special case of axisymmetric thick shells

For axisymmetric shells the formulation is simplified. Now the element mid-surface is defined by only two coordinates $\xi$, $\eta$ and a considerable saving in computer effort is obtained [38].

The element now is derived in a similar manner by starting from a two-dimensional definition of Fig. 14.4.

Equations (14.1) and (14.2) are now replaced by their two-dimensional equivalents defining the relation between coordinates as

$$\begin{aligned}\begin{Bmatrix} r \\ z \end{Bmatrix} &= \sum_a N_a(\xi) \left( \frac{1+\eta}{2} \begin{Bmatrix} \tilde{r}_a \\ \tilde{z}_a \end{Bmatrix}_{\text{top}} + \frac{1-\eta}{2} \begin{Bmatrix} \tilde{r}_a \\ \tilde{z}_a \end{Bmatrix}_{\text{bottom}} \right) \\ &= \sum_a N_a(\xi) \left( \begin{Bmatrix} \tilde{r}_a \\ \tilde{z}_a \end{Bmatrix} + \frac{1}{2} \eta \, t_a \mathbf{v}_{3a} \right)\end{aligned} \tag{14.25}$$

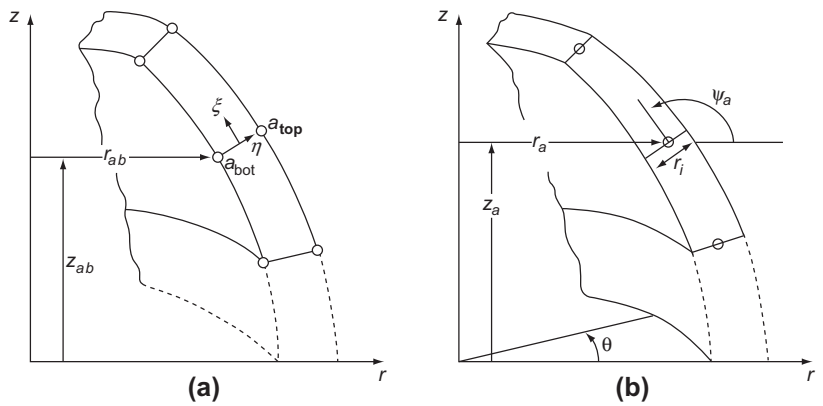



**FIGURE 14.4**
Coordinates for an axisymmetric shell: (a) coordinate representation and (b) shell representation.

with

$$\begin{Bmatrix} \tilde{r}_a \\ \tilde{z}_a \end{Bmatrix} = \frac{1}{2}\left( \begin{Bmatrix} \tilde{r}_a \\ \tilde{z}_a \end{Bmatrix}_{\text{top}} + \begin{Bmatrix} \tilde{r}_a \\ \tilde{z}_a \end{Bmatrix}_{\text{bottom}} \right) \quad \text{and} \quad \mathbf{v}_{3a} = \begin{Bmatrix} \cos\psi_a \\ \sin\psi_a \end{Bmatrix}$$

in which $\psi_a$ is the angle defined in Fig. 14.4(b) and $t_a$ is the shell thickness. Similarly, the displacement definition is specified by following the lines of Eq. (14.4).

Here we consider the case of axisymmetric loading in the $r$ and $z$ directions only (no torsion). Thus, we specify the two displacement components as

$$\begin{Bmatrix} u \\ w \end{Bmatrix} = \sum_a N_a \left( \begin{Bmatrix} \tilde{u}_a \\ \tilde{w}_a \end{Bmatrix} + \frac{\eta t_a}{2} \begin{Bmatrix} -\sin\tilde{\psi}_a \\ \cos\tilde{\psi}_a \end{Bmatrix} \tilde{\phi}_a \right) \tag{14.26}$$

In this $\tilde{\phi}_a$ is the rotation illustrated in Fig. 14.5, and $\tilde{u}_a$, $\tilde{w}_a$ are the displacements of the middle surface node.

Global strains are conveniently defined by the relationship [44]

$$\boldsymbol{\varepsilon} = \begin{Bmatrix} \varepsilon_r \\ \varepsilon_z \\ \varepsilon_\theta \\ \gamma_{rz} \end{Bmatrix} = \begin{Bmatrix} \dfrac{\partial u}{\partial r} \\ \dfrac{\partial w}{\partial z} \\ \dfrac{u}{r} \\ \dfrac{\partial u}{\partial z} + \dfrac{\partial w}{\partial r} \end{Bmatrix} \tag{14.27}$$

These strains are transformed to the local coordinates and the component normal to $\eta$ ($\eta$ = constant) is neglected.

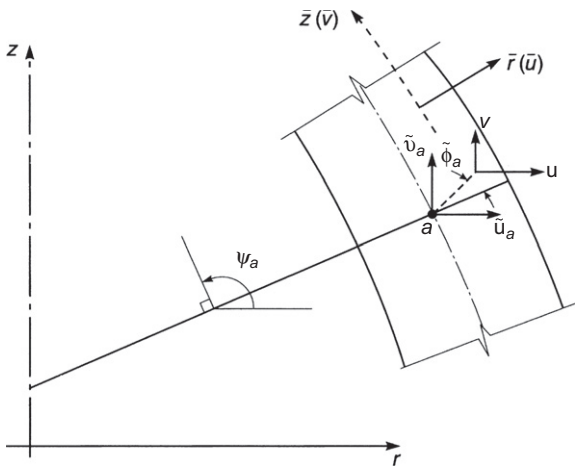



**FIGURE 14.5**
Global displacements in an axisymmetric shell.

All the transformations follow the pattern described in previous sections and need not be further commented on except perhaps to remark that they are now carried out only between sets of directions $\xi$, $\eta$, $r$, $z$, and $\bar{r}$, $\bar{z}$, thus involving only two variables.

Similarly the integration of element properties is carried out numerically with respect to $\xi$ and $\eta$ only, noting, however, that the volume element is

$$\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = 2\pi r \det\mathbf{J}\,\mathrm{d}\xi\,\mathrm{d}\eta\,\mathrm{d}\theta = 2\pi r\, j\,\mathrm{d}\xi\,\mathrm{d}\eta\,\mathrm{d}\theta \tag{14.28}$$

By suitable choice of shape functions $N_a(\xi)$, straight, parabolic, or cubic shapes of variable thickness elements can be used as shown in Fig. 14.6.

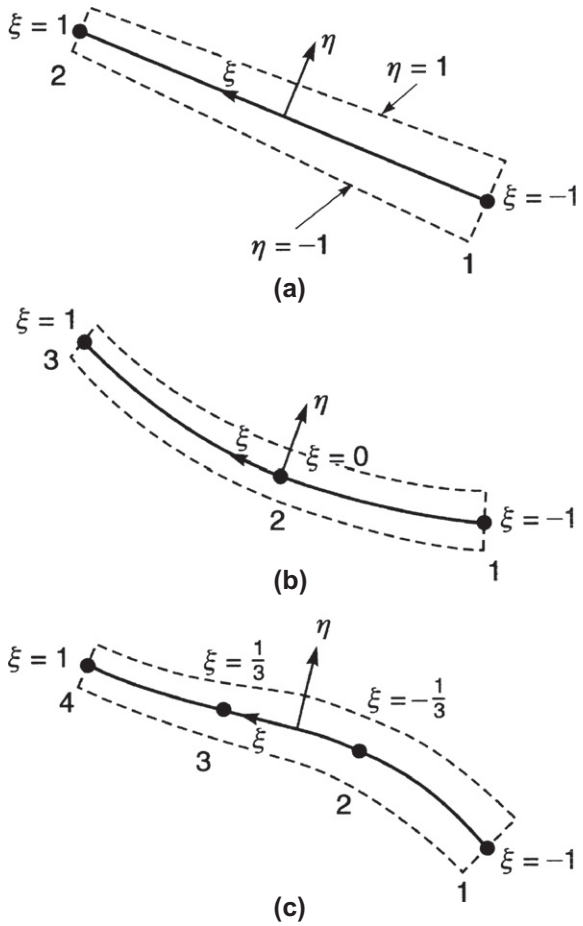



**FIGURE 14.6**
Axisymmetric shell elements: (a) linear, (b) parabolic, and (c) cubic.

## 14.4 Special case of thick plates

The transformations necessary in this chapter are somewhat involved and the programming steps are quite sophisticated. However, the application of the principle involved is available for thick plates and readers are advised to first test their comprehension on such a simple problem.

Here the following obvious simplifications arise.

1. $\zeta = 2z/t$ and unit vectors $\mathbf{v}_1$, $\mathbf{v}_2$, and $\mathbf{v}_3$ can be taken as $\mathbf{e}_x$, $\mathbf{e}_y$, and $\mathbf{e}_z$, respectively.
2. $\tilde{\alpha}_a$ and $\tilde{\beta}_a$ are simply the rotations $\tilde{\theta}_y$ and $\tilde{\theta}_x$, respectively (see Chapter 13).
3. It is no longer necessary to transform stress and strain components to a local system of axes $\bar{x}$, $\bar{y}$, $\bar{z}$ and global definitions $x$, $y$, $z$ can be used throughout. For elements of this type, numerical thickness integration can be avoided and, as an exercise, readers are encouraged to derive the stiffness matrices, etc., for, say, linear, rectangular elements. Forms will be found which are identical to those derived in Chapter 13 with an independent displacement and rotation interpolation and using shear constraints. This demonstrates the essential identity of the alternative procedures.

## 14.5 Convergence

Whereas in three-dimensional analysis it is possible to talk about absolute convergence to the true exact solution of the elasticity problem, in equivalent plate and shell problems such a convergence cannot happen. As the element size decreases the so-called convergent solution of a plate bending problem approaches only to the exact solution of the approximate model implied in the formulation. Thus, here again convergence of the above formulation will only occur to the exact solution constrained by the requirement that straight "normals" remain straight during deformation.

In elements of finite size it will be found that pure bending deformation modes are nearly always accompanied by some shear strains which in fact do not exist in the conventional thin plate or shell bending theory (although quite generally shear stresses may be deduced by equilibrium considerations on an element of the model, similar to the manner by which shear stresses in beams are deduced). Thus large elements deforming mainly under bending action (as would be the case of the shell element degenerated to a flat plate) tend to be appreciably too stiff. In such cases certain limits of the ratio of size of element to its thickness need to be imposed. However, it will be found that such restrictions often are relaxed by the simple expedient of *reducing the integration order* [40].

Figure 14.7 shows, for instance, the application of the quadratic eight-node element to a square plate situation. Here results for integration with $3 \times 3$ and $2 \times 2$ Gauss points are given and results plotted for different thickness-to-span ratios. For reasonably thick situations, the results are similar and both give the additional shear

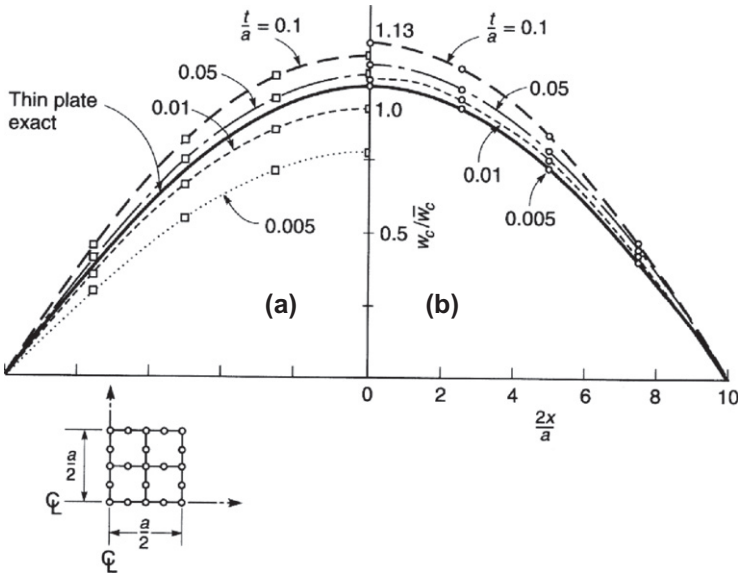



**FIGURE 14.7**

A simply supported square plate under uniform load $q_0$: plot of central deflection $w_c$ for eight-node elements with (a) 3 × 3 Gauss point integration and (b) with 2 × 2 (reduced) Gauss point integration. Central deflection is $\bar{w}_c$ for thin plate theory.

deformation not available by thin plate theory. However, for thin plates the results with the more exact integration tend to diverge rapidly from the now correct thin plate results whereas the reduced integration still gives excellent results. The reasons for this improved performance are discussed in Chapter 13 and the reader is referred there for further plate examples using different types of shape functions.

## 14.6 Some shell examples

A limited number of examples which show the accuracy and range of application of the axisymmetric shell formulation presented in this chapter will be given. For a fuller selection the reader is referred to references [19,38–43].

### 14.6.1 Spherical dome under uniform pressure

An "exact" solution of shell theory is known for this axisymmetric problem, illustrated in Fig. 14.8. Twenty-four cubic serendipity-type elements are used with graded size more closely spaced toward supports. Contrary to the "exact" shell theory solution, the present formulation can distinguish between the application of pressure on the outer (Case I) and inner (Case II) surfaces as shown in the figure.

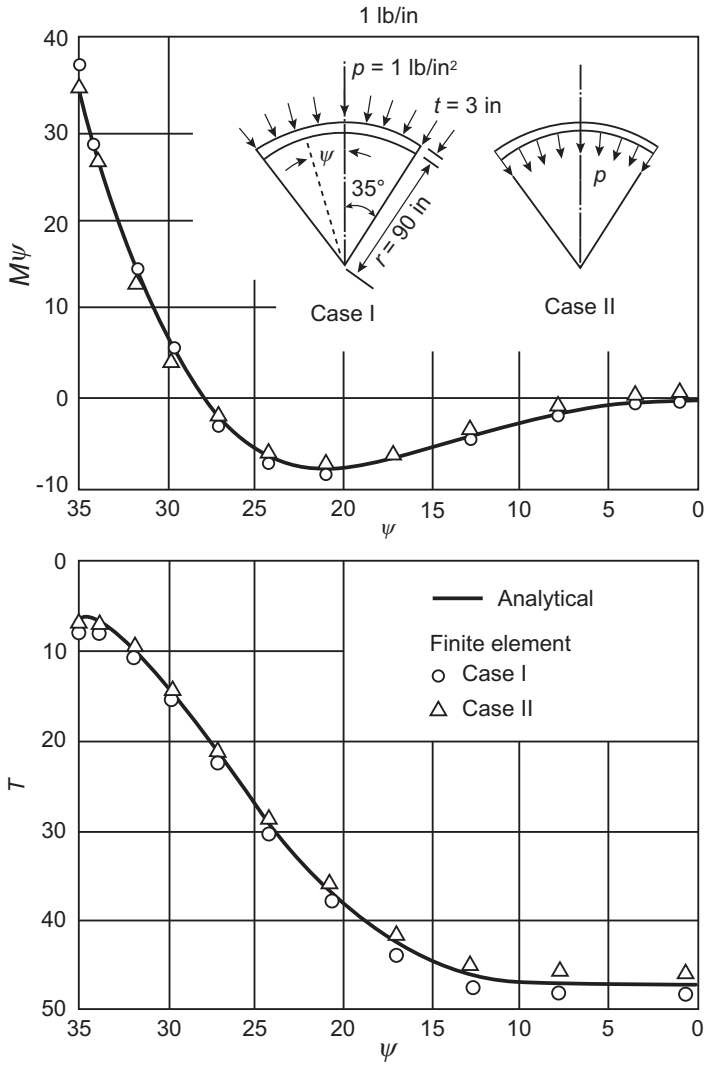



**FIGURE 14.8**

Spherical dome under uniform pressure analyzed with 24 cubic serendipity elements (first element subtends an angle of 0.1° from fixed end, others in arithmetic progression).

### 14.6.2 Edge-loaded cylinder

A further axisymmetric example is shown in Fig. 14.9 to study the effect of subdivision. Two, six, or fourteen cubic serendipity elements of unequal length are used and the results for both of the finer subdivisions are almost coincident with the exact solution. Even the two-element solution gives reasonable results and departs only in the vicinity of the loaded edge.

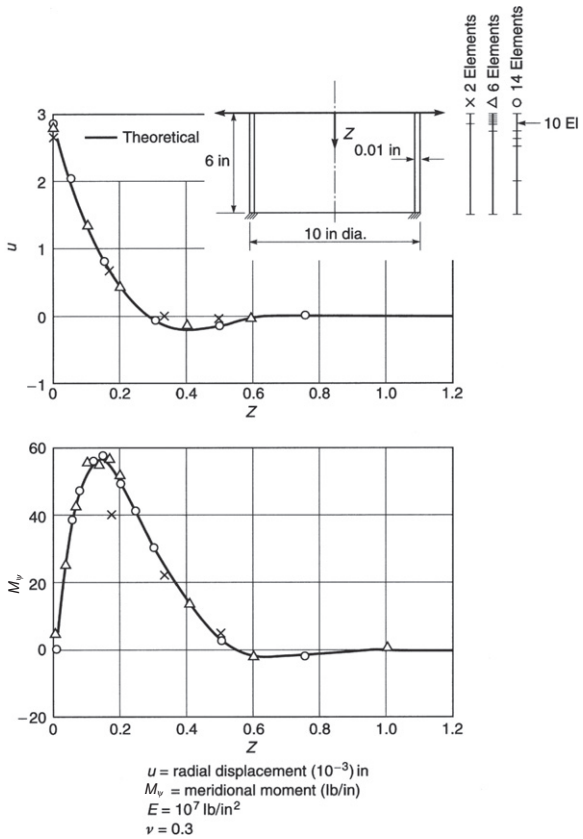


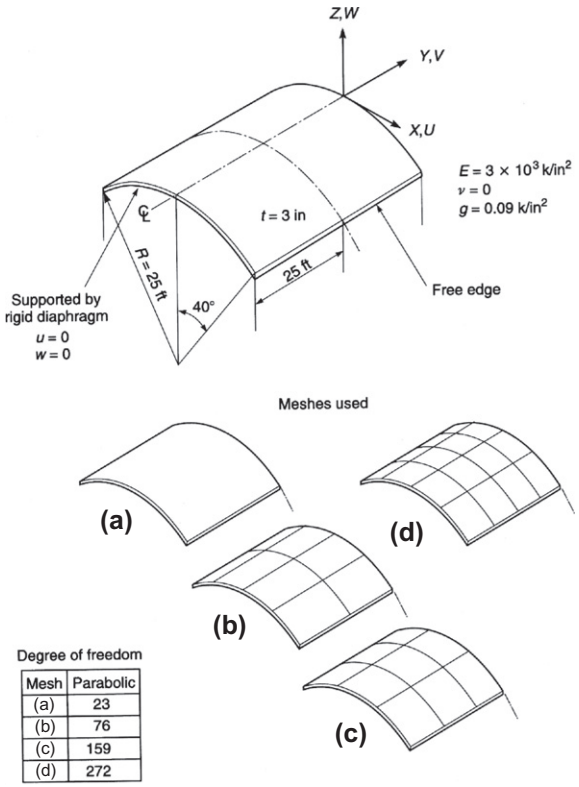

**FIGURE 14.9**
Thin cylinder under a unit radial edge load.

### 14.6.3 Cylindrical shell: Self-weight loading

This is a test example of application of the full process to a shell in which bending action is dominant as a result of supports restraining deflection at the ends.

In Fig. 14.10 the geometry, physical details of the problem, and subdivision are given, and in Fig. 14.11 the comparison of the effects of $3 \times 3$ and $2 \times 2$ integration using eight-node quadratic elements is shown on the displacements calculated. Both integrations result, as expected, in convergence. For the more exact integration, this is rather slow, but, with reduced integration order, very accurate results are obtained, even with one element. The improved convergence of displacements is matched by rapid convergence of stress components.

This example illustrates most dramatically the advantages of the simple expedient of reduced integration and is described more fully in Refs. [40,42]. The comparison




Degree of freedom

| Mesh | Parabolic |
|---|---|
| (a) | 23 |
| (b) | 76 |
| (c) | 159 |
| (d) | 272 |

**FIGURE 14.10**
Cylindrical shell example: self-weight behavior.

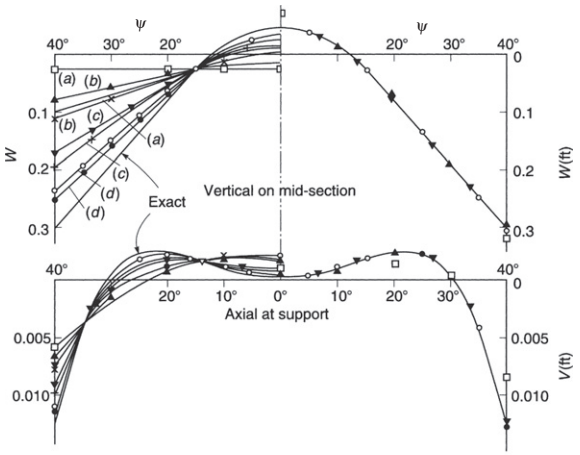
solution for this problem is one derived along more conventional lines by Scordelis and Lo [45].

### 14.6.4 Pipe penetration and spherical cap

The last two examples, a pipe penetration [46] shown in Figs 14.12 and 14.13 and a spherical cap [43] shown in Fig. 14.14, illustrate applications in which the irregular shape of elements is used. Both illustrate practical problems of some interest and show that with reduced integration a useful and very general shell element is available, even when the elements are quite distorted.

## 14.7 Concluding remarks

The elements described in this chapter using degeneration of solid elements are shown in plate problems to be nearly identical to those described in Chapter 13 where an independent slope and displacement interpolation are directly used in the middle




| Original element reference 3 | Integration for shear reduced | Mesh | Simple 2 × 2 integration |
|---|---|---|---|
| □ | ▲ | (a) | □ |
| ▲ | × | (b) | △ |
| ▼ | + | (c) | ▽ |
| • | • | (d) | ○ |

**FIGURE 14.11**
Displacement (parabolic element), cylindrical shell roof.

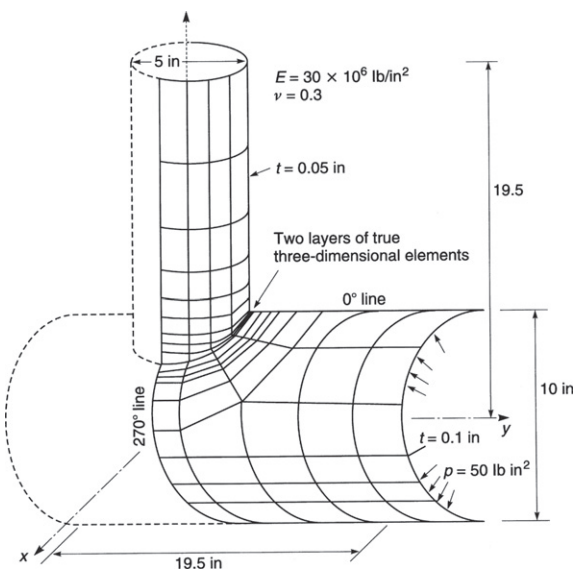



**FIGURE 14.12**
An analysis of cylinder intersection by means of reduced integration shell-type elements.

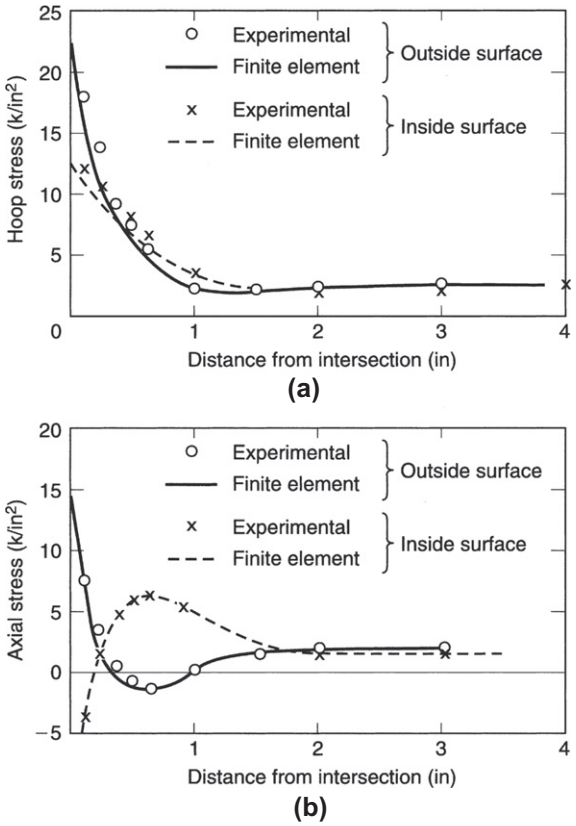



**FIGURE 14.13**
Cylinder-to-cylinder intersections of Fig. 14.12: (a) hoop stresses near 0deg line and (b) axial stresses near 0deg line.

plane. Application to axisymmetric problems is similarly accurate. For the general curved shell the analogy is less obvious but clearly still exists. We should therefore expect the conditions established in Chapter 13 for robustness of plate elements to be still valid. Further, it appears possible that other additional conditions on the various interpolations may have to be imposed in curved element forms. Both statements are true. The 8- and 12-node elements which we have shown in the previous section to perform well will fail under certain circumstances and for this reason many of the more successful plate elements also have been adapted to the shell problem.

The introduction of additional degrees of freedom in the interior of the eight-node serendipity element was first suggested by Cook [47,48] and later by Hughes [49–51] without, however, achieving complete robustness. The full Lagrangian cubic interpolation as shown in Chapter 13 is quite effective and has been shown to perform well. Another option is use of "local shear constraints" on boundaries of elements as suggested by Dvorkin and Bathe [52], Huang and Hinton [53], and Simo et al. [35,36]

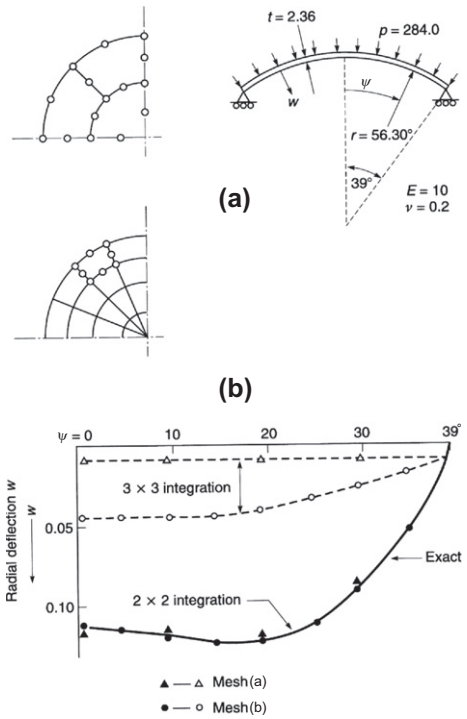



**FIGURE 14.14**
A spherical cap analysis with irregular isoparametric shell elements using full 3 × 3 and reduced 2 × 2 integration: (a) three-element mesh and (b) 16-element mesh.

While the importance of transverse shear strain constraints is now well understood, the constraints introduced by the "in-plane" (membrane) stress resultants are less amenable to analysis (although the elastic parameters $Et$ associated with these are of the same order as those of shear $Gt$). It is well known that *membrane locking* can occur in situations that do not permit inextensional bending. Such locking has been discussed [54–56] but to date the problem has not been rigorously solved. Additional effort to improve the formulation of the processes described in this chapter may be found in Refs. [56–59].

Finally, we note that a comprehensive background and development for shells is presented in companion volume, *The Finite Element Method for Solid and Structural Mechanical* [3]

## 14.8 Problems

**14.1** Use relations from Section 14.3.

**a.** Assume an isoparametric quadratic interpolation for displacements and stresses and develop expressions for the $\mathbf{B}_a$ strain-displacement matrices.

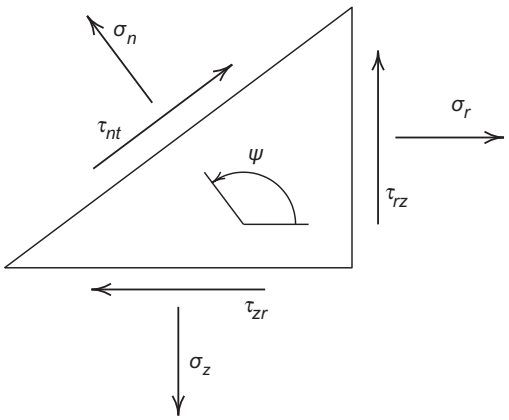



**FIGURE 14.15**
Stress transformation for Problem 14.2.

**b.** Write an irreducible weak form for the problem.

**14.2** For the axisymmetric shell formulation described in Sec. 14.3 use the transformation of stress shown in Fig. 14.15 to define stress resultants $P_{\bar{z}}$, $M_{\bar{z}}$, and $Q_{\bar{r}\bar{z}}$ where $\sigma_{\bar{z}} = \sigma_n$ and $\tau_{\bar{r}\bar{z}} = \tau_{nt}$.

## References

[1] W. Flügge, Stresses in Shells, Springer-Verlag, Berlin, 1960.
[2] S.P. Timoshenko, S. Woinowski-Krieger, Theory of Plates and Shells, second ed., McGraw-Hill, New York, 1959.
[3] O.C. Zienkiewicz, R.L. Taylor, D. Fox, The Finite Element Method for Solid and Structural Mechanics, seventh ed., Elsevier, Oxford, 2013.
[4] B.E. Greene, D.R. Strome, R.C. Weikel, Application of the stiffness method to the analysis of shell structures, in: Proceedings of the Aviation Conference of American Society of Mechanical Engineers ASME, Los Angeles, March 1961.
[5] R.W. Clough and J.L. Tocher, Analysis of thin arch dams by the finite element method, in: Proceedings of International Symposium on Theory of Arch Dams, Pergamon Press, Oxford, 1965.
[6] J.H. Argyris, Matrix displacement analysis of anisotropic shells by triangular elements, J. Roy. Aero. Soc. 69 (1965) 801–805.
[7] R.W. Clough, C.P. Johnson, A finite element approximation for the analysis of thin shells, J. Solids Struct. 4 (1968) 43–60.
[8] O.C. Zienkiewicz, C.J. Parekh, I.P. King, Arch dams analysed by a linear finite element shell solution program, in: Proceedings of International Symposium on Theory of Arch Dams, Pergamon Press, Oxford, 1965.

[9] R.H. Gallagher, Shell elements, in: World Conference on Finite Element Methods in Structural Mechanics, Bournemouth, England, October, 1975.
[10] F.K. Bogner, R.L. Fox, L.A. Schmit, A cylindrical shell element, J. AIAA 5 (1966) 745–750.
[11] J. Connor, C. Brebbia, Stiffness matrix for shallow rectangular shell element, Proc. Am. Soc. Civ. Eng. 93(EM1):43–65, 1967.
[12] A.J. Carr, A refined element analysis of thin shell structures including dynamic loading, Technical Report SEL Report 67–9, University of California, Berkeley, 1967.
[13] S. Utku, Stiffness matrices for thin triangular elements of non-zero Gaussian curvature, J. AIAA 5 (1967) 1659–1667.
[14] G. Cantin, Strain-displacement relationships for cylindrical shells, J. AIAA 6 (1968) 1787–1788.
[15] G. Cantin, R.W. Clough, A curved, cylindrical shell finite element, AIAA J. 6 (6) (1968) 1057–1062.
[16] G. Bonnew, G. Dhatt, Y.M. Giroux, L.P.A. Robichaud, Curved triangular elements for analysis of shells, in: Procedings of the Second Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-68-150, Wright Patterson Air Force Base, Ohio, October 1968.
[17] G.E. Strickland, W.A. Loden, A doubly curved triangular shell element, in: Procedings of the Second Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-68-150, Wright Patterson Air Force Base, Ohio, October 1968.
[18] B.E. Greene, R.E. Jones, D.R. Strome, Dynamic analysis of shells using doubly curved finite elements, in: Procedings of the Second Conference on Matrix Methods in Structural Mechanics, volume AFFDL-TR-68-150, Wright Patterson Air Force Base, Ohio, October 1968.
[19] S. Ahmad, Curved finite elements in the analysis of solids, shells and plate structures. Ph.D thesis, Department of Civil Engineering, University of Wales, Swansea, 1969.
[20] G.R. Cowper, G.M. Lindberg, M.D. Olson, A shallow shell finite element of triangular shape, Int. J. Solids Struct. 6 (1970) 1133–1156.
[21] G. Dupuis, J.J. Goël, A curved finite element for thin elastic shells, Int. J. Solids Struct. 6 (1970) 987–996.
[22] S.W. Key, Z.E. Beisinger, The analysis of thin shells by the finite element method, in: High Speed Computing of Elastic Structures, University of Liége Press, vol. 1, pp. 209–252. 1971.
[23] D.J. Dawe, The analysis of thin shells using a facet element. Technical Report CEGB RD/B/N2038, Berkeley Nuclear Laboratory, England, 1971.
[24] D.J. Dawe, Rigid-body motions and strain-displacement equations of curved shell finite elements, Int. J. Mech. Sci. 14 (1972) 569–578.
[25] D.G. Ashwell, A. Sabir, A new cylindrical shell finite element based on simple independent strain functions, Int. J. Mech. Sci. 4 (1973) 37–47.
[26] G.R. Thomas, R.H. Gallagher, A triangular thin shell finite element: linear analysis, Technical Report CR-2582, NASA, 1975.

[27] B.M. Irons, The semi-Loof shell element, in: DG Ashwell, RH Gallagher (Eds.), Finite Elements for Thin Shells and Curved Members, John Wiley & Sons, Chichester, 1976, pp. 197–222, Chapter 11.
[28] D.G. Ashwell, Strain elements with application to arches, rings, and cylindrical shells, in: D.G. Ashwell, RH Gallagher (Eds.), Finite Elements for Thin Shells and Curved Members, Chapter 6, John Wiley & Sons, Chichester, 1976, pp. 91–111.
[29] N. Carpenter, H. Stolarski, T. Belytschko, A flat triangular shell element with improved membrane interpolation, Commun. Appl. Num. Methods 1 (1985) 161–168.
[30] C. Pratt, Shell finite element via Reissner's principle, Int. J. Solids Struct. 5 (1969) 1119–1133.
[31] J. Connor, G. Will, A mixed finite element shallow shell formulation, in: R.H. Gallagher et al., (Eds.), Advances in Matrix Methods of Structural Analysis and Design, University of Alabama Press, 1969, pp. 105–137.
[32] L.R. Herrmann, W.E. Mason, Mixed formulations for finite element shell analysis, in: Conference on Computer-Oriented Analysis of Shell Structures, Paper AFFDL-TR-71-79, June 1971.
[33] G. Edwards, J.J. Webster, Hybrid cylindrical shell elements, in: DG Ashwell, RH Gallagher (Eds.), Finite Elements for Thin Shells and Curved Members, John Wiley & Sons, Chichester, 1976, pp. 171–195, Chapter 10.
[34] M. Bischoff, W.A. Wall, K.-U. Bletzinger, E. Ramm, Models and finite elements for thin-walled structures, in: E. Stein, R. DeBorst, T.J.R. Hughes (Eds.), Encyclopedia of Computational Mechanics, John Wiley & Sons, Ltd., Chichester, 2004, pp. 59–137.
[35] J.C. Simo, D.D. Fox, On a stress resultant geometrically exact shell model. Part I: Formulation and optimal parametrization, Comput. Methods Appl. Mech. Eng. 72 (1989) 267–304.
[36] J.C. Simo, D.D. Fox, M.S. Rifai, On a stress resultant geometrically exact shell model. Part II: The linear theory; computational aspects, Comput. Methods Appl. Mech. Eng. 73 (1989) 53–92.
[37] M. Bischoff, E. Ramm, Solid-like shell or shell-like solid formulation? A personal view, in: W. Wunderlich (Ed.), Proc. Eur Conf on Comp. Mech (ECCM'99 on CD-ROM), Munich, September 1999.
[38] S. Ahmad, B.M. Irons, O.C. Zienkiewicz, Curved thick shell and membrane elements with particular reference to axi-symmetric problems, in: Proc. 2nd Conf. Matrix Methods in Structural Mechanics, Wright Patterson Air Force Base, Ohio, vol. AFFDL-TR-68-150, October 1968, pp. 539–572.
[39] S. Ahmad, B.M. Irons, O.C. Zienkiewicz, Analysis of thick and thin shell structures by curved finite elements, Int. J. Num. Methods Eng. 2 (1970) 419–451.
[40] O.C. Zienkiewicz, J. Too, R.L. Taylor, Reduced integration technique in general analysis of plates and shells, Int. J. Num. Methods Eng. 3 (1971) 275–290.
[41] S.F. Pawsey, R.W. Clough, Improved numerical integration of thick slab finite elements, Int. J. Num. Methods Eng. 3 (1971) 575–586.

[42] S.F. Pawsey, The analysis of moderately thick to thin shells by the finite element method, Department of Civil Engineering, University of California, Berkeley, Ph.D dissertation (also SESM Report 70–12), 1970.
[43] J.J.M. Too, Two dimensional, plate, shell and finite prism isoparametric elements and their application, Ph.D thesis, Department of Civil Engineering, University of Wales, Swansea, 1970.
[44] I.S. Sokolnikoff, The Mathematical Theory of Elasticity, second ed., McGraw-Hill, New York, 1956.
[45] A.C. Scordelis, K.S. Lo, Computer analysis of cylindrical shells, J. Am. Concr. Inst. 61 (1964) 539–561.
[46] S.A. Bakhrebah, W.C. Schnobrich, Finite element analysis of intersecting cylinders, Technical Report UILU-ENG-73-2018, University of Illinois, Civil Engineering Studies, 1973.
[47] R.D. Cook, More on reduced integration and isoparametric elements, Int. J. Num. Methods Eng. 5 (1972) 141–142.
[48] R.D. Cook, Concepts and Applications of Finite Element Analysis, John Wiley & Sons, Chichester, 1982.
[49] T.J.R. Hughes, M. Cohen, The heterosis finite element for plate bending, Comput. Struct. 9 (1978) 445–450.
[50] T.J.R. Hughes, W.K. Liu, Non linear finite element analysis of shells. Part I, Comput. Methods Appl. Mech. Eng. 26 (1981) 331–362.
[51] T.J.R. Hughes, W.K. Liu, Non linear finite element analysis of shells, Part II, Comput. Methods Appl. Mech. Eng. 27 (1981) 331–362.
[52] E.N. Dvorkin, K.-J. Bathe, A continuum mechanics based four node shell element for general non-linear analysis, Engineering Computations 1 (1984) 77–88.
[53] E.C. Huang, E. Hinton, Elastic, plastic and geometrically non-linear analysis of plates and shells using a new, nine-noded element, in: P. Bergan et al., (Ed.), Finite elements for Non Linear Problems, Springer-Verlag, Berlin, 1986, pp. 283–297.
[54] H. Stolarski, T. Belytschko, Membrane locking and reduced integration for curved elements, J. Appl. Mech. 49 (1982) 172–176.
[55] H. Stolarski, T. Belytschko, Shear and membrane locking in curved $C^0$ elements, Comput. Methods Appl. Mech. Eng. 41 (1983) 279–296.
[56] R.V. Milford, W.C. Schnobrich, Degenerated isoparametric finite elements using explicit integration, Int. J. Num. Methods Eng. 23 (1986) 133–154.
[57] D. Bushnell, Computerized analysis of shells – governing equations, Computers and Structures 18 (1984) 471–536.
[58] M.A. Cilia, N.G. Gray, Improved coordinate transformations for finite elements. the Lagrange cubic case, Int. J. Num. Methods Eng., 23 (1986) 1529–1545.
[59] S. Vlachoutsis, Explicit integration for three dimensional degenerated shell finite elements, Int. J. Num. Methods Eng. 29 (1990) 861–880.

This page is intentionally left blank

CHAPTER

# 15 Errors, Recovery Processes, and Error Estimates

## 15.1 Definition of errors

We have stressed from the beginning of this book the approximate nature of the finite element method and on many occasions we have compared it with known exact solutions. Also, in reference to the "accuracy" of the procedures, we suggested and discussed the manner by which this accuracy could be improved. Indeed one of the objectives of this chapter is concerned with the question of accuracy and a possible improvement on it by *a posteriori* treatments of the finite element data. We refer to such processes as *recovery*. We shall also consider the discretization error of the finite element approximation and *a posteriori* estimates of such error. In particular, we describe two distinct types of *a posteriori* error estimators, *recovery-based error estimators* and *residual-based error estimators*. The importance of highly accurate recovery methods in the computation of the recovery-based error estimators is discussed. We also demonstrate how various recovery methods can be used in the construction of residual-based error estimators.

Before proceeding further it is necessary to define what we mean by error. This we consider to be the difference between the exact solution and the approximate one. This can apply to the basic function, such as displacement, which we have called $\mathbf{u}$, and is given as

$$\mathbf{e} = \mathbf{u} - \hat{\mathbf{u}} \tag{15.1}$$

where, as before, $\hat{\mathbf{u}}$ denotes a finite element solution and $\mathbf{u}$ the exact solution. In a similar way, however, we could focus on the error in the strains (i.e., gradients in the solution), such as $\boldsymbol{\epsilon}$ or stresses $\boldsymbol{\sigma}$ and describe the error in these quantities as

$$\begin{aligned} \mathbf{e}_\varepsilon &= \boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}} \\ \mathbf{e}_\sigma &= \boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}} \end{aligned} \tag{15.2}$$

The specification of local error in the manner given in Eqs. (15.1) and (15.2) is generally not convenient and occasionally misleading. For instance, under a point load both errors in displacements and stresses will be locally infinite but the overall solution may well be acceptable. Similar situations will exist near re-entrant corners where, as is well known, stress singularities exist in elastic analysis and gradient singularities develop in field problems. For this reason various "norms" representing some *integral scalar quantity* are often introduced to measure the error.

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00015-0

### 15.1.1 Norms of errors

If, for instance, we are concerned with a general linear equation of the form of Eq. (8.1) (cf. Chapter 8), i.e.,

$$\mathcal{L}\mathbf{u} + \mathbf{b} = \mathbf{0} \tag{15.3}$$

we can define an *energy norm* written for the error as

$$\|\mathbf{e}\| = \left| \int_\Omega \mathbf{e}^{\mathrm{T}} \mathcal{L}\mathbf{e}\, \mathrm{d}\Omega \right|^{\frac{1}{2}} \equiv \left| \int_\Omega (\mathbf{u} - \hat{\mathbf{u}})^{\mathrm{T}}\, \mathcal{L}\, (\mathbf{u} - \hat{\mathbf{u}})\, \mathrm{d}\Omega \right|^{\frac{1}{2}} \tag{15.4}$$

where $|\cdot|$ denotes the absolute value of the argument.

This scalar measure corresponds in fact to the square root of the quadratic functional such as we discussed in Section 4.1 of Chapter 4 where we sought its minimum in the case of a self-adjoint operator $\mathcal{L}$.

For elasticity problems the energy norm is defined in the same manner and yields

$$\|\mathbf{e}\| = \left[ \int_\Omega (\mathcal{S}\mathbf{e})^{\mathrm{T}}\, \mathbf{D}\mathcal{S}\mathbf{e}\, \mathrm{d}\Omega \right]^{\frac{1}{2}} \tag{15.5}$$

(with symbols as used in Chapters 2 and 7).

Here $\mathbf{e}$ is given by Eq. (15.1), the operator $\mathcal{S}$ defines the strains as

$$\boldsymbol{\epsilon} = \mathcal{S}\mathbf{u} \quad \text{and} \quad \hat{\boldsymbol{\epsilon}} = \mathcal{S}\hat{\mathbf{u}} \tag{15.6a}$$

and $\mathbf{D}$ is the elasticity matrix (see Chapters 2 and 7), giving the stress as

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\epsilon} \quad \text{and} \quad \hat{\boldsymbol{\sigma}} = \mathbf{D}\hat{\boldsymbol{\epsilon}} \tag{15.6b}$$

in which for simplicity we ignore initial stresses and strains.

Using the above relations the energy norm of Eq. (15.5) can be written alternatively as

$$\begin{aligned} \|\mathbf{e}\| &= \left[ \int_\Omega (\boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}})^{\mathrm{T}}\, \mathbf{D}\, (\boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}})\, \mathrm{d}\Omega \right]^{\frac{1}{2}} \\ &= \left[ \int_\Omega (\boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}})^{\mathrm{T}}\, (\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}})\, \mathrm{d}\Omega \right]^{\frac{1}{2}} \\ &= \left[ \int_\Omega (\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}})^{\mathrm{T}}\, \mathbf{D}^{-1}\, (\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}})\, \mathrm{d}\Omega \right]^{\frac{1}{2}} \end{aligned} \tag{15.7}$$

and its relation to strain energy is evident.

Other scalar norms can easily be devised. For instance, the $L_2$ norm of displacement error can be written as

$$\|\mathbf{e}\|_{L_2} = \left[ \int_\Omega (\mathbf{u} - \hat{\mathbf{u}})^{\mathrm{T}}\, (\mathbf{u} - \hat{\mathbf{u}})\, \mathrm{d}\Omega \right]^{\frac{1}{2}} \tag{15.8a}$$

and that for stress error as

$$\|\mathbf{e}_\sigma\|_{L_2} = \left[\int_\Omega \left(\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}}\right)^{\mathrm{T}} \left(\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}}\right) \mathrm{d}\Omega\right]^{\frac{1}{2}} \tag{15.8b}$$

Such norms allow us to focus on a particular quantity of interest and indeed it is possible to evaluate "root mean square" (RMS) values of its error. For instance, the RMS error in displacement, $\Delta\mathbf{u}$, becomes for the domain $\Omega$

$$|\Delta\mathbf{u}| = \left(\frac{\|\mathbf{e}\|^2_{L_2}}{\Omega}\right)^{\frac{1}{2}} \tag{15.9}$$

Similarly, the RMS error in stress, $\Delta\boldsymbol{\sigma}$, becomes for the domain $\Omega$

$$|\Delta\boldsymbol{\sigma}| = \left(\frac{\|\mathbf{e}_\sigma\|^2_{L_2}}{\Omega}\right)^{\frac{1}{2}} \tag{15.10}$$

Any of the above norms can be evaluated over the whole domain, any subdomain, or even an individual element.

We note that

$$\|\mathbf{e}\| = \left(\sum_{K=1}^{m} \|\mathbf{e}\|^2_K\right)^{\frac{1}{2}} \tag{15.11}$$

where $K$ refers to individual elements $\Omega_K$ such that their sum (union) is $\Omega$.

We note further that the energy norms given in terms of the stresses, the $L_2$ norm of stress, and the RMS stress error have a very similar structure and that these are similarly approximated.

#### 15.1.1.1 *Effect of a singularity*

At this stage it is of interest to invoke the discussion of Chapter 7 (Section 7.9) concerning the rates of convergence. We noted there that with trial functions in the displacement formulation of degree $p$, the errors in the derivative and stress were of the order $O(h^p)$. This order of error should therefore apply to the energy norm error $\|\mathbf{e}\|$. While the arguments are correct for well-behaved problems with no singularity, it is of interest to see how the above rule is violated when singularities exist.

To describe the behavior of stress analysis problems we define the variation of the *relative energy norm error* (percentage) as

$$\eta = \frac{\|\mathbf{e}\|}{\|\mathbf{u}\|} \times 100\% \tag{15.12}$$

where

$$\|\mathbf{u}\| = \left[\int_\Omega \boldsymbol{\epsilon}^{\mathrm{T}} \mathbf{D}\, \boldsymbol{\epsilon}\, \mathrm{d}\Omega\right]^{\frac{1}{2}} \tag{15.13}$$

is the energy norm of the solution. In Figs. 15.1 and 15.2 we consider two similar stress analysis problems. In the first a strong singularity is present, however, in the

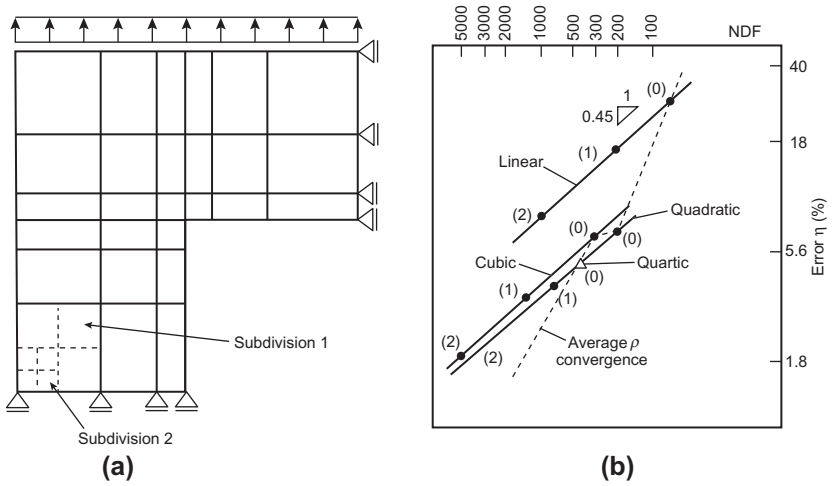



**FIGURE 15.1**
Analysis of L-shaped domain with singularity.

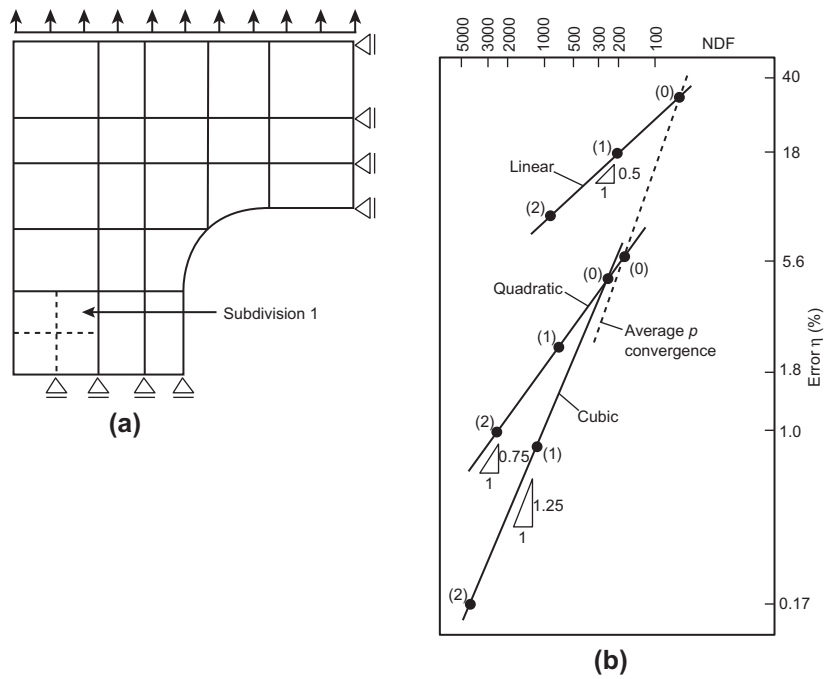



**FIGURE 15.2**
Analysis of L-shaped domain without singularity.

second the singularity is removed by introducing a rounded corner. In both figures we show the relative energy norm error for an $h$-refinement constructed by uniform subdivision of the initial mesh and for a $p$-refinement in which polynomial order is increased throughout the original mesh.

We note two interesting facts. First, the $h$ convergence rate for various polynomial orders of the shape functions is nearly the same in the example with singularity (Fig. 15.1) and is well below the theoretically predicted optimal order $O(h^p)$ [or $O(\text{NDF})^{-p/2}$ as the NDF (number of degrees of freedom) is approximately inversely proportional to $h^2$ for a two-dimensional problem].

Second, in the case shown in Fig. 15.2, where the singularity is avoided by rounding the re-entrant corner, the $h$ convergence rate improves for elements of higher order, although again the theoretical (asymptotic) rate is not quite achieved.

The reason for this behavior is clearly the singularity, and in general it can be shown that the rate of convergence for problems with singularity is

$$O(\text{NDF})^{-[\min(\lambda, p)]/2} \tag{15.14}$$

where $\lambda$ is a number associated with the intensity of the singularity. For elasticity problems $\lambda$ ranges from 0.5 for a nearly closed crack to $\sqrt{0.711}$ for a 90° corner. The rate of convergence illustrated in Fig. 15.2 approaches the theoretically optimal order for all values of $p$ used in the elements.

## 15.2 Superconvergence and optimal sampling points

In this section we shall consider the location of points at which the stresses, or displacements, give their most accurate values in typical problems of a self-adjoint kind. We shall note that on many occasions the displacements, or the function itself, are most accurately sampled at the nodes defining an element and that the gradients or stresses are best sampled at some interior points. Indeed, in one dimension at least, we find that such points often exhibit the quality known as *superconvergence* (i.e., the values sampled at these points show an error which decreases more rapidly than elsewhere). Obviously, the user of finite element analysis should be encouraged to employ such points but at the same time note that the errors overall may be much larger. To clarify these ideas we start with a typical problem of second order in one dimension.

### 15.2.1 A one-dimensional example

Here we consider a problem of a second-order equation such as we discussed in Chapter 3 and which may be typical of either one-dimensional heat conduction or the displacements of an elastic bar with varying cross-section. This equation can readily be written as

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(k\frac{\mathrm{d}u}{\mathrm{d}x}\right) + \beta u + Q = 0 \tag{15.15}$$

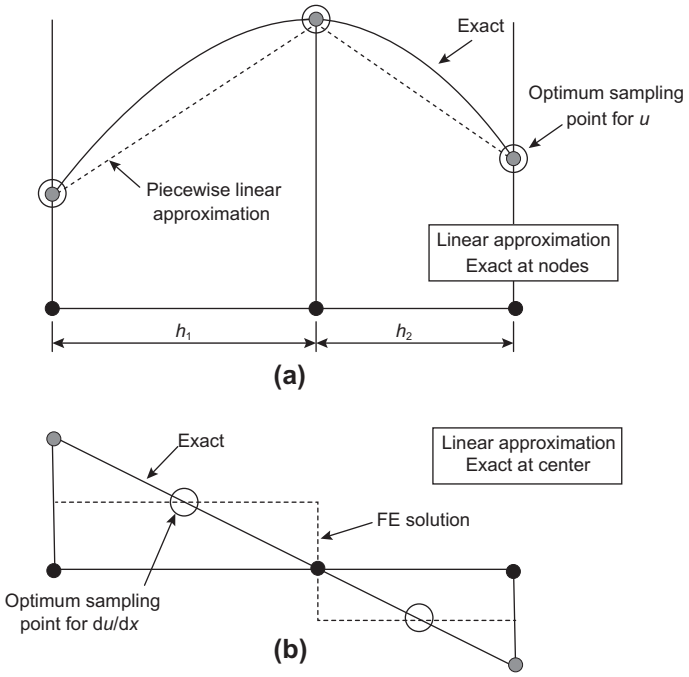



**FIGURE 15.3**
Optimal sampling points for the function (a) and its gradient (b) in one dimension (linear elements).

with the boundary conditions either defining the values of the function $u$ or of its gradients at the ends of the domain.

Let us consider a typical problem as illustrated in Fig. 15.3. Here we show an exact solution for $u$ and $\mathrm{d}u/\mathrm{d}x$ for a span of several elements and indicate the type of solution which will result from a finite element calculation using linear elements. We have already noted that on occasions we shall obtain exact solutions for $u$ at nodes (see Figs. 3.10 and 3.11). This will happen when the weighting function contains the exact solution of the homogeneous differential equation (Appendix G)—a situation which happens for Eq. (15.15) when $\beta = 0, k$ is constant in each element, and polynomial shape functions are used. In all cases, even when $\beta$ is nonzero and linear shape functions are used, the nodal values generally will be much more accurate than those elsewhere (Fig. 15.3a). For the gradients shown in Fig. 15.3b we observe large discrepancies of the finite element solution from the exact solution but we note that somewhere within each element the results are exact.

It would be useful to locate such points and indeed we have already remarked in the context of two-dimensional analysis that values obtained within the elements tend to be more accurate for gradients (strains and stresses) than those values calculated at nodes. Clearly, for the problem illustrated in Fig. 15.3b we should sample somewhere near the center of each element.

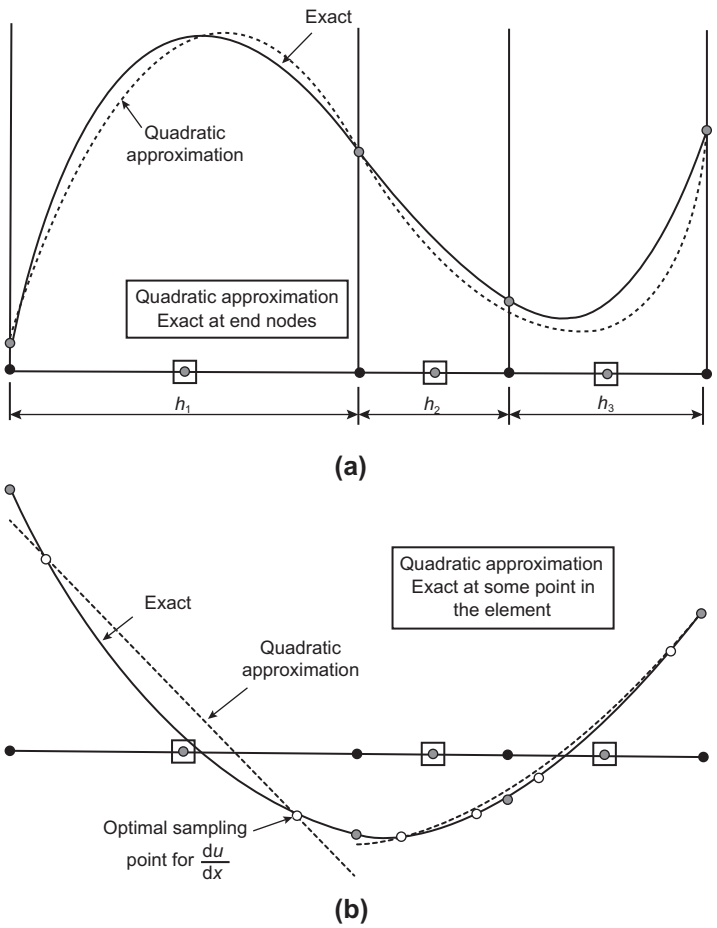



**FIGURE 15.4**

Optimal sampling points for the function (a) and its gradient (b) in one dimension (quadratic elements).

Pursuing this problem further in a heuristic manner we note that if higher order elements (e.g., quadratic elements) are used the solution still remains exact or nearly exact at the end nodes of an element but may depart from exactness at the interior nodes, as shown in Fig. 15.4a. The stresses, or gradients, in this case will be optimal at points which correspond to the two Gauss quadrature points for each element as indicated in Fig. 15.4b. This fact was observed experimentally by Barlow [1].

We shall now state in an axiomatic manner that

**(a)** the displacements are best sampled at the nodes of the element, whatever the order of element used, and

**(b)** the best accuracy for gradients or stresses is obtained at the Gauss points corresponding to the order of polynomial used in the solution.

At such points the order of the convergence of the function or its gradients is at least one order higher than that which would be anticipated from the appropriate polynomial and thus such points are known as *superconvergent*. The reason for such superconvergence will be shown in the next section where we introduce the reader to a theorem developed by Herrmann [2].

### 15.2.2 The Herrmann theorem and optimal sampling points

The concept of least squares fitting has additional justification in self-adjoint problems in which an energy functional is minimized. In such cases, typical of a displacement formulation of elasticity, it can be readily shown that the minimization is equivalent to a least squares fit of the approximate stresses to the exact ones. Thus quite generally we can start from a theory given by the differential equation

$$\mathcal{L}\mathbf{u} = \boldsymbol{\mathcal{S}}^{\mathrm{T}}(\mathbf{A}\boldsymbol{\mathcal{S}}\mathbf{u}) = \mathbf{p} \tag{15.16}$$

In the above, $\mathcal{L}$ is a self-adjoint operator defined by $\boldsymbol{\mathcal{S}}$ and $\mathbf{A}$ (symmetric) and $\mathbf{p}$ are prescribed matrices of position. The *minimization of an energy functional* $\Pi$ defined as

$$\Pi = \frac{1}{2}\int_{\Omega} (\boldsymbol{\mathcal{S}}\mathbf{u})^{\mathrm{T}}\,\mathbf{A}\,\boldsymbol{\mathcal{S}}\mathbf{u}\,\mathrm{d}\Omega - \int_{\Omega}\mathbf{u}^{\mathrm{T}}\mathbf{p}\,\mathrm{d}\Omega \tag{15.17}$$

gives at an absolute minimum the exact solution $\mathbf{u} = \bar{\mathbf{u}}$ and is equivalent to minimization of another functional $\Pi^{\star}$ defined as

$$\Pi^{\star} = \frac{1}{2}\int_{\Omega}\left[\boldsymbol{\mathcal{S}}\left(\mathbf{u}-\bar{\mathbf{u}}\right)\right]^{\mathrm{T}}\mathbf{A}\boldsymbol{\mathcal{S}}\left(\mathbf{u}-\bar{\mathbf{u}}\right)\,\mathrm{d}\Omega \tag{15.18}$$

The above quadratic functional [Eq. (15.17)] arises in all linear self-adjoint problems.

For elasticity problems this theorem is given by Herrmann [2] and shows that the approximate solution for $\boldsymbol{\mathcal{S}}\mathbf{u}$ approaches the exact one $\boldsymbol{\mathcal{S}}\bar{\mathbf{u}}$ as a *weighted least squares approximation*.

The proof of the Herrmann theorem is as follows. The variation of $\Pi$ defined in Eq. (15.17) gives, at $\mathbf{u} = \bar{\mathbf{u}}$ (the exact solution),

$$\delta\Pi = \frac{1}{2}\int_{\Omega}(\boldsymbol{\mathcal{S}}\delta\mathbf{u})^{\mathrm{T}}\,\mathbf{A}\,\boldsymbol{\mathcal{S}}\bar{\mathbf{u}}\,\mathrm{d}\Omega + \frac{1}{2}\int_{\Omega}(\boldsymbol{\mathcal{S}}\bar{\mathbf{u}})^{\mathrm{T}}\,\mathbf{A}\,\boldsymbol{\mathcal{S}}\delta\mathbf{u}\,\mathrm{d}\Omega - \int_{\Omega}\delta\mathbf{u}^{\mathrm{T}}\mathbf{p}\,\mathrm{d}\Omega = 0$$

or as $\mathbf{A}$ is symmetric

$$\delta\Pi = \int_{\Omega}(\boldsymbol{\mathcal{S}}\delta\mathbf{u})^{\mathrm{T}}\,\mathbf{A}\,\boldsymbol{\mathcal{S}}\bar{\mathbf{u}}\,\mathrm{d}\Omega - \int_{\Omega}\delta\mathbf{u}^{\mathrm{T}}\mathbf{p}\,\mathrm{d}\Omega = 0$$

in which $\delta\mathbf{u}$ is any arbitrary variation. Thus we can select

$$\delta\mathbf{u} = \mathbf{u}$$

and

$$\int_{\Omega}(\boldsymbol{\mathcal{S}}\mathbf{u})^{\mathrm{T}}\,\mathbf{A}\,\boldsymbol{\mathcal{S}}\bar{\mathbf{u}}\,\mathrm{d}\Omega - \int_{\Omega}\mathbf{u}^{\mathrm{T}}\mathbf{p}\,\mathrm{d}\Omega = 0$$

Subtracting the above from Eq. (15.17) and noting the symmetry of the **A** matrix, we can write

$$\Pi = \frac{1}{2}\int_{\Omega} \left[\boldsymbol{\mathcal{S}}\left(\mathbf{u}-\bar{\mathbf{u}}\right)\right]^{\mathrm{T}} \mathbf{A}\, \boldsymbol{\mathcal{S}}\left(\mathbf{u}-\bar{\mathbf{u}}\right)\, \mathrm{d}\Omega - \frac{1}{2}\int_{\Omega} \left(\boldsymbol{\mathcal{S}}\bar{\mathbf{u}}\right)^{\mathrm{T}} \mathbf{A}\, \boldsymbol{\mathcal{S}}\bar{\mathbf{u}}\, \mathrm{d}\Omega \tag{15.19}$$

where the last term is not subject to variation. Thus

$$\Pi^{\star} = \Pi + \text{constant} \tag{15.20}$$

and its stationarity is equivalent to the stationarity of $\Pi$.

It follows directly from the Herrmann theorem that, for one dimension and by a well-known property of the Gauss-Legendre quadrature points, if the approximate gradients are defined by a polynomial of degree $p-1$, where $p$ is the degree of the polynomial used for the unknown function $u$, then stresses taken at these quadrature points must be superconvergent. The single point at the center of an element integrates precisely all linear functions passing through that point and, hence, if the stresses are exact to the linear form they will be exact at that point of integration. For any higher order polynomial of order $p$, the Gauss-Legendre points numbering $p$ will also provide points of superconvergent sampling. We see this from Fig. 15.5 directly. Here we indicate one-, two-, and three-point Gauss-Legendre quadrature showing why exact results are recovered there for gradients and stresses.

For points based on rectangles and products of polynomial functions it is clear that the exact integration points will exist at the product points as shown in Fig. 15.6 for various rectangular elements assuming that the weighting matrix **A** is diagonal. In the same figure we show some triangles that appear to be "good" but are not necessarily superconvergent sampling points. Though we find that superconvergent points do not exist in triangles, the points shown in Fig. 15.6 are optimal. In Fig. 15.6 we contrast these points with the minimum number of quadrature points necessary for obtaining an accurate (though not always stable) stiffness representation and find these to be almost coincident at all times.

In Fig. 15.7 representing an analysis of a cantilever beam by four rectangular quadratic serendipity elements we see how well the stresses sampled at superconvergent points behave compared to the overall stress pattern computed in each element.

The extension of the idea of superconvergent points from one-dimensional elements to two-dimensional rectangles is fairly obvious. However, the full order of superconvergence is lost when isoparametric distortion of elements occurs. We have shown, however, that results at the $p$th-order Gauss-Legendre points still remain excellent and we suggest that superconvergent properties of the integration points continue to be used for sampling.

In all of the above discussion we have assumed that the weighting matrix **A** is diagonal. If a diagonal structure does not exist the existence of superconvergent points is questionable. However excellent results are still available through the sampling points defined as above.

Finally, we refer readers to Refs. [3–8] for surveys on the superconvergence phenomenon and its detailed analyses.

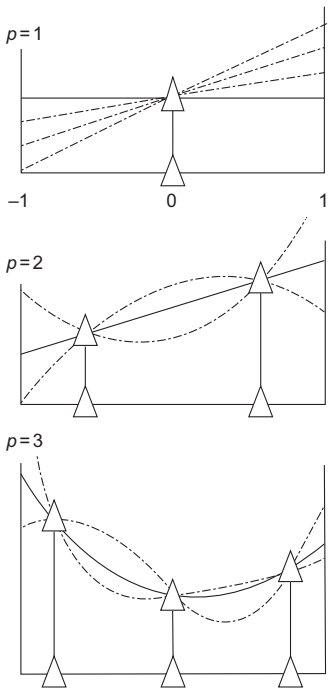



**FIGURE 15.5**
The integration property of Gauss points: $p = 1$, $p = 2$, and $p = 3$ which guarantees superconvergence.

## 15.3 Recovery of gradients and stresses

In the previous section we have shown that sampling of the gradients and stresses at certain points within an element is optimal and higher order accuracy can be achieved. However, we would also like to have similarly accurate quantities elsewhere within each element for general analysis purposes, and in particular we need such highly accurate displacements, gradients, and stresses when energy norm or other norms representing the particular quantity of interest have to be evaluated in error estimates. We have already shown how with some elements very large errors exist beyond the superconvergent point and attempts have been made from the earliest days to obtain a complete picture of stresses which is more accurate overall. Here attempts are generally made to recover the nodal values of stresses and gradients from those sampled internally and then to assume that throughout the element the recovered stresses $\boldsymbol{\sigma}^\star$ are obtained by interpolation in the same manner as the displacements

$$\boldsymbol{\sigma}^\star = \mathbf{N}_u \tilde{\boldsymbol{\sigma}}^\star \tag{15.21}$$

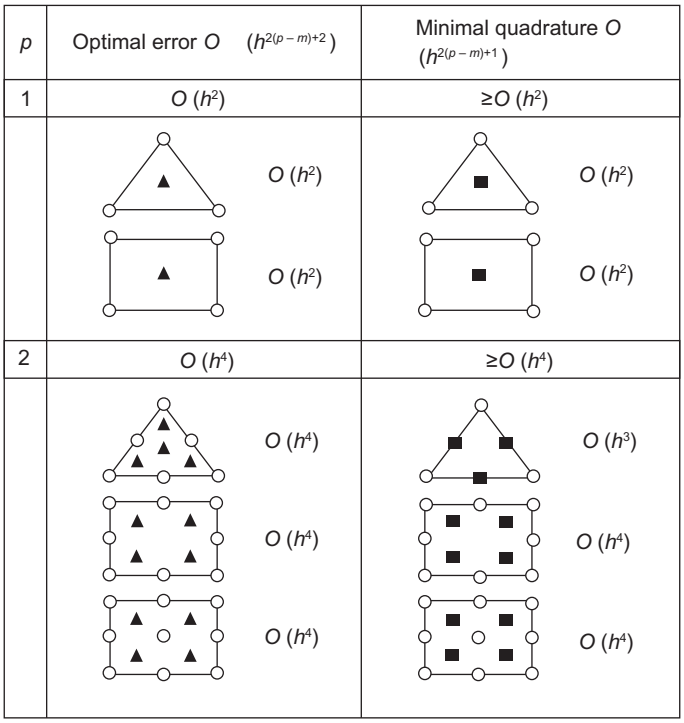



**FIGURE 15.6**
Optimal superconvergent sampling and minimum integration points for some $C_0$ elements.

We have already suggested a process used almost from the beginning of finite element calculations for triangular elements, where elements are sampled at the centroid (assuming linear shape functions have been used) and then the stresses are averaged at nodes. We referred to such recovery in Chapter 7. However this is not the best for triangles and for higher order elements such averaging is inadequate. Here other procedures were necessary, for instance Hinton and Campbell [9] suggested a method in which stresses at all nodes were calculated by extrapolating the Gauss point values. A method of a similar kind was suggested by Brauchli and Oden [10] who used the stresses in the manner given by Eq. (15.21) and assumed that these stresses should represent in a least squares sense the actual finite element stresses. This is therefore an $L_2$ projection. Although this has a similarity with the ideas contained in the Herrmann theorem, it reverses the order of least squares application and has not proved to be always stable and accurate, especially for even order elements. In the following presentation we will show that highly improved results can be obtained by direct polynomial "smoothing" of the optimal values. Here the first method of importance is called *superconvergent patch recovery* [11–13].

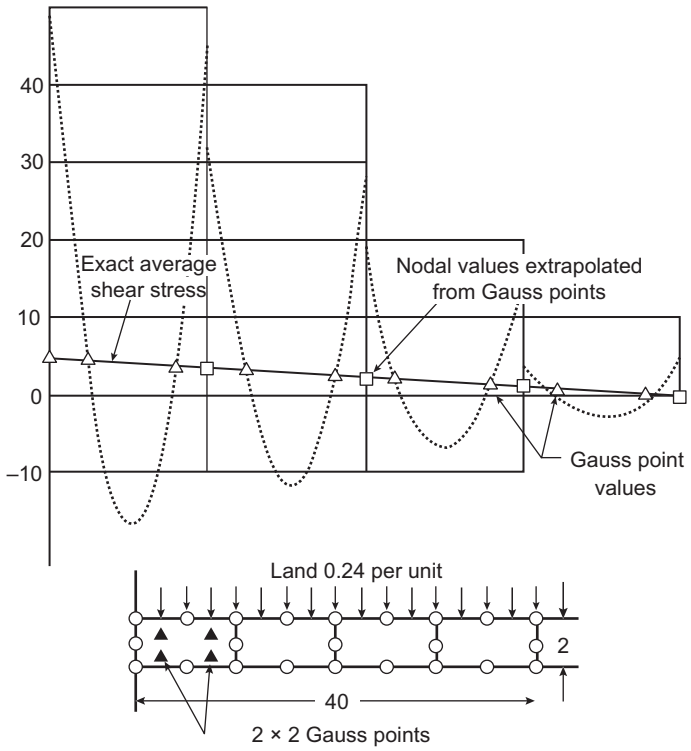



**FIGURE 15.7**

Cantilever beam with four quadratic (Q8) elements. Stress sampling at cubic order ($2 \times 2$) Gauss points with extrapolation to nodes.

## 15.4 Superconvergent patch recovery (SPR)

### 15.4.1 Recovery for gradients and stresses

We have noted above that the stresses sampled at certain points in an element possess a superconvergent property (i.e., converge at a rate comparable to that of displacement) and have errors of order $O(h^{p+1})$. A fairly obvious procedure for utilizing such sampled values seems to the authors to be that of involving a smoothing of such values by a polynomial of order $p$ within a *patch of elements* for which the number of sampling points can be taken as greater than the number of parameters in the polynomial. In Fig. 15.8 we show several such patches each assembled around an interior vertex (corner) node. The first four represent rectangular elements where the superconvergent points are well defined. The last two give patches of triangles where the "optimal" sampling points used are not quite superconvergent.

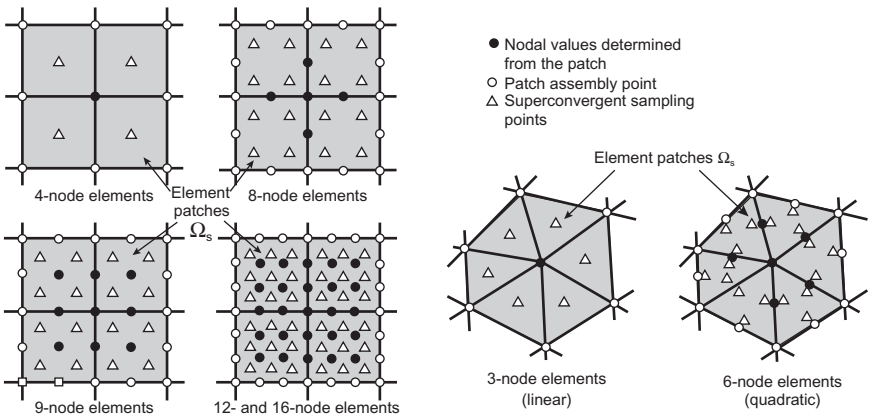



**FIGURE 15.8**

Interior superconvergent patches for quadrilateral elements (linear, quadratic, and cubic) and triangles (linear and quadratic).

If we accept the superconvergence of $\hat{\boldsymbol{\sigma}}$ at certain points $k$ in each element then it is a simple matter (which also turns out computationally much less expensive than the $L_2$ projection) *to compute* $\boldsymbol{\sigma}^\star$ *which is superconvergent at all points within the element*. The procedure is illustrated for two dimensions in Fig. 15.8, where we shall consider interior patches (assembling all elements at interior nodes) as shown.

At each superconvergent point the values of $\hat{\boldsymbol{\sigma}}$ are accurate to order $p+1$ (not $p$ as is true elsewhere). However, we can easily obtain an approximation $\bar{\boldsymbol{\sigma}}^\star$ given by a polynomial of degree $p$, with identical order to those occurring in the shape function for displacement, which has superconvergent accuracy everywhere when this polynomial is made to fit the superconvergent points in a least squares manner.

Thus we proceed for each component $\hat{\sigma}_i$ of $\hat{\boldsymbol{\sigma}}$ as follows: writing the recovered solution as

$$\bar{\sigma}_i^\star = \mathbf{p}(x, y)\mathbf{a}_i \tag{15.22a}$$

in which

$$\begin{aligned} \mathbf{p}(x, y) &= \left[\, 1 \;\; \bar{x} \;\; \bar{y} \;\; \ldots \;\; \bar{y}^p \,\right] \\ \mathbf{a}_i &= \left[\, a_1 \;\; a_2 \;\; \ldots \;\; a_m \,\right]^{\mathrm{T}} \end{aligned} \tag{15.22b}$$

with $\bar{x} = x - x_c$, $\bar{y} = y - y_c$ where $x_c$, $y_c$ are the coordinates of the interior vertex node describing the patch.

For each element patch we minimize a least squares functional with $n$ sampling points,

$$\Pi = \frac{1}{2} \sum_{k=1}^{n} \left[ \hat{\sigma}_i(x_k, y_k) - \mathbf{p}_k \mathbf{a}_i \right]^2 \tag{15.23}$$

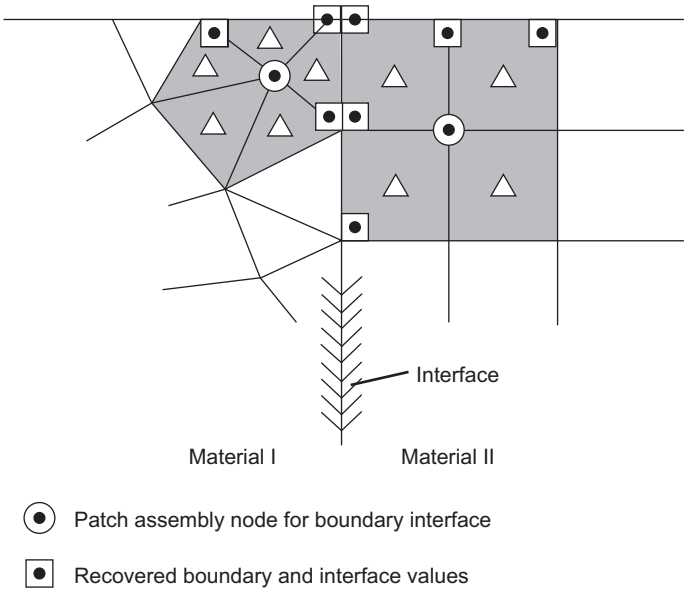



**FIGURE 15.9**
Recovery of boundary or interface gradients.

where

$$\mathbf{p}_k = \mathbf{p}(x_k, y_k)$$

[$(x_k, y_k)$ correspond to the coordinates of the sampling superconvergent point $k$] obtaining immediately the coefficient $\mathbf{a}_i$ as

$$\mathbf{a}_i = \mathbf{A}^{-1}\mathbf{b}_i \tag{15.24}$$

where

$$\mathbf{A} = \sum_{k=1}^{n} \mathbf{p}_k^{\mathrm{T}}\mathbf{p}_k \quad \text{and} \quad \mathbf{b}_i = \sum_{k=1}^{n} \mathbf{p}_k^{\mathrm{T}}\hat{\sigma}_i(x_k, y_k) \tag{15.25}$$

The availability of $\bar{\boldsymbol{\sigma}}^\star$ allows superconvergent values of $\tilde{\boldsymbol{\sigma}}^\star$ to be determined at all nodes. For example, each component of the recovered solution at node $a$ in the element patch is obtained by

$$\left(\tilde{\sigma}_i^\star\right)_a = \bar{\sigma}_i^\star(x_a, y_a) = \mathbf{p}(x_a, y_a)\,\mathbf{a}_i \tag{15.26}$$

It should be noted that on external boundaries or indeed on interfaces where stresses are discontinuous the nodal values should be calculated from interior patches and evaluated in the manner shown in Fig. 15.9. As some nodes belong to more than one patch, average values of $\tilde{\boldsymbol{\sigma}}^\star$ are best obtained. The superconvergence of $\boldsymbol{\sigma}^\star$ throughout each element is established by Eq. (15.21).

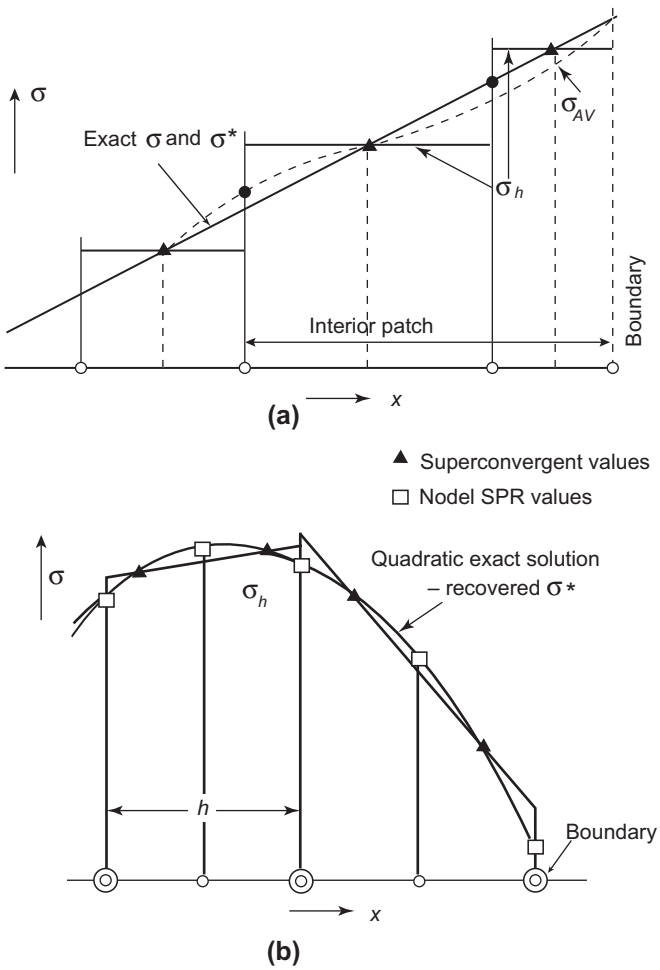



**FIGURE 15.10**
Recovery of exact $\sigma$ of degree $p$ by linear elements ($p = 1$) and quadratic elements ($p = 2$).

In Fig. 15.10 we show in a one-dimensional example how the superconvergent patch recovery reproduces *exactly* the stress (gradient) solutions of order $p + 1$ for linear or quadratic elements. Following the arguments of Chapter 8 on the patch test it is evident that superconvergent recovery is now achieved at all points. Indeed, the same figure shows why averaging (or $L_2$ projection) is inferior (particularly on boundaries).

Fig. 15.11 shows experimentally determined convergence rates for a one-dimensional problem (stress distribution in a bar of length $L = 1, 0 \leq x \leq 1$ and prescribed body forces). A uniform subdivision is used here to form the elements, and the convergence rates for the stress error at $x = 0.5$ are shown using the direct

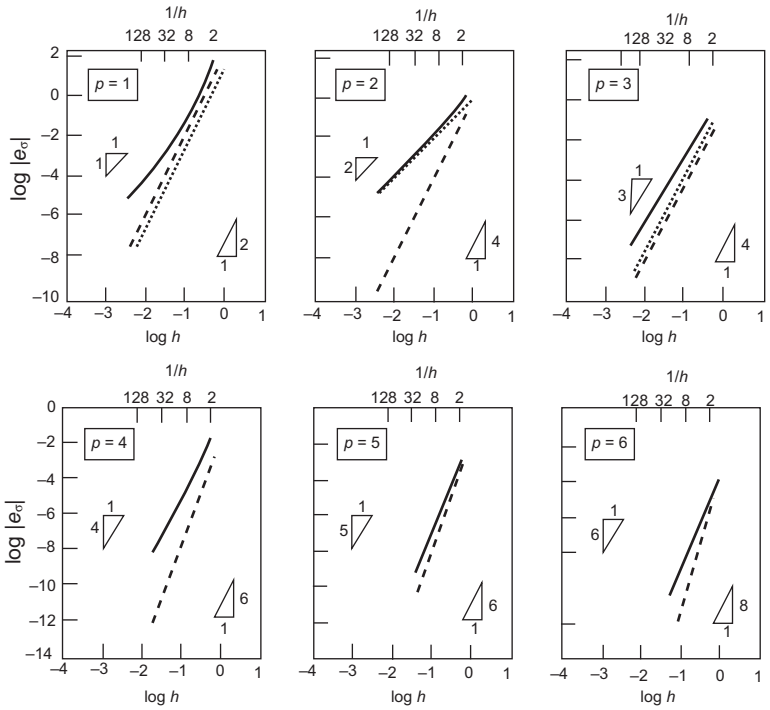



**FIGURE 15.11**

Problem of a stressed bar. Rates of convergence (error) of stress, where $x = 0.5$ ($0 \leq x \leq 1$) ($\hat{\sigma}$ ———; $\sigma_L$ · · · ·; $\sigma^\star$ - - - - -).

stress approximation $\hat{\sigma}$, the $L_2$ recovery $\sigma_L$, and $\sigma^\star$ obtained by the SPR procedure using elements from order $p = 1$ to $p = 6$. It is immediately evident that $\sigma^\star$ is superconvergent with a rate of convergence being at least one order higher than that of $\hat{\sigma}$. However, as anticipated, the $L_2$ recovery gives much poorer answers, showing superconvergence only for odd values of $p$ and almost no improvement for even values of $p$, while $\sigma^\star$ shows a two-order increase of convergence rate for even order elements (tests on higher order polynomials are reported in reference [14]). This *ultra-convergence* has been verified mathematically [15,16]. Although it is not observed when elements of varying sizes are used, the important tests shown in Figs. 15.12 and 15.13 indicate how well the recovery process works for problems in two dimensions.

In the first of these, Fig. 15.12, a field problem is solved in two dimensions using a very irregular mesh for which the existence of superconvergent points is only inferred heuristically. The very small error in $\sigma_x^\star$ is compared with the error of $\hat{\sigma}_x$ and the improvement is obvious. Here $\sigma_x = \partial u/\partial x$ where $u$ is the field variable.

In the second, i.e., Fig. 15.13, a problem of stress analysis, for which an exact solution is known, is solved using three different recovery methods. Once again the recovered solution $\sigma^\star$ (SPR) shows much improved values compared with $\sigma_L$. It is

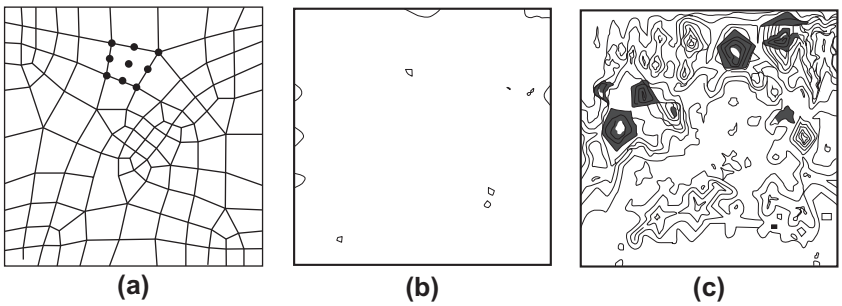



**FIGURE 15.12**
Poisson equation in two dimensions solved using arbitrary shaped quadratic quadrilaterals: (a) Arbitrary mesh, (b) error of $\sigma_x^*$ and (c) error of $\hat{\sigma}_x$.

clear that the SPR process *should be included in all codes if simply to present improved stress values*, to which we have already alluded in Chapter 7.

The SPR procedure which we have just outlined has proved to be a very powerful tool leading to superconvergent results on regular meshes and much improved results (nearly superconvergent) on irregular meshes. It has been shown numerically that it produces superconvergent recovery even for triangular elements which do not have superconvergent points within the element. Mathematical proofs have confirmed these capabilities of SPR [17,16,18–21]. It is also found, for linear elements on irregular meshes, that SPR produces superconvergence of order $O(h^{1+\alpha})$ with $\alpha$ greater than zero [22]. The SPR procedure, introduced by Zienkiewicz and Zhu [11–13], is recommended as the best recovery procedure that is simple to use. However, the patch recovery procedure has been modified by various investigators [23–31]. Many variants of the patch recovery procedure have also been proposed [32–39]. Some of the approaches have been shown to produce improved results in certain instances but with additional computational costs. Many of the modifications are based on satisfaction of equilibrium equations and/or boundary conditions to improve the accuracy. While the satisfaction of known boundary traction can be useful, the additional constraints introduced in most cases do not improve the superconvergent properties.

**Example 15.1.** SPR stress projection for rectangular element patch
As an example we consider the SPR projection for a stress component $\sigma_i$ on the patch of rectangular elements shown in Fig. 15.14. The elements are four-node rectangles in which shape functions are given by bilinear interpolations. Thus, the optimal sampling points are given by the points at the center of each element.
The recovered solution is given by a linear polynomial expressed as

$$\sigma_i^\star = [1 \quad (x - x_1) \quad (y - y_1)] \begin{Bmatrix} \bar{a}_1 \\ \bar{a}_2 \\ \bar{a}_3 \end{Bmatrix}$$

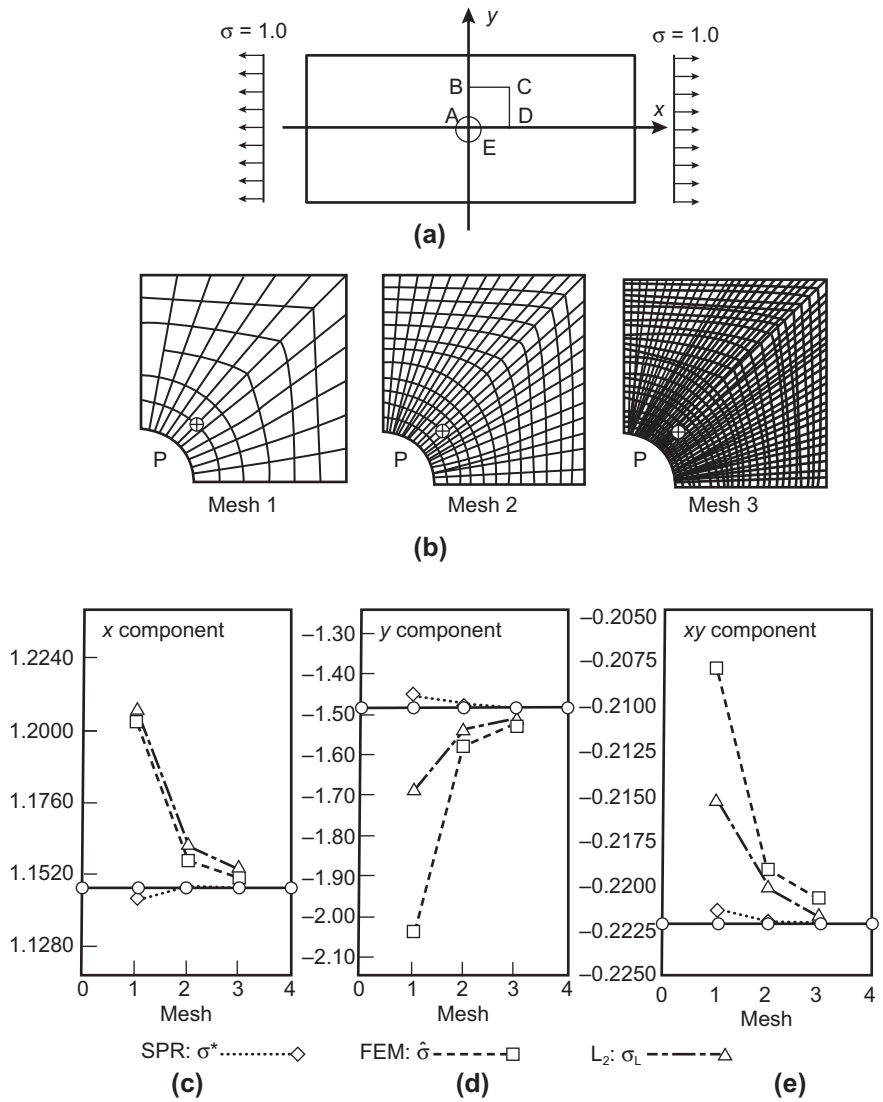



**FIGURE 15.13**
Plane stress analysis of stresses around a circular hole in a uniaxial field.

For this patch of elements, (15.23) is given by

$$\Pi = \frac{1}{2} \sum_{k=1}^{4} \left[ \hat{\sigma}_i(x_k, y_k) - \mathbf{p}_k \mathbf{a} \right]^2$$

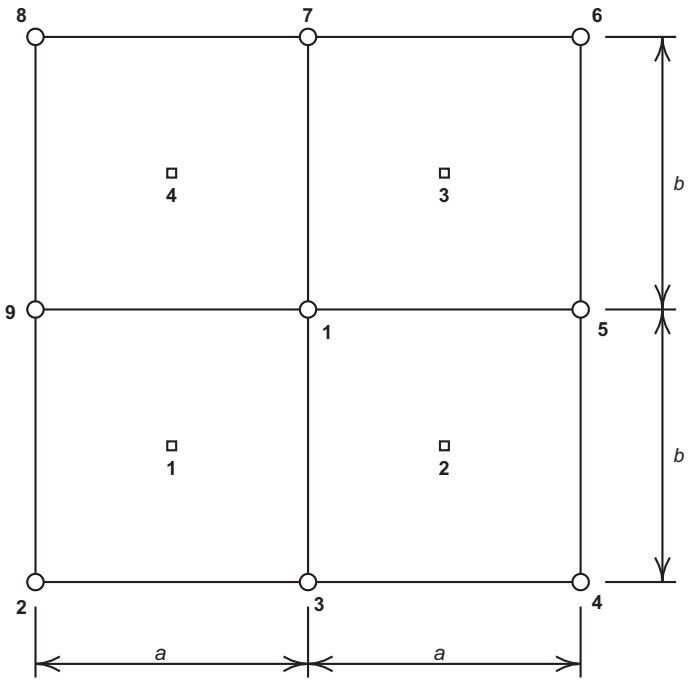



**FIGURE 15.14**
Patch of rectangular elements for SPR projection. Optimal points to sample stresses indicated by □.

where

$$\mathbf{p}_k = \begin{cases} [1 & -a/2 & -b/2] & \text{for } k = 1 \\ [1 & a/2 & -b/2] & \text{for } k = 2 \\ [1 & a/2 & b/2] & \text{for } k = 3 \\ [1 & -a/2 & b/2] & \text{for } k = 4 \end{cases}$$

Evaluating the minimum for $\Pi$ and performing the sum gives the equations

$$\mathbf{A}\,\mathbf{a} = \mathbf{b}$$

where

$$\mathbf{A} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & a^2 & 0 \\ 0 & 0 & b^2 \end{bmatrix}$$

and

$$\mathbf{b} = \left( \begin{Bmatrix} 1 \\ -a/2 \\ -b/2 \end{Bmatrix} \hat{\sigma}_{i1} + \begin{Bmatrix} 1 \\ a/2 \\ -b/2 \end{Bmatrix} \hat{\sigma}_{i2} + \begin{Bmatrix} 1 \\ a/2 \\ b/2 \end{Bmatrix} \hat{\sigma}_{i3} + \begin{Bmatrix} 1 \\ -a/2 \\ b/2 \end{Bmatrix} \hat{\sigma}_{i4} \right)$$

The solution for the parameters is given by

$$a_1 = \frac{1}{4}\left[\hat{\sigma}_{i1} + \hat{\sigma}_{i2} + \hat{\sigma}_{i3} + \hat{\sigma}_{i4}\right]$$
$$a_2 = \frac{1}{2a}\left[-\hat{\sigma}_{i1} + \hat{\sigma}_{i2} + \hat{\sigma}_{i3} - \hat{\sigma}_{i4}\right]$$
$$a_3 = \frac{1}{2b}\left[-\hat{\sigma}_{i1} - \hat{\sigma}_{i2} + \hat{\sigma}_{i3} + \hat{\sigma}_{i4}\right]$$

Inserting the parameters into the equation for the recovered stress gives

$$\sigma_i^\star = \begin{bmatrix} \frac{1}{4} & \frac{(x - x_1)}{2a} & \frac{(y - y_1)}{2b} \end{bmatrix} \begin{Bmatrix} \hat{\sigma}_{i1} + \hat{\sigma}_{i2} + \hat{\sigma}_{i3} + \hat{\sigma}_{i4} \\ -\hat{\sigma}_{i1} + \hat{\sigma}_{i2} + \hat{\sigma}_{i3} - \hat{\sigma}_{i4} \\ -\hat{\sigma}_{i1} - \hat{\sigma}_{i2} + \hat{\sigma}_{i3} + \hat{\sigma}_{i4} \end{Bmatrix}$$

We note that the above yields SPR values at an internal node of a regular mesh which are the same as that obtained by averaging. Unfortunately, this is not the case when the mesh is irregular or boundary nodes are considered, as the reader can easily establish, where SPR will retain high accuracy but averaging will not.

### 15.4.2 SPR for displacements and stresses

The superconvergent patch recovery can be extended to produce superconvergent displacements. The procedure for the displacements is quite simple if we assume the superconvergent points to be at nodes of the patch. However, as we have already observed it is always necessary to have more data than the number of coefficients in the particular polynomial to be able to execute a least squares minimization. Here of course we occasionally need a patch which extends further than before, particularly since the displacements will be given by a polynomial one order higher than that used for the shape functions. In Fig. 15.8 however we show for most assemblies that an identical patch to that used for stresses will suffice. Larger element patches have also been suggested in Refs. [32,37], and are useful especially at the boundaries of the domain.

The recovered solution $\mathbf{u}^\star$ has on occasion been used in dynamic problems (e.g., Wiberg [32,33]), since in this class of problems the displacements themselves are often important. We also find such recovery useful in problems of fluid dynamics.

When both recovered displacements and stresses are desired, it is advantageous to compute the recovered stresses directly using the derivatives of the recovered displacements. The advantage of computing recovered stresses directly from displacements means that we have now obtained fully superconvergent results for all element types. Indeed, the study by Zhang and Naga [37], for field problems, has found that SPR using nodal field variable sampling, also known as PPR (polynomial preserving recovery), produces better recovered gradients in certain instances. For example, although both SPR and PPR achieve ultra-convergence in the recovered gradient at vertex nodes of quadratic triangles, ultra-convergence of the recovered gradient at the

mid-edge nodes can only be obtained by PPR. Its application in the computation of superconvergent eigenvalues also has been reported [40]. A similar procedure to that studied in Refs. [37,40] has been used by Wiberg and Hager [41]. Thus, field variable recovery should probably always be used for triangular and tetrahedral elements, as well as for other element types when both superconvergent displacements and stresses or strains are required.

The SPR recovery technique described in this section takes advantage of the superconvergence property of the finite element solutions and/or the availability of optimal sampling points. A recovery method which does not need such information has been devised and will be discussed in the next section.

## 15.5 Recovery by equilibration of patches (REP)

Although SPR has proved to work well generally and much research has been devoted to its mathematical analyses, the reason behind its capability of producing an accurate recovered solution even when superconvergent points do not in fact exist remains an open question. We have therefore sought to determine viable recovery alternatives. One of these, known by the acronym REP (recovery by equilibrium of patches), will be described next. This procedure was first presented in Ref. [34] and later improved in Ref. [35].

To some extent the motivation is similar to that of Ladevèze et al. [42,43] who sought to establish (for somewhat different reasons) a fully equilibrating stress field which can replace that of the finite element approximation. However, we believe that the process presented here and in Ref. [35] is simpler although equilibrium is satisfied in an approximate manner.

The starting point for REP is the governing equilibrium equation

$$\boldsymbol{\mathcal{S}}^{\mathrm{T}}\boldsymbol{\sigma} + \mathbf{b} = \mathbf{0} \tag{15.27}$$

In a finite element approximation this becomes

$$\int_{\Omega_p} \mathbf{B}^{\mathrm{T}}\hat{\boldsymbol{\sigma}}\,\mathrm{d}\Omega - \int_{\Omega_p} \mathbf{N}^{\mathrm{T}}\mathbf{b}\,\mathrm{d}\Omega - \int_{\Gamma_p} \mathbf{N}^{\mathrm{T}}\mathbf{t}\,\mathrm{d}\Gamma = \mathbf{0} \tag{15.28}$$

where $\hat{\boldsymbol{\sigma}}$ are the stresses from the finite element solution. In the above $\Omega_p$ is the domain of a patch and the last term comes from the traction on the boundary of the patch domain $\Gamma_p$. These can, of course, represent the whole problem, a patch of a few elements, or a single element.

As is well known the stresses $\hat{\boldsymbol{\sigma}}$ which result from the finite element analysis will in general be discontinuous and we shall seek to replace them in *every element patch* by a recovered system which is smooth and continuous.

To achieve the recovery we proceed in an analogous way to that used in the SPR procedure, *first* approximating the stress in each patch by a polynomial of appropriate order $\bar{\boldsymbol{\sigma}}^{\star}$, *second* using this approximation to obtain nodal values of $\tilde{\boldsymbol{\sigma}}^{\star}$, and *finally* interpolating these values by standard shape functions.

The stress $\boldsymbol{\sigma}$ is taken as a vector of appropriate components, which for convenience we write as

$$\boldsymbol{\sigma} = \begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_n \end{Bmatrix} \tag{15.29}$$

The above notation is general with, for instance, $\sigma_1 = \sigma_x$, $\sigma_2 = \sigma_y$, and $\sigma_3 = \tau_{xy}$ describing a two-dimensional plane elastic analysis.

We shall write each component of the above as a polynomial expansion of the form

$$\bar{\sigma}_i^\star = \begin{bmatrix} 1 & \bar{x} & \bar{y} & \dots \end{bmatrix} \mathbf{a}_i = \mathbf{p}(x, y)\mathbf{a}_i \tag{15.30}$$

where $\mathbf{p}$ is a vector of polynomials, $\mathbf{a}_i$ is a set of unknown coefficients for the $i$th component of stress, and $\bar{x}$, $\bar{y}$ are as described for (15.22b).

For equilibrium we shall always attempt to ensure that the smoothed stress $\bar{\boldsymbol{\sigma}}^\star$ satisfies in a least squares sense the same patch equilibrium conditions as the finite element solution. Accordingly,

$$\int_{\Omega_p} \mathbf{B}^{\mathrm{T}} \hat{\boldsymbol{\sigma}} \, \mathrm{d}\Omega \approx \int_{\Omega_p} \mathbf{B}^{\mathrm{T}} \bar{\boldsymbol{\sigma}}^\star \, \mathrm{d}\Omega \tag{15.31}$$

where

$$\bar{\boldsymbol{\sigma}}^\star = \mathbf{P}\mathbf{a} = \begin{bmatrix} \mathbf{p} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{p} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{p} \end{bmatrix} \begin{Bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{Bmatrix} \tag{15.32}$$

written here again for the case of three stress components. Obvious modifications are made for more or fewer components.

It has been found in practice that the constraints provided by Eq. (15.31) are not sufficient to always produce nonsingular least squares minimization. Accordingly, the equilibrium constraints are split into an alternative form in which each component of stress is subjected to equilibrium requirements. This may be achieved by expressing the stress as

$$\begin{aligned} \bar{\boldsymbol{\sigma}}^\star &= \sum_i \mathbf{1}_i \bar{\sigma}_i^\star = \sum_i \bar{\boldsymbol{\sigma}}_i^\star \\ \hat{\boldsymbol{\sigma}} &= \sum_i \mathbf{1}_i \hat{\sigma}_i = \sum_i \hat{\boldsymbol{\sigma}}_i \end{aligned} \tag{15.33}$$

in which

$$\mathbf{1}_1 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^{\mathrm{T}}, \quad \mathbf{1}_2 = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^{\mathrm{T}} \quad \text{etc.} \tag{15.34}$$

The equations are now obtained by imposing the set of constraints

$$\int_{\Omega_p} \mathbf{B}^{\mathrm{T}} \hat{\boldsymbol{\sigma}}_i \, \mathrm{d}\Omega \approx \int_{\Omega_p} \mathbf{B}^{\mathrm{T}} \bar{\boldsymbol{\sigma}}_i^\star \, \mathrm{d}\Omega = \int_{\Omega_p} \mathbf{B}^{\mathrm{T}} \mathbf{1}_i \mathbf{p} \, \mathrm{d}\Omega \, \mathbf{a}_i \tag{15.35}$$

The imposition of the approximate Eq. (15.35) allows each set of coefficients $\mathbf{a}_i$ to be solved independently reducing considerably the solution cost and here repeating a procedure used with success in SPR.

A least squares minimization of Eq. (15.35) is expressed as

$$\Pi = \frac{1}{2}\left(\mathbf{H}_i\mathbf{a}_i - \mathbf{f}_i^p\right)^{\mathrm{T}}\left(\mathbf{H}_i\mathbf{a}_i - \mathbf{f}_i^p\right) \tag{15.36}$$

where

$$\mathbf{H}_i = \int_{\Omega_p} \mathbf{B}^{\mathrm{T}}\mathbf{1}_i\mathbf{p}\,\mathrm{d}\Omega \quad \text{and} \quad \mathbf{f}_i^p = \int_{\Omega_p} \mathbf{B}^{\mathrm{T}}\hat{\boldsymbol{\sigma}}_i\,\mathrm{d}\Omega \tag{15.37}$$

The minimization condition results in

$$\mathbf{a}_i = \left[\mathbf{H}_i^{\mathrm{T}}\mathbf{H}_i\right]^{-1}\mathbf{H}_i^{\mathrm{T}}\mathbf{f}_i^p \tag{15.38}$$

Nodal values $\tilde{\boldsymbol{\sigma}}^\star$ are obtained from Eq. (15.30) and the final recovered solution is given by Eq. (15.21).

The REP procedure follows precisely the details of SPR near boundaries and gives overall an approximation which does not require knowledge of any superconvergent points. The accuracy of both processes is comparable.

## 15.6 Error estimates by recovery

One of the most important applications of the recovery methods is its use in the computation of *a posteriori* error estimators. With the recovered solutions available, we can now evaluate errors simply by replacing the exact values of quantities such as $\mathbf{u}$, $\boldsymbol{\sigma}$, etc., which are in general unknown, in Eqs. (15.1) and (15.2), by the recovered values which are more accurate than the direct finite element solution. We write the error estimators in various norms such as

$$\begin{aligned} \|\mathbf{e}\| &\approx \|\hat{\mathbf{e}}\| = \|\mathbf{u}^\star - \hat{\mathbf{u}}\| \\ \|\mathbf{e}\|_{L_2} &\approx \|\hat{\mathbf{e}}\|_{L_2} = \|\mathbf{u}^\star - \hat{\mathbf{u}}\|_{L_2} \\ \|\mathbf{e}_\sigma\|_{L_2} &\approx \|\hat{\mathbf{e}}_\sigma\|_{L_2} = \|\boldsymbol{\sigma}^\star - \hat{\boldsymbol{\sigma}}\|_{L_2} \end{aligned} \tag{15.39}$$

For example, an error estimator of the energy norm for elasticity problems has the form

$$\|\hat{\mathbf{e}}\| = \left[\int_\Omega \left(\boldsymbol{\sigma}^\star - \hat{\boldsymbol{\sigma}}\right)^{\mathrm{T}}\mathbf{D}^{-1}\left(\boldsymbol{\sigma}^\star - \hat{\boldsymbol{\sigma}}\right)\,\mathrm{d}\Omega\right]^{\frac{1}{2}} \tag{15.40}$$

Similarly, estimates of the RMS error in displacement and stress can be obtained through Eqs. (15.9) and (15.10). Error estimators formulated by replacing the exact solution with the recovered solution are sometimes called *recovery-based error estimators*. This type of error estimator was first introduced by Zienkiewicz and Zhu [44].

The accuracy or the quality of the error estimators is measured by the *effectivity index* $\theta$, which is defined as

$$\theta = \frac{\|\hat{\mathbf{e}}\|}{\|\mathbf{e}\|} \tag{15.41}$$

A theorem presented by Zienkiewicz and Zhu [12] shows that for all estimators based on recovery we can establish the following bounds for the effectivity index:

$$1 - \frac{\|\mathbf{e}^\star\|}{\|\mathbf{e}\|} \leq \theta \leq 1 + \frac{\|\mathbf{e}^\star\|}{\|\mathbf{e}\|} \tag{15.42}$$

where $\mathbf{e}$ is the actual error and $\mathbf{e}^\star$ is the error of the recovered solution, e.g.,

$$\|\mathbf{e}^\star\| = \|\mathbf{u} - \mathbf{u}^\star\| \tag{15.43}$$

The proof of the above theorem is straightforward if we write Eq. (15.40) as

$$\|\hat{\mathbf{e}}\| = \|\mathbf{u}^\star - \hat{\mathbf{u}}\| = \| \left(\mathbf{u} - \hat{\mathbf{u}}\right) - \left(\mathbf{u} - \mathbf{u}^\star\right) \| = \|\mathbf{e} - \mathbf{e}^\star\| \tag{15.44}$$

Using now the triangle inequality we have

$$\|\mathbf{e}\| - \|\mathbf{e}^\star\| \leq \|\hat{\mathbf{e}}\| \leq \|\mathbf{e}\| + \|\mathbf{e}^\star\| \tag{15.45}$$

from which the inequality (15.42) follows after division by $\|\mathbf{e}\|$. Obviously, the theorem is also true for error estimators of other norms. Two important conclusions follow:

1. *Any recovery process* which results in reduced error will give a reasonable error estimator.
2. More importantly, if the recovered solution converges at a higher rate than the finite element solution we shall always have asymptotically exact estimation.

To prove the second point we consider a typical finite element solution with shape functions of order $p$ where we know that the error (in the energy norm) is

$$\|\mathbf{e}\| = O(h^p) \tag{15.46}$$

If the recovered solution gives an error of a higher order, e.g.,

$$\|\mathbf{e}^\star\| = O(h^{p+\alpha}), \qquad \alpha > 0 \tag{15.47}$$

then the bounds of the effectivity index are

$$1 - O(h^\alpha) \leq \theta \leq 1 + O(h^\alpha) \tag{15.48}$$

and the error estimator is asymptotically exact, that is

$$\theta \to 1 \quad \text{as} \quad h \to 0 \tag{15.49}$$

This means that the error estimator converges to the true error. This is a very important property of error estimators based on recovery and is not generally shared by residual-based estimators, which we discuss in the next section.

## 15.7 Residual-based methods

Other methods to obtain error estimators have been proposed by many investigators working in the field [45–57]. Most of these make use of the residuals of the finite element approximation, either explicitly or implicitly. Error estimators based on these methods are often called *residual error estimators*. Those using residuals explicitly are termed explicit residual error estimators; the others are called implicit residual error estimators.

In this section we are concerned with both explicit and implicit residual error estimators. To simplify the presentation, we use the quasi-harmonic equation in a two-dimensional domain as the model problem. The governing equation of the problem is given by

$$-\nabla^{\mathrm{T}}\left(k\,\nabla\phi\right)+Q=0 \quad \text{in } \Omega \tag{15.50}$$

with boundary conditions

$$\begin{aligned}\phi &= \bar{\phi} \quad \text{on } \Gamma_\phi\\ \mathbf{q}^{\mathrm{T}}\mathbf{n} = q_n &= \bar{q} \quad \text{on } \Gamma_q\end{aligned}$$

In the above

$$\mathbf{q}=-k\,\nabla\phi=\begin{bmatrix}q_x & q_y\end{bmatrix}^{\mathrm{T}} \tag{15.51}$$

is a flux, $\mathbf{n}$ is the outward normal to the boundary $\Gamma$, and $q_n$ is the flux normal to the boundary (see Chapters 3 and 5).

The error of the finite element solution $\hat{\phi}$ is written as

$$e=\phi-\hat{\phi} \tag{15.52}$$

The global energy norm error for domain $\Omega$ [viz. Eq. (15.11)] is

$$\|e\|=\left(\sum_{K=1}^{m}\|e\|_K^2\right)^{\frac{1}{2}} \tag{15.53}$$

where for each element $K$

$$\begin{aligned}\|e\|_K^2 &= \int_{\Omega_K}(\nabla e)^{\mathrm{T}}k\,\nabla e\,\mathrm{d}\Omega\\ &= \int_{\Omega_K}\frac{1}{k}\left[(q_x-\hat{q}_x)^2+(q_y-\hat{q}_y)^2\right]\mathrm{d}\Omega\end{aligned} \tag{15.54}$$

In what follows, we shall first discuss the explicit residual error estimator.

### 15.7.1 Explicit residual error estimator

The energy norm for an explicit residual error estimator has been derived by various authors [56,57] and has a general form

$$\|\hat{e}\|=\left(\sum_{K=1}^{m}\|\hat{e}\|_{r_K}^2\right)^{\frac{1}{2}} \tag{15.55}$$

with element contributions

$$\|\hat{e}\|_{r_K}^2 = C_1 \int_{\Omega_K} r_K^2 \, \mathrm{d}\Omega + C_2 \int_{\Gamma_K} J^2 \, \mathrm{d}\Gamma \tag{15.56}$$

where

$$r_K = -\nabla^{\mathrm{T}} \left( k \nabla \hat{\phi} \right) + Q \tag{15.57}$$

is the element interior residual and $J$ is the discontinuity in the normal flux $q_n$ at each edge of element $K$, which we call a *jump discontinuity*. For example, at an edge shared by element $K$ and its neighboring element $I$, we have

$$J = \hat{q}_{n_K} + \hat{q}_{n_I} \tag{15.58}$$

where

$$\hat{q}_{n_K} = \hat{\mathbf{q}}^{\mathrm{T}} \mathbf{n}_K \quad \text{and} \quad \hat{q}_{n_I} = \hat{\mathbf{q}}^{\mathrm{T}} \mathbf{n}_I$$

are the finite element normal fluxes.

The constants $C_1$ and $C_2$ that appear in (15.56) are mesh-dependent parameters and generally are unknown. This renders the explicit residual error estimators in the form of Eq. (15.56) less useful in practical computations.

For the particular case of constant $k$ an explicit form for $C_1$ and $C_2$ has been obtained for a four-node quadrilateral element [46,47]. This element explicit residual error estimator has the form

$$\|\hat{e}\|_{r_K}^2 = \frac{h^2}{24k} \int_{\Omega_K} r^2 \, \mathrm{d}\Omega + \frac{h}{24k} \int_{\Gamma_K} J^2 \, \mathrm{d}\Gamma \tag{15.59}$$

The derivation of Eq. (15.59) was achieved following some heuristic assumptions on the error distribution and manipulations of the element residuals. It was found that the major contribution to the error estimator is from the term involving the jump discontinuities and that the term for the element interior residual is of higher order. Therefore, in practice the form

$$\|\hat{e}\|_{r_K}^2 = \frac{h}{24k} \int_{\Gamma_K} J^2 \, \mathrm{d}\Gamma \tag{15.60}$$

is often used. Indeed, this form of the explicit residual error estimator has been the most widely used type.

In the following, we shall show, as an example, that the explicit residual error estimator of Eq. (15.60) can also be derived from a particular recovery-based error estimator.

**Example 15.2.** Deriving explicit residual error estimator

For simplicity we consider a square element $\Omega_K$ and its neighboring elements as shown in Fig. 15.15. The element contribution of the recovery-based error estimator is in the form

$$\|\hat{e}\|_K^2 = \int_{\Omega_K} \frac{1}{k} \left[ \left( q_x^\star - \hat{q}_x \right)^2 + \left( q_y^\star - \hat{q}_y \right)^2 \right] \mathrm{d}\Omega \tag{15.61}$$

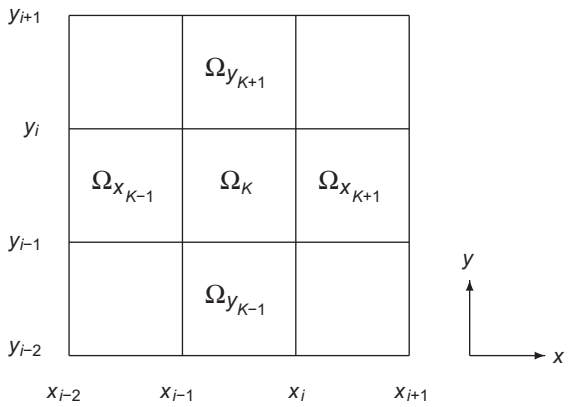



**FIGURE 15.15**
An element patch. Element $\Omega_K$ and its neighbors.

The main steps involved in the derivation of the residual error estimator of Eq. (15.60) are as follows:

1. Construct a recovered solution for element $\Omega_K$ from elements $\Omega_K$, $\Omega_{x_{K+1}}$, and $\Omega_{y_{K+1}}$, forming recovery-based error estimator $\|\hat{e}\|^2_{K_1}$.
2. Construct a recovered solution for element $\Omega_K$ from elements $\Omega_K$, $\Omega_{x_{K-1}}$, and $\Omega_{y_{K-1}}$, forming recovery-based error estimator $\|\hat{e}\|^2_{K_2}$.
3. Average $\|\hat{e}\|^2_{K_1}$ and $\|\hat{e}\|^2_{K_2}$ to obtain the final recovery-based error estimator which results in the explicit residual error estimator of Eq. (15.60).

In the first step, consider $\Omega_K$ and its two neighboring elements $\Omega_{x_{K+1}}$ and $\Omega_{y_{K+1}}$; the recovered solutions are expressed as

$$\begin{aligned} q^\star_{x_1} &= \hat{q}_{x_K} + \alpha_1 Z_x(x) \\ q^\star_{y_1} &= \hat{q}_{y_K} + \beta_1 Z_y(y) \end{aligned} \tag{15.62}$$

where $Z_x$ and $Z_y$ are linear functions in $x$ and $y$, respectively, i.e.,

$$\begin{aligned} Z_x(x) &= 1 - 2\frac{x_i - x}{h} \\ Z_y(y) &= 1 - 2\frac{y_i - y}{h} \end{aligned} \tag{15.63}$$

with $h$ the edge length of the square element

$$h = x_i - x_{i-1} = y_i - y_{i-1}$$

and $\alpha_1$, $\beta_1$ unknown parameters to be determined by a recovery process. A recovery process for $q^\star_{x_1}$ is shown in Fig. 15.16. A similar result holds for $q^\star_{y_1}$. The recovery

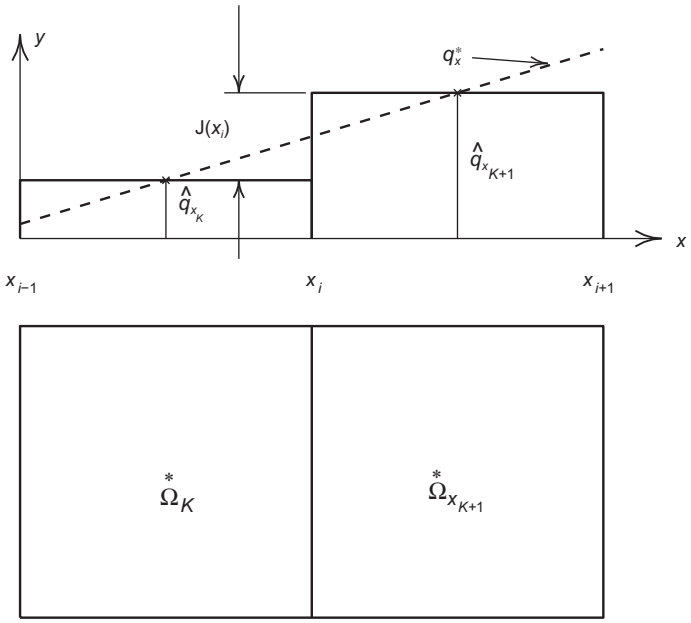



**FIGURE 15.16**
Recovered solution and jump for $\Omega_{x_K}$ and $\Omega_{x_{K+1}}$.

method of averaging is used by requiring the recovered solution to be the average of the finite element solution at the boundary of the element, i.e., at the edge shared by $\Omega_K$ and $\Omega_{x_{K+1}}$

$$q^{\star}_{x_1}(x_i) = \frac{1}{2}\left[\hat{q}_{x_K}(x_i) + \hat{q}_{x_{K+1}}(x_i)\right] \tag{15.64a}$$

and at the edge shared by element $\Omega_K$ and $\Omega_{y_{K+1}}$

$$q^{\star}_{y_1}(y_i) = \frac{1}{2}\left[\hat{q}_{y_K}(y_i) + \hat{q}_{y_{K+1}}(y_i)\right] \tag{15.64b}$$

Substituting Eq. (15.62) into Eqs. (15.64a) and (15.64b), $\alpha_1$ and $\beta_1$ have the solution

$$\begin{aligned} \alpha_1 &= \frac{1}{2Z_x(x_i)}\left(\hat{q}_{x_{K+1}}(x_i) - \hat{q}_{x_K}(x_i)\right) = -\frac{1}{2}J(x_i) \\ \beta_1 &= \frac{1}{2Z_y(y_i)}\left(\hat{q}_{y_{K+1}}(y_i) - \hat{q}_{y_K}(y_i)\right) = -\frac{1}{2}J(y_i) \end{aligned} \tag{15.65}$$

where $J(x_i)$ is the jump discontinuity along edge $x_i$ (viz. Fig. 15.16) and $J(y_i)$ is the jump discontinuity along edge $y_i$. In the above, we have used the fact that at $x_i$

$$\hat{q}_{n_K} = \hat{q}_{x_K} \quad \text{and} \quad \hat{q}_{n_{K+1}} = -\hat{q}_{x_{K+1}}$$

and at $y_i$

$$\hat{q}_{n_K} = \hat{q}_{y_K} \quad \text{and} \quad \hat{q}_{n_{K+1}} = -\hat{q}_{y_{K+1}}$$

The determined recovered solutions are now in the form of

$$\begin{aligned} q^\star_{x_1} &= \hat{q}_{x_K} - \frac{1}{2} J(x_i) Z_x \\ q^\star_{y_1} &= \hat{q}_{y_K} - \frac{1}{2} J(y_i) Z_y \end{aligned} \tag{15.66}$$

Error estimator $\|\hat{e}\|^2_{K_1}$ for element $\Omega_K$ is attained by substituting the above $q^\star_{x_1}$ and $q^\star_{y_1}$ into Eq. (15.61):

$$\|\hat{e}\|^2_{K_1} = \frac{1}{4k} \int_{\Omega_K} (J(x_i)^2 Z_x^2 + J(y_i)^2 Z_y^2)\, \mathrm{d}\Omega \tag{15.67}$$

Notice that $J(x_i)$ and $Z_y$ are only functions of $y$ and $J(y_i)$ and $Z_x$ are only functions of $x$. We have the first recovery-based error estimator for element $\Omega_K$:

$$\begin{aligned} \|\hat{e}\|^2_{K_1} &= \frac{1}{4k} \left( \int_{x_{i-1}}^{x_i} Z_x^2\, \mathrm{d}x \int_{y_{i-1}}^{y_i} J(x_i)^2\, \mathrm{d}y + \int_{y_{i-1}}^{y_i} Z_y^2\, \mathrm{d}y \int_{x_{i-1}}^{x_i} J(y_i)^2\, \mathrm{d}x \right) \\ &= \frac{h}{12k} \left( \int_{\Gamma_{y_i}} J(x_i)^2\, \mathrm{d}\Gamma + \int_{\Gamma_{x_i}} J(y_i)^2\, \mathrm{d}\Gamma \right) \end{aligned} \tag{15.68}$$

where $\Gamma_{x_i}$ denotes the limits from $x_{i-1}$ to $x_i$ and $\Gamma_{y_i}$ from $y_{i-1}$ to $y_i$. In the above we have used the following results:

$$\int_{x_{i-1}}^{x_i} Z_x^2\, \mathrm{d}x = \int_{y_{i-1}}^{y_i} Z_y^2\, \mathrm{d}y = \frac{h}{3}$$

Similarly, in the second step consider elements $\Omega_K$, $\Omega_{x_{K-1}}$, and $\Omega_{y_{K-1}}$ with the recovered solutions written as

$$\begin{aligned} q^\star_{x_2} &= \hat{q}_{x_K} + \alpha_2 Z_x \\ q^\star_{y_2} &= \hat{q}_{y_K} + \beta_2 Z_y \end{aligned} \tag{15.69}$$

To determine the unknown parameters $\alpha_2$ and $\beta_2$ we again use the recovery method of averaging and require that at the edge shared by $\Omega_K$ and $\Omega_{x_{K-1}}$

$$q^\star_{x_2}(x_{i-1}) = \frac{1}{2} \left( \hat{q}_{x_{K-1}}(x_{i-1}) + \hat{q}_{x_K}(x_{i-1}) \right) \tag{15.70}$$

and at the edge shared by element $\Omega_K$ and $\Omega_{y_{K-1}}$

$$q^\star_{y_2}(y_{i-1}) = \frac{1}{2} \left( \hat{q}_{y_{K-1}}(y_{i-1}) + \hat{q}_{y_K}(y_{i-1}) \right) \tag{15.71}$$

Following the exact procedure used in step 1, $\alpha_2$ and $\beta_2$ are solved as

$$\begin{aligned}\alpha_2 &= \frac{1}{2Z_x(x_{i-1})}\left(\hat{q}_{x_{K-1}}(x_{i-1}) - \hat{q}_{x_K}(x_{i-1})\right) = -\frac{1}{2}J(x_{i-1}) \\ \beta_2 &= \frac{1}{2Z_y(y_{i-1})}\left(\hat{q}_{y_{K-1}}(y_{i-1}) - \hat{q}_{y_K}(y_{i-1})\right) = -\frac{1}{2}J(y_{i-1})\end{aligned} \tag{15.72}$$

Here $J(x_{i-1})$ and $J(y_{i-1})$ are jump discontinuities along edges $x_{i-1}$ and $y_{i-1}$, respectively. The recovered solutions are now written as

$$\begin{aligned}q^{\star}_{x_2} &= \hat{q}_{x_K} - \frac{1}{2}J(x_{i-1})Z_x \\ q^{\star}_{y_2} &= \hat{q}_{y_K} - \frac{1}{2}J(y_{i-1})Z_y\end{aligned} \tag{15.73}$$

Substituting Eq. (15.73) into Eq. (15.61) the second recovery-based error estimator can be obtained:

$$\|\hat{e}\|^2_{K_2} = \frac{h}{12k}\left(\int_{\Gamma_{x_{i-1}}} J(y_{i-1})^2\,\mathrm{d}\Gamma + \int_{\Gamma_{y_{i-1}}} J(x_{i-1})^2\,\mathrm{d}\Gamma\right) \tag{15.74}$$

Finally, to include the influence of all the neighboring elements, it is only natural to let the recovery error estimator for element $\Omega_K$ be taken as the average of $\|\hat{e}\|^2_{K_1}$ and $\|\hat{e}\|^2_{K_2}$, i.e.,

$$\|\hat{e}\|^2_K = \frac{1}{2}\left(\|\hat{e}\|^2_{K_1} + \|\hat{e}\|^2_{K_2}\right) \tag{15.75}$$

Substituting the expressions of $\|\hat{e}\|^2_{K_1}$ and $\|\hat{e}\|^2_{K_2}$, we have

$$\begin{aligned}\|\hat{e}\|^2_K &= \frac{h}{24k}\left(\int_{\Gamma_{y_i}} J(x_i)^2\,\mathrm{d}\Gamma + \int_{\Gamma_{x_i}} J(y_i)^2\,\mathrm{d}\Gamma + \int_{\Gamma_{x_{i-1}}} J(y_{i-1})^2\,\mathrm{d}\Gamma\right. \\ &\quad \left. + \int_{\Gamma_{y_{i-1}}} J(x_{i-1})^2\,\mathrm{d}\Gamma\right) \\ &= \frac{h}{24k}\int_{\Gamma_K} J^2\,\mathrm{d}\Gamma\end{aligned} \tag{15.76}$$

This is exactly the explicit residual error estimator of Eq. (15.60).

We have demonstrated, by the above example, that the explicit residual error estimator for a bilinear element can be derived from a recovery-based error estimator using averaging as the recovery method. For more general discussions on the relationship between recovery-based error estimators and explicit residual error estimators we refer to Refs. [58,59]; for discussion on the equivalence of recovery-based error estimators with certain explicit residual estimators we refer to Refs. [56,60]; and for using a recovery method in the computation of explicit residual error estimator the reader is referred to Ref. [61]. We shall now turn our attention to how to use a recovery method in the computation of implicit residual error estimators.

### 15.7.2 Implicit residual error estimators

The computation of implicit residual error estimators requires solving an auxiliary boundary value problem with residuals as input data for the approximation error. Among all the existing implicit residual error estimators, the *equilibrated residual estimator* has been shown to be the most robust [62–64].

In what follows we restrict our discussion to the equilibrated residual error estimator for the model problem of Eq. (15.50). The construction of an equilibrated residual error estimator for other problems, such as elasticity problems, proceeds in an analogous manner [65].

We again consider an interior element $K$. Substituting the finite element solution $\hat{\phi}$ into Eq. (15.50) results in, for element K,

$$-\nabla^{\mathrm{T}}(k\,\nabla\hat{\phi}) + Q = r_K \quad \text{in } \Omega_K \tag{15.77}$$

and

$$-(k\,\nabla\hat{\phi})^{\mathrm{T}}\mathbf{n} = \hat{q}_n \quad \text{on } \Gamma_K$$

Subtracting the above equations from Eq. (15.50) gives an element boundary value problem for error $e$ as

$$-\nabla^{\mathrm{T}}\left(k\,\nabla e\right) + r_K = 0 \quad \text{in } \Omega_K \tag{15.78}$$

with boundary condition

$$-\left(k\,\nabla e\right)^{\mathrm{T}}\mathbf{n} = q_n - \hat{q}_n \quad \text{on } \Gamma_K$$

We notice immediately that Eq. (15.78) is not solvable because the exact normal flux $q_n$ on the element boundary is in general unknown. A natural strategy to overcome this difficulty is to replace the exact solution by a recovered solution $q_n^\star$ which can be computed from the finite element flux in element $K$ and its surrounding elements (as we did in the computation of a recovery-based error estimator).

We can now write the Neumann boundary value problem for the element error as

$$-\nabla^{\mathrm{T}}\left(k\,\nabla e\right) + r_K = 0 \quad \text{in } \Omega_K \tag{15.79}$$

with boundary condition

$$-\left(k\,\nabla e\right)^{\mathrm{T}}\mathbf{n} = q_n^\star - \hat{q}_n \quad \text{on } \Gamma_K$$

An approximate solution to the above equation for $\hat{e}$ appearing in the energy norm, $\|\hat{e}\|_K$, defines an *implicit element residual error estimator*.

Various recovery techniques can be used to compute the normal flux $q_n^\star$ [42,50,51]. However, the Neumann problem of Eq. (15.79) will have a solution if $q_n^\star$ is computed such that the residuals satisfy the *equilibrium condition*

$$\int_{\Omega_K} N_c r_K \,\mathrm{d}\Omega + \int_{\Gamma_K} N_c\left(q_n^\star - \hat{q}_n\right)\mathrm{d}\Gamma = 0 \tag{15.80}$$

where $N_c$ is the shape function for node $c$ of element $K$. Although $N_c$ can be a shape function of any order, a linear shape function seems to be the most practical in the following computation.

The residuals which satisfy Eq. (15.80) are said to be equilibrated, thus the recovered solution $q_n^\star$ satisfying Eq. (15.80) is called the equilibrated flux. An error estimator which uses the solution of the element error problem of Eq. (15.80) with the equilibrated flux $q_n^\star$ is termed an *equilibrated residual error estimator*. This type of residual error estimator was first introduced by Bank and Weiser [50] and later more rigorously pursued by Ainsworth and Oden [54].

It is apparent that the most important step in the computation of the equilibrated residual error estimator is to achieve the recovered normal flux $q_n^\star$ which satisfies Eq. (15.80). Once $q_n^\star$ is determined, the error problem Eq. (15.79) can be readily solved for an element following a standard finite element procedure. Therefore we shall focus our attention on the recovery process.

The technique of recovering normal flux by equilibrated residuals was first proposed by Ladevéze et al. [42] and Kelly [49], and followed by Ohtsubo and Mitamura [66]. A different version of this technique was later used by Ainsworth and Oden [57] where a detailed description of the application to various mesh patterns can be found. Here we shall consider a typical element patch of triangles as shown in Fig. 15.17.

To determine $q_n^\star$, we first substitute the residual $r_K$ of Eq. (15.77) into Eq. (15.80) and upon integrating by parts obtain

$$\int_{\Omega_K} N_c Q \, \mathrm{d}\Omega + \int_{\Omega_K} \boldsymbol{\nabla}^{\mathrm{T}} N_c \left( k \, \boldsymbol{\nabla} \hat{\phi} \right) \mathrm{d}\Omega + \int_{\Gamma_K} N_c q_n^\star \, \mathrm{d}\Gamma = 0 \tag{15.81}$$

Let the recovered interelement boundary normal flux have the form

$$q_n^\star = \frac{1}{2} \left( \hat{\mathbf{q}}_K + \hat{\mathbf{q}}_I \right)^{\mathrm{T}} \mathbf{n}_s + Z_s \tag{15.82}$$

where the first term on the right-hand side is the average of the normal flux of the finite element solution from element $K$ and its neighbor element $I$ as shown in Fig. 15.18; $\mathbf{n}_s$ is the outward normal on the edge $s$ of element $K$; and $Z_s$ is a linear function defined on the edge $S$, shared by elements $K$ and $I$, with end nodes $a$ and $b$ and

$$Z_s = \widehat{N}_a^s a_a^s + \widehat{N}_b^s a_b^s \tag{15.83}$$

where $\widehat{N}_a^s$, $\widehat{N}_b^s$ are the dual shape functions introduced in Section 11.2.1 and in the present case are given by

$$\widehat{N}_a^s = \frac{2}{|h_s|} \left[ 2 \, N_a^s - N_b^s \right] \quad \text{and} \quad \widehat{N}_b^s = \frac{2}{|h_s|} \left[ 2 \, N_b^s - N_a^s \right] \tag{15.84}$$

where $N_a^s$ and $N_b^s$ are the linear shape functions defined for edge $S$ and $|h_s|$ is the length of the edge. The unknown parameters $a_a^s$ and $a_b^s$ are to be determined from the residual equilibrium Eq. (15.81).

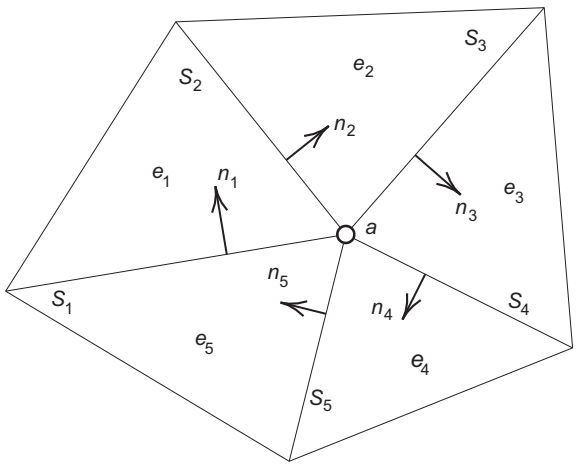



**FIGURE 15.17**
Typical patch with interior vertex node $a$ showing a local numbering of elements $e_i$ and edges $S_i$.

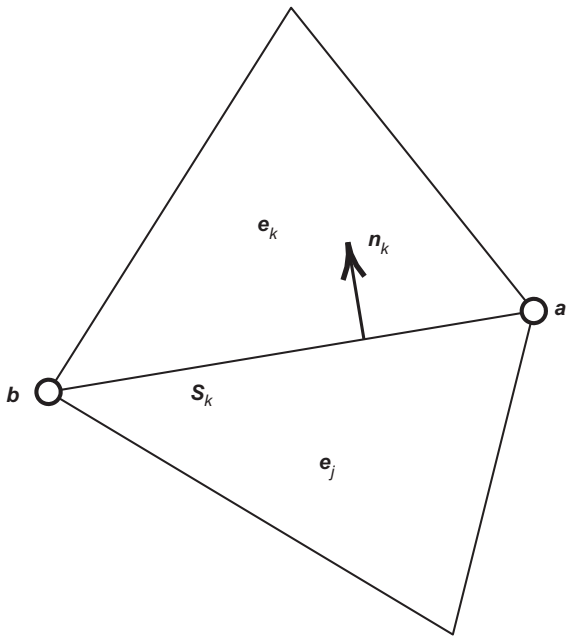



**FIGURE 15.18**
Element interface for equilibrated flux recovery.

It is easy to verify that

$$\int_s N_a^s \widehat{N}_b^s \, \mathrm{d}\Gamma = \delta_{ab} \tag{15.85}$$

where $\delta_{ab}$ is the Kronecker delta given by

$$\delta_{ab} = \begin{cases} 1, & a = b \\ 0, & a \neq b \end{cases} \tag{15.86}$$

Let $a$ denote a typical interior vertex node. Choose $N_c = N_a$ in Eq. (15.81) and consider the element patch associated with the linear shape function $N_a$ as shown in Fig. 15.17. Assign element 1 as element $K$ in the patch (i.e., $e_1 = e_K$).

It is obvious that $N_a$ is zero at the exterior boundary of the element patch. A local numbering for the elements and edges connected to node $a$ in the patch is given. The edge normals shown here are the result of a global edge orientation.

For element $e_1$ in the patch, substituting Eq. (15.82) into Eq. (15.81) for each edge and observing that $N_a$ is nonzero only on the edges $s_1$ and $s_2$ and in the $n_1$ and $n_2$ directions we have

$$\begin{aligned} &\int_{\Omega_{e_1}} N_a Q \, \mathrm{d}\Omega + \int_{\Omega_{e_1}} \left(\nabla N_a\right)^{\mathrm{T}} (k \nabla \hat{\phi}) \mathrm{d}\Omega - \int_{s_1} \frac{1}{2} N_a (\hat{\mathbf{q}}_{e_1} + \hat{\mathbf{q}}_{e_5})^{\mathrm{T}} \mathbf{n}_{s_1} \, \mathrm{d}\Gamma \\ &\quad + \int_{s_2} \frac{1}{2} N_a (\hat{\mathbf{q}}_{e_1} + \hat{\mathbf{q}}_{e_2})^{\mathrm{T}} \mathbf{n}_{s_2} \, \mathrm{d}\Gamma - \int_{s_1} N_a Z_{s_1} \, \mathrm{d}\Gamma + \int_{s_2} N_a Z_{s_2} \, \mathrm{d}\Gamma = 0 \end{aligned} \tag{15.87}$$

where a boundary integral takes a negative sign if the edge normal shown in Figs. 15.17 and 15.18 points inward to the element.

Let $f_{e_1}$ denote the first four computable terms of the above equation and notice that [using Eq. (15.85)]

$$\int_{s_1} N_a Z_{s_1} \, \mathrm{d}\Gamma = \int_{s_1} N_a \left(\widehat{N}_a^s a_a^{s_1} + \widehat{N}_b^s a_b^{s_1}\right) \mathrm{d}\Gamma = a_a^{s_1} \tag{15.88a}$$

and

$$\int_{s_2} N_a Z_{s_2} \, \mathrm{d}\Gamma = \int_{s_2} N_a \left(\widehat{N}_a^s a_a^{s_2} + \widehat{N}_b^s a_b^{s_2}\right) \mathrm{d}\Gamma = a_a^{s_2} \tag{15.88b}$$

Equation (15.87) now becomes

$$-a_a^{s_1} + a_a^{s_2} = -f_{e_1} \tag{15.89}$$

Similar equations result for element $e_2$ to $e_5$ of the patch in Fig. 15.17 giving the equation set

$$\mathbf{Aa} = \mathbf{f} \tag{15.90}$$

where

$$\mathbf{A} = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \\ 1 & 0 & 0 & 0 & -1 \end{bmatrix}$$

$$\mathbf{a} = \left[ a_a^{s1} \; a_a^{s2} \; a_a^{s3} \; a_a^{s4} \; a_a^{s5} \right]^{\mathrm{T}}$$

and

$$\mathbf{f} = -\left[ f_{e1} \; f_{e2} \; f_{e3} \; f_{e4} \; f_{e5} \right]^{\mathrm{T}}$$

It is easy to verify that these equations are linearly dependent but have solutions determined up to an arbitrary constant. A procedure to obtain an *optimal* particular solution is described as follows [43,50,57]. First, a particular solution $\mathbf{a}_0$ of Eq. (15.90) is found by choosing, for example, $a_a^{s5} = 0$. Second, the corresponding homogeneous equation

$$\mathbf{A}\mathbf{b} = \mathbf{0} \tag{15.91}$$

with $\mathbf{b} = \left[b_1, b_2, b_3, b_4, b_5\right]^{\mathrm{T}}$ is solved for a nonzero particular solution with the choice of, corresponding to $a_a^{s5}$, $b_5 = 1$. It is easy to verify that $b_i$ is either 1 or $-1$ due to the structure of $\mathbf{A}$. In the element patch considered here $\mathbf{b} = \left[1, 1, 1, 1, 1\right]^{\mathrm{T}}$.

The final particular solution of Eq. (15.90) takes the form of

$$\mathbf{a} = \mathbf{a}_0 + \gamma \mathbf{b} \tag{15.92}$$

where the constant $\gamma$ is determined by the minimization of

$$\Pi = \mathbf{a}^{\mathrm{T}}\mathbf{a} \tag{15.93}$$

The minimization condition gives

$$\gamma = -\frac{\mathbf{b}^{\mathrm{T}}\mathbf{a}_0}{\mathbf{b}^{\mathrm{T}}\mathbf{b}} \tag{15.94}$$

The solution gives the nodal value $a_a^{si}$ at node $a$ for each connected edge of the element patch.

Boundary nodes and their related element patches can be considered in the same fashion except that we can take $q_n^\star = \bar{q}_n$, the known flux, for the element edge being part of $\Gamma_q$. For edges coincident with $\Gamma_\phi$, we let the first term on the right-hand side of Eq. (15.82) be zero. By considering each vertex node of the mesh and its associated element patch, we will be able to determine $a_a^s$ and $a_b^s$ in Eq. (15.83) for every edge, thus the recovered normal flux $q_n^\star$ in the form defined by Eq. (15.82) on the element boundary is achieved. The procedure described above for recovering the normal flux is a *recovery by element residuals*.

We note that the nonuniqueness of the solution of Eq. (15.90) represents the nonuniqueness of the equilibrium status of the element residuals. The choice of the arbitrary constant in solving Eq. (15.90) will certainly affect the accuracy of the recovered solution $q_n^\star$, and therefore the accuracy of the error estimator.

With $q_n^\star$ determined, the local error problem Eq. (15.79) is usually solved by a higher order (e.g., $p+1$ or even $p+2$) approximation. The solution of the problem is then employed in the equilibrated residual error estimator $\|\hat{e}\|_{r_K}$. The global error estimator $\|\hat{e}\|$ is obtained through Eq. (15.55). The global error estimator has been shown to be an upper bound of the exact error [54], although it is not a trivial task to prove its convergence.

We have shown here that a proper recovery method is the key to the computation of equilibrated residual error estimators. Indeed, carefully chosen recovery methods are very important in the computation of all the implicit residual error estimators. Numerical performance of residual-based error estimators was tested by Babuška et al. [62–64] and Carstensen et al. [67] and compared with that of recovery-based error estimators.

## 15.8 Asymptotic behavior and robustness of error estimator: The Babuška patch test

It is well known that elements in which polynomials of order $p$ are used to represent the unknown $\mathbf{u}$ will reproduce exactly any problem for which the exact solution is also defined by such a polynomial. Indeed the verification of this behavior is an essential part of the "patch test" which has to be satisfied by all elements to ensure convergence, as we have discussed in Chapter 8.

Thus if we are attempting to determine the error in a general smooth solution we will find that this error is dominated by terms of order $p+1$. The response of any patch to an exact solution of order $p+1$ will therefore determine the asymptotic behavior when both the size of the patch and of all the elements tends to zero. If the patch is assumed to be one of a repeatable kind, its behavior when subjected to an exact solution of order $p+1$ will give the exact asymptotic error of the finite element solution. Thus, any estimator can be compared with this exact value and the asymptotic effectivity index can be established. Figure 15.19 shows such a repeatable patch of quadrilateral elements which evaluate the performance of the error estimators for quite irregular meshes.

We have indeed shown how true superconvergent behavior reproduces exactly such higher order solutions and thus leads to an effectivity index of unity in the asymptotic limit. In the papers presented by Babuška et al. [62–64] the procedure of dealing with such repeatable patches for various patterns of two-dimensional elements is developed. Thus, if we are interested in solving the differential equation

$$\mathcal{L}u + f = 0 \tag{15.95}$$

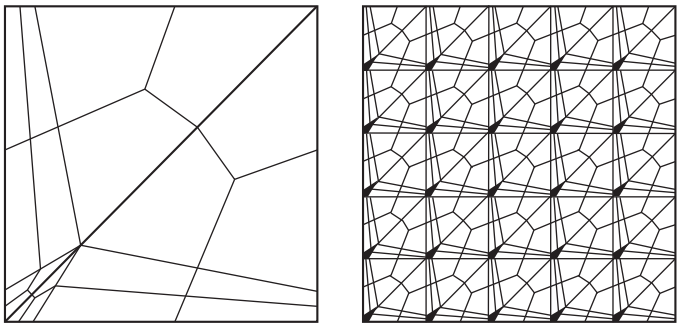


**FIGURE 15.19**
Repeating patch of irregular and quadrilateral elements.

where $\mathcal{L}$ is a linear differential operator of order $2p$, we consider *exact solutions* (harmonic solutions) to the homogeneous equation ($f = 0$) of the form

$$u_{\text{ex}} = \sum_m a_m x^m y^n = \mathbf{P}(x, y)\mathbf{a}, \quad n = p + 1 - m \tag{15.96}$$

The boundary conditions are taken as

$$u_{\text{ex}}|_{x+L_x} = u_{\text{ex}}|_x \quad \text{and} \quad u_{\text{ex}}|_{y+L_y} = u_{\text{ex}}|_y \tag{15.97}$$

where $L_x$ and $L_y$ are periodic distances in the $x$ and $y$ directions, respectively (viz. repeatability, Section 2.2.5). In general, the individual terms of Eq. (15.96) do not satisfy the differential equation and it is necessary to consider linear combinations in terms of the parameters in $\mathcal{L}$ as

$$\mathbf{a}' = \mathbf{T}\mathbf{a} \tag{15.98}$$

This solution serves as the basis for conducting a patch test in which the boundary conditions are assigned to be periodic and to prevent constant changes to $u$.[1] The correct constant value may be computed from

$$\int_{\text{patch}} \left(\mathbf{N}\tilde{\mathbf{u}} + C\right) \, \mathrm{d}\Omega = \int_{\text{patch}} u_{\text{ex}} \, \mathrm{d}\Omega \tag{15.99}$$

To compute upper and lower bounds ($\theta_U$ and $\theta_L$) on the possible effectivity indices of the error estimators, all possible combinations of the harmonic solution must be considered. This may be achieved by constructing an error norm of the solutions, for example the $L_2$ norm of the flux (or stress)

$$\|\mathbf{e}_q\|_{L_2}^2 = \int_{\text{patch}} \left(\mathbf{q}_{\text{ex}} - \hat{\mathbf{q}}\right)^{\text{T}} \left(\mathbf{q}_{\text{ex}} - \hat{\mathbf{q}}\right) \, \mathrm{d}\Omega = \left(\mathbf{a}'\right)^{\text{T}} \mathbf{T}^{\text{T}} \mathbf{E}_{\text{ex}} \mathbf{T}\mathbf{a}' \tag{15.100}$$

[1] For elasticity-type problems the periodic boundary conditions prevent rigid rotations.

and

$$\|\hat{\mathbf{e}}_q\|_{L_2}^2 = \int_{\text{patch}} \left(\mathbf{q}^\star - \hat{\mathbf{q}}\right)^{\mathrm{T}} \left(\mathbf{q}^\star - \hat{\mathbf{q}}\right) \mathrm{d}\Omega = \left(\mathbf{a}'\right)^{\mathrm{T}} \mathbf{T}^{\mathrm{T}} \mathbf{E}^\star \mathbf{T} \mathbf{a}' \tag{15.101}$$

and solving the eigenproblem

$$\mathbf{T}^{\mathrm{T}} \mathbf{E}^\star \mathbf{T} \mathbf{a}' = \theta^2 \mathbf{T}^{\mathrm{T}} \mathbf{E}_{\text{ex}} \mathbf{T} \mathbf{a}' \tag{15.102}$$

to determine the minimum (lower bound) and maximum (upper bound) effectivity indices. Further details of the process summarized here are given in Boroomand and Zienkiewicz [34,35] and by Zienkiewicz et al. [68].

These bounds on the effectivity index are very useful for comparing various error estimators and their behavior for different mesh and element patterns. However, a single parameter called the *robustness index* has also been devised [62] and is useful as a guide to the robustness of any particular estimator

$$R = \max\left( |1 - \theta_L| + |1 - \theta_U|, \left|1 - \tfrac{1}{\theta_L}\right| + \left|1 - \tfrac{1}{\theta_U}\right| \right) \tag{15.103}$$

A large value of this index obviously indicates a poor performance. Conversely the best behavior is that in which

$$\theta_L = \theta_U = 1 \tag{15.104}$$

and this gives

$$R = 0 \tag{15.105}$$

In the series of tests reported in Refs. [62–64] various estimators have been compared. Table 15.1 shows the highest robustness index value of an equilibrating residual-based error estimator, ERpB, and the SPR recovery error estimator for a set of particular patches of triangular elements [62].

This performance comparison is quite remarkable and it seems that in all the tests quoted by Babuška et al. [62–64] and summarized in Babuška and Strouboulis [69] the recovery estimator using SPR performs best. Indeed we shall observe that in many cases of regular subdivision, when full superconvergence occurs the ideal, asymptotically exact solution characterized by $R = 0$ will be obtained.

In Table 15.2 we show some results obtained for regular meshes of triangles and rectangles with linear and quadratic elements. In the rectangular elements used for problems of heat conduction type, superconvergent points are exact and the ideal result is obtained for both linear and quadratic elements. It is surprising that this also occurs in elasticity where the proof of superconvergent points is lacking [since for

**Table 15.1** Robustness Index $R$ for Equilibrated Residual ERpB and SPR (ZZ-Discrete) Estimators for a Variety of Anisotropic Situations and Element Patterns, $p = 2$

| Estimator Form | Robustness Index—$R$ |
|---|---|
| ERpB | 10.21 |
| SPR (ZZ-discrete) | 0.02 |

**Table 15.2** Effectivity Bounds and Robustness of SPR and REP Recovery Estimator for Regular Meshes of Triangles and Rectangles with Linear and Quadratic Shape Function (Applied to Heat Conduction and Elasticity Problems). Aspect Ratio = Length ($L$)/Height ($H$) of Elements in Patch Tested

| | SPR | | | REP | | |
|---|---|---|---|---|---|---|
| **Aspect Ratio $L/H$** | $\theta_L$ | $\theta_U$ | $R$ | $\theta_L$ | $\theta_U$ | $R$ |
| **Linear Triangles and Rectangles (Heat Conduction/Elasticity)** | | | | | | |
| 1/1 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| 1/2 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| 1/4 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| 1/8 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| 1/16 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| 1/32 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| 1/64 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| **Quadratic Rectangles (Heat Conduction)** | | | | | | |
| 1/1 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| 1/2 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| 1/4 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| 1/8 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| 1/16 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| 1/32 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| 1/64 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 1.0000 | 0.0000 |
| **Quadratic Rectangles (Elasticity)** | | | | | | |
| 1/1 | 1.0000 | 1.0000 | 0.0000 | 0.9991 | 1.0102 | 0.0111 |
| 1/2 | 1.0000 | 1.0000 | 0.0000 | 0.9991 | 1.0181 | 0.0189 |
| 1/4 | 1.0000 | 1.0000 | 0.0000 | 0.9991 | 1.0136 | 0.0145 |
| 1/8 | 1.0000 | 1.0000 | 0.0000 | 0.9991 | 1.0030 | 0.0039 |
| 1/16 | 1.0000 | 1.0000 | 0.0000 | 0.9968 | 1.0001 | 0.0033 |
| 1/32 | 1.0000 | 1.0000 | 0.0000 | 0.9950 | 1.0000 | 0.0050 |
| 1/64 | 1.0000 | 1.0000 | 0.0000 | 0.9945 | 1.0000 | 0.0055 |
| **Quadratic Triangles (Elasticity)** | | | | | | |
| 1/1 | 0.9966 | 1.0929 | 0.0963 | 0.9562 | 1.0503 | 0.0940 |
| 1/2 | 0.9966 | 1.0931 | 0.0965 | 0.9559 | 1.0481 | 0.0923 |
| 1/4 | 0.9967 | 1.0937 | 0.0970 | 0.9535 | 1.0455 | 0.0924 |
| 1/8 | 0.9967 | 1.0943 | 0.0976 | 0.9522 | 1.0603 | 0.1081 |
| 1/16 | 0.9966 | 1.0946 | 0.0980 | 0.9518 | 1.0666 | 0.1148 |
| 1/32 | 0.9966 | 1.0947 | 0.0981 | 0.9517 | 1.0684 | 0.1167 |
| 1/64 | 0.9965 | 1.0947 | 0.0982 | 0.9516 | 1.0688 | 0.1172 |

$\nu > 0$ $\mathbf{A}$ in (15.17) is not diagonal]. Further, the REP procedure also seems to yield superconvergence except for elasticity with quadratic elements.

For regular meshes of quadratic triangles generally superconvergence is not expected and it does not occur for either heat conduction or elasticity problems.

However, the robustness index has very small values ($R < 0.10$ for SPR and $R < 0.12$ for REP) and these estimators are therefore very accurate.

In Fig. 15.20 and Table 15.3 very irregular meshes of triangular and quadrilateral elements are analyzed in repeatable patterns. It is of course not possible to present here all tests conducted by the effectivity patch test. The results shown are, however, typical—others are given in Ref. [34]. It is interesting to observe that the performance measured by the robustness index on quadrilateral elements is always superior to that measured on triangles.

## 15.9 Error bounds and error estimates for quantities of interest

Although we have shown that excellent estimators of errors exist today, many are striving to know that these estimators are not only close but that they are *bounded*. The strain energy was one of the first quantities in which bounds could be established. Here the classical work of Fraeijs de Veubeke in the mid-1960s is of vital importance [70,71].

It was quickly realized by Fraeijs de Veubeke that the standard (displacement) procedures from which structural analysis usually started would provide a lower bound of the strain energy contained in the structure and thus always underestimated the value of strain energy. He therefore sought procedures which could solve the same structural problem by concentrating on so-called complementary energy which would allow the establishment of solutions in which the strain energy would always be overestimated. This process proved very difficult as equilibrating solutions have to be established at all stages. A possible way, useful for many two-dimensional problems, was suggested in Ref. [72] in which stress functions and the slab analogy were used. Nevertheless the methodology never succeeded as a practical way of providing the bounds of strain energy in an actual analysis.

Much later when the residual method was being applied to determine error in structural analysis it was realized that once again opportunity existed for establishing bounds. The residuals are nothing else but a measure by which the numerical solution fails to satisfy the differential equations of the problem. By using a local solution, generally based on a few elements or even a single element, the error can be estimated locally and the total error obtained by combining these estimates for all elements. This solution can be carried out in a number of ways [57,69], such as the procedure discussed in Section 15.7.2 for the computation of the equilibrated residual error estimator. As many local problems are solved, it is assumed the total error is a combination of local solutions. The upper bound property of the residual error estimators is a direct consequence of the equilibrated stress (flux) resulting from the satisfaction of the equilibrium conditions of the residuals. It seems that one of the first to extend the concept of establishing upper bounds is Kelly [49] and subsequent work [73]. He endeavored to obtain solutions for the residual placed as a load with equilibrating methodologies. Two similar alternative approaches, though

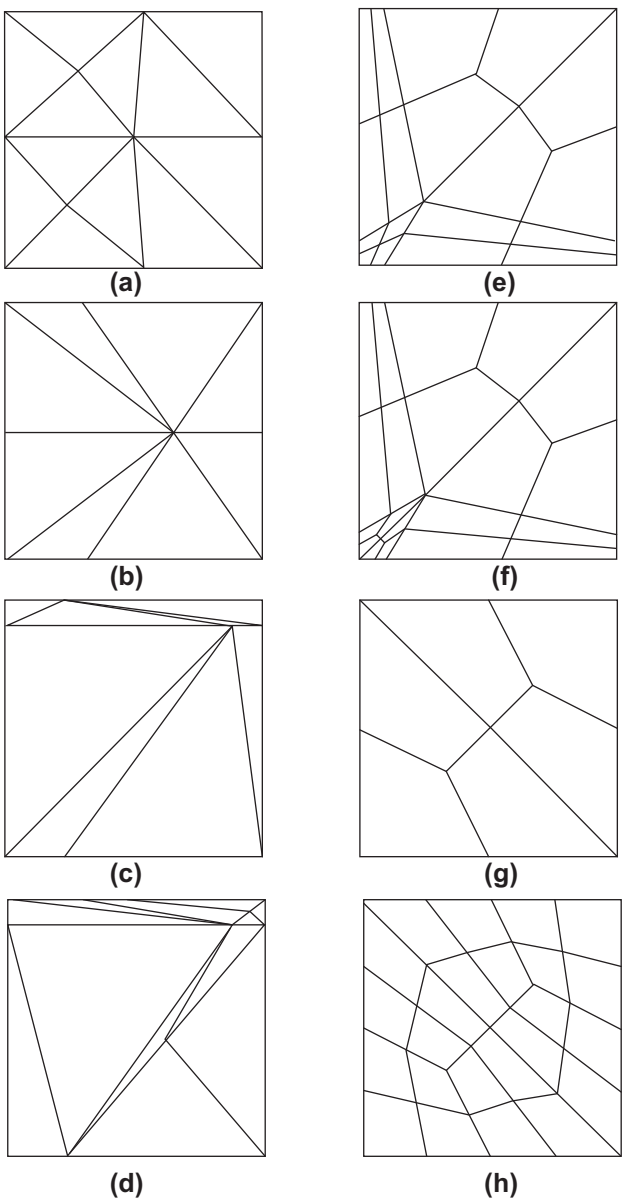



**FIGURE 15.20**
Repeating patch types (a) through (h). See Table 15.3 for results.

**Table 15.3** Effectivity Bounds and Robustness of SPR and REP Recovery Estimator for Irregular Meshes of Triangles (a, b, c, d) and Quadrilaterals (e, f, g, h)

| | SPR | | | REP | | |
|---|---|---|---|---|---|---|
| **Mesh Pattern** | $\theta_L$ | $\theta_U$ | $R$ | $\theta_L$ | $\theta_U$ | $R$ |
| **Linear Elements (Heat Conduction)** | | | | | | |
| a | 0.9626 | 1.0054 | 0.0442 | 0.9709 | 1.0145 | 0.0443 |
| b | 0.9715 | 1.0156 | 0.0447 | 0.9838 | 1.0167 | 0.0329 |
| c | 0.9228 | 1.4417 | 0.5189 | 0.8938 | 1.8235 | 0.9297 |
| d | 0.8341 | 1.2027 | 0.3685 | 0.9463 | 1.9272 | 0.9810 |
| e | 0.9943 | 1.0175 | 0.0232 | 0.9800 | 1.0589 | 0.0789 |
| f | 0.9969 | 1.0152 | 0.0183 | 0.9849 | 1.0582 | 0.0733 |
| g | 0.9987 | 1.0175 | 0.0188 | 0.9987 | 1.0175 | 0.0188 |
| h | 0.9991 | 1.0068 | 0.0077 | 0.9979 | 1.0062 | 0.0083 |
| **Linear Elements (Elasticity)** | | | | | | |
| a | 0.9404 | 1.0109 | 0.0741 | 0.9468 | 1.0148 | 0.0707 |
| b | 0.8869 | 1.0250 | 0.1520 | 0.9392 | 1.0275 | 0.0915 |
| c | 0.8550 | 1.6966 | 0.8415 | 0.8037 | 2.0522 | 1.2486 |
| d | 0.7945 | 1.2734 | 0.4788 | 0.7576 | 1.9416 | 1.1840 |
| e | 0.9946 | 1.0247 | 0.0301 | 0.9579 | 1.0508 | 0.0928 |
| f | 1.0038 | 1.0281 | 0.0318 | 0.9612 | 1.0467 | 0.0855 |
| g | 0.9959 | 1.0300 | 0.0341 | 0.9960 | 1.0298 | 0.0338 |
| h | 0.9972 | 1.0139 | 0.0168 | 0.9965 | 1.0122 | 0.0157 |
| **Quadratic Elements (Heat Conduction)** | | | | | | |
| a | 0.9443 | 1.0295 | 0.0877 | 0.9339 | 1.0098 | 0.0805 |
| b | 0.8146 | 1.0037 | 0.2313 | 0.9256 | 1.0028 | 0.0832 |
| c | 0.7640 | 1.0486 | 0.3000 | 0.9559 | 1.2229 | 0.2670 |
| d | 0.8140 | 1.0141 | 0.2423 | 0.9091 | 1.2808 | 0.3717 |
| e | 0.9762 | 1.0053 | 0.0296 | 0.9901 | 1.0177 | 0.0276 |
| f | 0.9691 | 1.0045 | 0.0363 | 0.9901 | 1.0322 | 0.0421 |
| g | 0.9692 | 1.0004 | 0.0322 | 0.9833 | 1.0024 | 0.0195 |
| h | 0.9906 | 1.0113 | 0.0207 | 1.0045 | 1.0261 | 0.0307 |
| **Quadratic Elements (Elasticity)** | | | | | | |
| a | 0.9144 | 1.0353 | 0.1277 | 0.9197 | 1.0244 | 0.1111 |
| b | 0.7302 | 1.0355 | 0.4038 | 0.8643 | 1.0346 | 0.1905 |
| c | 0.7556 | 1.1024 | 0.4163 | 0.8387 | 1.2422 | 0.4035 |
| d | 0.7624 | 1.0323 | 0.3430 | 0.8244 | 1.2632 | 0.4388 |
| e | 0.9702 | 1.0102 | 0.0408 | 0.9682 | 1.0058 | 0.0386 |
| f | 0.9651 | 1.0085 | 0.0446 | 0.9749 | 1.0286 | 0.0537 |
| g | 0.9457 | 1.0115 | 0.0688 | 0.9807 | 1.0125 | 0.0321 |
| h | 0.9852 | 1.0141 | 0.0290 | 0.9996 | 1.0522 | 0.0526 |

stemming from completely different origins, were proposed by Ladevèze [42,74] and almost simultaneously by Bank and Weiser [50]. These ideas were later adopted by Ainsworth and Oden forming much of the basis to their book [57]. Different approaches for constructing upper bounds of the error can also be found in the work of Cottereau et al. [75]. Upper bounds can also be established for recovery-based error estimators by enforcing equilibrium in the recovery procedures. One such procedure was proposed by Díez et al., which resulted in accurate upper bounds of the error [29].

It is worth noting that to be useful practically the error estimator should provide a tight bound with an effectivity index close to unity. The methodologies so far produced are very effective when the basic quantity of interest is a simple one, such as strain energy or the energy norm. However, when the quantity of interest is more localized and if for instance it is the value of displacements or stresses in a part of the structure, rather than an overall measure of stresses as it is in the case of energy, different procedures arise and a simple examination of energy errors does not suffice or it is not very selective in showing how to obtain the solution quantity of engineering or scientific interest. For this reason much effort has been given in recent years to discussing the possibilities of computing bounds for such quantities of interest, estimating errors in the approximate solutions of the quantities, and developing adaptive strategies by which localized goals of analysis can be solved with specified accuracy. Much of the recent work in this field concentrates on such methodologies.

The first to give attention to the possible extension of norms to other quantities of interest appeared in a series of papers presented by Babuška and Miller [76–78]. These papers laid the foundations for much of the work continued some 10 years later [79–90] and which today still occupies much interest [91–97].

To estimate the error in a quantity of interest, the key device is duality [79–83]. The dual problem may be solved on the same mesh as the original problem but now with the quantity of interest given as loads for solution that will be used in estimating the error. Such dual problems may be identified with different names: adjoint problems and extraction problems are two that are used. A variety of methodologies have been proposed to estimate the error of the quantity of interest [81,83,95,98,99]. In a series of papers Oden and Prudhomme described how such error estimates can be given in terms of the energy norm estimates of the errors in the numerical approximation of both the original problem and its dual problem. They also show how to compute upper and lower bounds for the error estimates [83,87,90].

It has been found that to achieve error estimates with effectivity index close to unity, the solution of the dual problem must have a higher order accuracy than the finite element approximation used for the original problem [98]. Different approaches were investigated to obtain the higher order approximation for the dual problem. Some involve using recovery methods to post-process the lower order approximation, others require solving the dual problem with a higher order element [98,99,95–97]. Clearly, more research is needed to develop reliable and computationally efficient error estimates for a quantity of interest.

## 15.10 Which errors should concern us?

In this chapter we have shown how various recovery procedures can accurately estimate the overall error of the finite element approximation and thus provide a very accurate error estimating method. We have also shown that estimators based on SPR recovery are superior to those based on residual computation. The error estimation discussed here concerns however only the original solution and if the user takes advantage of the recovered values a much better solution is already available. In the next chapter we shall be concerned with adaptivity processes which are aimed at reduction of the original finite element error. Here again we shall show the excellent values of the effectivity index which can be obtained with SPR-type methods on examples for which an "exact" solution is available from very fine mesh computations. What perhaps we should also be concerned with are the errors remaining in the recovered solutions, if indeed these are to be made use of. This problem is still unsolved and at the moment all the adaptive methods simply aim at the reduction of various norms of error in the finite element solution directly provided.

## 15.11 Problems

**15.1** Let the assumed stress for Example 15.1 in Section 15.4.1 be given as

$$\sigma_i^\star = \begin{bmatrix} 1 & (x-x_1) & (y-y_1) & (x-x_1) & (y-y_1) \end{bmatrix} \begin{Bmatrix} \bar{a}_1 \\ \bar{a}_2 \\ \bar{a}_3 \\ \bar{a}_4 \end{Bmatrix}$$

Compute the recovered stress and compare the result with that of Example 15.1. Is this result superconvergent?

**15.2** Show all the relations necessary to extend the SPR algorithm to three-dimensional elastic problems.
What is the expression for $\sigma_i^\star$ that should be used for eight-node hexahedral elements?

**15.3** Program development project: Implement the SPR procedure in the solver system developed in Problem 7.20 and subsequent chapters. Assume the problem is modeled by the quasi-harmonic equation using four-node quadrilateral elements. (Hint: Extend the result from Problem 6.32.)

**15.4** Program development project: Implement the SPR procedure in the solver developed in Problem 7.20 and subsequent additions. Assume a linear elastic problem that is modeled using four-node quadrilateral elements. (Hint: Extend the result from Problem 6.32.)

**15.5** The element size $h$ appearing in the explicit residual error estimator given by Eq. (15.59) is often taken as a constant for a particular element of certain

shape. Consider results from Example 15.2 and explain why the accuracy of the explicit residual error estimator will deteriorate when the aspect ratio of the element increases, i.e., when the mesh becomes more anisotropic.

**15.6** Extend the technique of *recovering normal flux by equilibrated residuals* described in Section 15.7.2 to two-dimensional elastic problems. Consider both plane and axisymmetric geometry.

**15.7** Extend the technique of *recovering normal flux by equilibrated residuals* described in Section 15.7.2 to three-dimensional elastic problems.

**15.8** Program development project: Implement a *recovery-based error estimator* or a *residual-based error estimator* in the solver system developed in Problem 7.20 and subsequent exercises.

**15.9** Program development project: Extend the program developed in Problem 7.20 to compute the SPR solution for displacements. Use the recovered displacements to compute strains and from these stresses.

Follow the procedure given in Section 15.4.2 to project three-node triangular and four-node quadrilateral element values to nodes.

Test your program using (a) the patch test of Problem 7.20 and (b) the curved beam problem shown in Fig. 7.13a.

Report results for both displacements and stresses.

## References


[1] J. Barlow, Optimal stress locations in finite element models, Int. J. Numer. Methods Eng. 10 (1976) 243–251.

[2] L.R. Herrmann, Interpretation of finite element procedures in stress error minimization, Proc. Am. Soc. Civ. Eng. 98 (EM5) (1972) 1331–1336.

[3] M. Krizek, P. Neitaanmaki, On superconvergence techniques, Acta Appl. Math. 9 (1987) 75–198.

[4] Q.D. Zhu, Q. Lin, Superconvergence Theory of the Finite Element Methods, Hunan Science and Technology Press, Hunan, China, 1989.

[5] L.B. Wahlbin, Superconvergence in Galerkin Finite Element Methods, Lectures Notes in Mathematics, vol. 1605, Springer, Berlin, 1995.

[6] C.M. Chen, Y. Huang, High Accuracy Theory of Finite Element Methods, Hunan Science and Technology Press, Hunan, China, 1995.

[7] Q. Lin, N. Yan, Construction and Analyses of Highly Effective Finite Elements, Hebei University Press, Hebei, China, 1996.

[8] M. Krizek, P. Neitaanmaki, R. Stenberg (Eds.), Finite Element Methods: Superconvergence, Post-processing, and a Posteriori Estimates, Lecture Notes in Mathematics, No. 363, Marcel Dekker, New York, 1997.

[9] E. Hinton, J. Campbell, Local and global smoothing of discontinuous finite element function using a least squares method, Int. J. Numer. Methods Eng. 8 (1974) 461–480.

[10] H.J. Brauchli, J.T. Oden, On the calculation of consistent stress distributions in finite element applications, Int. J. Numer. Methods Eng. 3 (1971) 317–325.
[11] O.C. Zienkiewicz, J.Z. Zhu, The superconvergent patch recovery (SPR) and a posteriori error estimates. Part 1: The recovery technique, Int. J. Numer. Methods Eng. 33 (1992) 1331–1364.
[12] O.C. Zienkiewicz, J.Z. Zhu, The superconvergent patch recovery (SPR) and a posteriori error estimates. Part 2: Error estimates and adaptivity, Int. J. Numer. Methods Eng. 33 (1992) 1365–1382.
[13] O.C. Zienkiewicz, J.Z. Zhu, The superconvergent patch recovery (SPR) and adaptive finite element refinement, Comput. Methods Appl. Mech. Eng. 101 (1992) 207–224.
[14] O.C. Zienkiewicz, J.Z. Zhu, J. Wu, Superconvergent recovery techniques—some further tests, Commun. Numer. Methods Eng. 9 (1993) 251–258.
[15] Z. Zhang, Ultraconvergence of the patch recovery technique, Math. Comput. 65 (1996) 1431–1437.
[16] Q.D. Zhu, Q. Zhao, SPR technique and finite element correction, Numer. Math. 96 (2003) 185–196.
[17] Z. Zhang, Ultraconvergence of the patch recovery technique II, Math. Comput. 69 (2000) 141–158.
[18] Z. Zhang, H.D. Victory Jr., Mathematical analysis of Zienkiewicz-Zhu's derivative patch recovery techniques, Numer. Methods Partial Diff. Eq. 12 (1996) 507–524.
[19] Z. Zhang, J.Z. Zhu, Analysis of the superconvergent patch recovery technique and a posteriori error estimator in the finite element method (II), Comput. Methods Appl. Mech. Eng. 163 (1998) 159–170.
[20] B. Li, Z. Zhang, Analysis of a class of superconvergence patch recovery techniques for linear and bilinear finite elements, Numer. Methods Partial Diff. Eq. 15 (1999) 151–167.
[21] Z. Zhang, R. Lin, Ultraconvergence of ZZ patch recovery at mesh symmetry points, Numer. Math. 95 (2003) 781–801.
[22] J. Xu, Z. Zhang, Analysis of recovery type a posteriori error estimators for mildly structured grids, Math. Comput. 73 (2004) 1139–1152.
[23] N.-E. Wiberg, F. Abdulwahab, Patch recovery based on superconvergent derivatives and equilibrium, Int. J. Numer. Methods Eng. 36 (1993) 2703–2724.
[24] N.-E. Wiberg, F. Abdulwahab, S. Ziukas, Enhanced superconvergent patch recovery incorporating equilibrium and boundary conditions, Int. J. Numer. Methods Eng. 37 (1994) 3417–3440.
[25] T.D. Blacker, T. Belytschko, Superconvergent patch recovery with equilibrium and conjoint interpolation enhancements, Int. J. Numer. Methods Eng. 37 (1994) 517–536.
[26] T. Lee, H.C. Park, S.W. Lee, A superconvergent stress recovery technique with equilibrium constraint, Int. J. Numer. Methods Eng. 40 (1997) 1139–1160.

[27] T. Kvamsdal, K.M. Okstad, Error estimation based on superconvergent patch recovery using statically admissible stress field, Int. J. Numer. Methods Eng. 42 (1998) 443–472.
[28] B. Heimsund, X. Tai, J. Wang, Superconvergence for gradient of finite element approximations by $l^2$ projections, SIAM Numer. Anal. 40 (2002) 1263–1280.
[29] P. Díez, J.J. Ródenas, O.C. Zienkiewicz, Equilibrated patch recovery error estimates: simple and accurate upper bounds of the error, Int. J. Numer. Methods Eng. 69 (2007) 2075–2098.
[30] J.J. Ródenas, M. Tur, F.J. Fuenmayor, A. Vercher, Improvement of the superconvergent patch recovery technique by the use of constraint equations: the (SPR-C) technique, Int. J. Numer. Methods Eng. 70 (2007) 705–727.
[31] V. Carey, G.F. Carey, Flexible patch post-processing recovery strategies for solution enhancement and adaptive mesh refinement, Int. J. Numer. Methods Eng. 87 (2011) 424–436.
[32] X.D. Li, N.-E. Wiberg, A posteriori error estimate by element patch postprocessing, adaptive analysis in energy and $l_2$ norm, Comput. Struct. 53 (1994) 907–919.
[33] N.-E. Wiberg, X.D. Li, Superconvergent patch recovery of finite element solutions and a posteriori $l_2$ norm error estimate, Commun. Numer. Methods Eng. 10 (1994) 313–320.
[34] B. Boroomand, O.C. Zienkiewicz, Recovery by equilibrium patches (REP), Int. J. Numer. Methods Eng. 40 (1997) 137–154.
[35] B. Boroomand, O.C. Zienkiewicz, An improved REP recovery and the effectivity robustness test, Int. J. Numer. Methods Eng. 40 (1997) 3247–3277.
[36] F. Ubertini, Patch recovery based on complementary energy, Int. J. Numer. Methods Eng. 59 (2004) 1501–1538.
[37] Z. Zhang, A. Naga, A new finite element gradient recovery method: superconvergence property, SIAM J. Sci. Comput. 26 (2005) 1192–1213.
[38] A. Benedetti, S. de Miranda, F. Ubertini, A poeteriori error estimation based on superconvergent recovery by compatibility in patches, Int. J. Numer. Methods Eng. 67 (2006) 108–131.
[39] Z. Zhang, Polynomial preserving recovery for meshes from Delaunay triangulation or with high aspect ratio, Numer. Methods Partial Diff. Eq. 24 (2008) 960–971.
[40] A. Naga, Z. Zhang, Function value recovery and its application in eigenvalue problems, SIAM J. Numer. Anal. 50 (2012) 272–286.
[41] N.-E. Wiberg, P. Hager, Error estimation and adaptivity for $h$-version eigenfrequency analysis, in: P. Ladevèze, J.T. Oden (Eds.), Advances in Adaptive Computational Methods in Mechanics, Elsevier, 1998, pp. 461–476.
[42] P. Ladevèze, D. Leguillon, Error estimate procedure in the finite element method and applications, SIAM J. Numer. Anal. 20 (3) (1983) 485–509.
[43] P. Ladevèze, J.P. Pelle, P. Rougeot, Error estimation and mesh optimization for classical finite elements, Eng. Comput. 8 (1991) 69–80.

[44] O.C. Zienkiewicz, J.Z. Zhu, A simple error estimator and adaptive procedure for practical engineering analysis, Int. J. Numer. Methods Eng. 24 (1987) 337–357.
[45] I. Babuška, C. Rheinboldt, A-posteriori error estimates for the finite element method, Int. J. Numer. Methods Eng. 12 (1978) 1597–1615.
[46] I. Babuška, W.C. Rheinboldt, Reliable error estimation and mesh adaptation for the finite element method, in: J.T. Oden (Ed.), Computational Methods in Nonlinear Mechanics, North Holland, Amsterdam, 1980, pp. 67–1084.
[47] D.W. Kelly, J.P. De, S.R. Gago, O.C. Zienkiewicz, I. Babuška, A posteriori error analysis and adaptive processes in the finite element method. Part I—Error analysis, Int. J. Numer. Methods Eng. 19 (1983) 1593–1619.
[48] O.C. Zienkiewicz, J.P. De, S.R. Gago, D.W. Kelly, The hierarchical concept in finite element analysis, Comput. Struct. 16 (1983) 53–65.
[49] D.W. Kelly, The self-equilibration of residuals and complementary a posteriori error estimates in the finite element method, Int. J. Numer. Methods Eng. 20 (1984) 1491–1506.
[50] R.E. Bank, A. Weiser, Some a posteriori error estimators for elliptic partial differential equations, Math. Comput. 44 (1985) 283–301.
[51] J.T. Oden, L. Demkowicz, W. Rachowicz, T.A. Westermann, Toward a universal h-p adaptive finite element strategy. Part 2: A posteriori error estimation, Comput. Methods Appl. Mech. Eng. 77 (1989) 113–180.
[52] R. Verfurth, A posteriori error estimators for the Stokes equations, Numer. Math. 55 (1989) 309–325.
[53] C. Johnson, P. Hansbo, Adaptive finite element methods in computational mechanics, Comput. Methods Appl. Mech. Eng. 101 (1992) 143–181.
[54] M. Ainsworth, J.T. Oden, A unified approach to a posteriori error estimation using element residual methods, Numer. Math. 65 (1993) 23–50.
[55] P. Díez, J.J. Egozcue, A. Huerta, A posteriori error estimation for standard finite element analysis, Comput. Methods Appl. Mech. Eng. 163 (1998) 141–157.
[56] R. Verfurth, A Review of A Posteriori Error Estimation and Adaptive Mesh Refinement Techniques, Wiley-Teubner, New York, 1996.
[57] M. Ainsworth, J.T. Oden, A Posteriori Error Estimation in Finite Element Analysis, John Wiley & Sons, New York, 2000.
[58] J.Z. Zhu, O.C. Zienkiewicz, Superconvergence recovery technique and a posteriori error estimators, Int. J. Numer. Methods Eng. 30 (1990) 1321–1339.
[59] J.Z. Zhu, A posteriori error estimation—the relationship between different procedures, Comput. Methods Appl. Mech. Eng. 150 (1997) 411–422.
[60] R. Rodriguez, Some remarks on the Zienkiewicz-Zhu estimator, Numer. Methods Partial Diff. Eq. 10 (1994) 625–635.
[61] M. Picasso, An anisotropic error indicator based on Zienkiewicz-Zhu error estimator: application to elliptic and parabolic problems, SIAM J. Sci. Comput. 4 (2003) 1328–1355.
[62] I. Babuška, T. Strouboulis, C.S. Upadhyay, A model study of the quality of a posteriori error estimators for linear elliptic problems. Error estimation in the

interior of patchwise uniform grids of triangles, Comput. Methods Appl. Mech. Eng. 114 (1994) 307–378.
[63] I. Babuška, T. Strouboulis, C.S. Upadhyay, S.K. Gangaraj, K. Copps, Validation of a posteriori error estimators by numerical approach, Int. J. Numer. Methods Eng. 37 (1994) 1073–1123.
[64] I. Babuška, T. Strouboulis, C.S. Upadhyay, A model study of the quality of a posteriori error estimators for finite element solutions of linear elliptic problems, with particular reference to the behavior near the boundary, Int. J. Numer. Methods Eng. 40 (1997) 2521–2577.
[65] S. Ohnimus, E. Stein, E. Walhorn, Local error estimates of FEM for displacements and stresses in linear elasticity by solving local Neumann problems, Int. J. Numer. Methods Eng. 52 (2001) 727–746.
[66] H. Ohtsubo, M. Kitamura, Numerical investigation of element-wise a posteriori error estimation in two and three-dimensional elastic problem, Int. J. Numer. Methods Eng. 34 (1992) 969–977.
[67] C. Carstensen, J. Alberty, Averaging techniques for reliable a posteriori FE-error control in elastoplasticity with hardening, Comput. Methods Appl. Mech. Eng. 192 (2003) 1435–1450.
[68] O.C. Zienkiewicz, B. Boroomand, J.Z. Zhu, Recovery procedures in error estimation and adaptivity: adaptivity in linear problems, Comput. Methods Appl. Mech. Eng. 176 (1999) 111–125.
[69] I. Babuška, T. Strouboulis, The Finite Element Method and Its Reliability, Clarendon Press, Oxford, 2001.
[70] B. Fraeijs de Veubeke, Displacement and equilibrium models in finite element method, in: O.C. Zienkiewicz, G.S. Holister (Eds.), Stress Analysis, John Wiley & Sons, Chichester, 1965, pp. 145–197 (Chapter 9).
[71] B. Fraeijs de Veubeke, Bending and stretching of plates. Special models for upper and lower bounds, in: Proceedings of the First Conference on Matrix Methods in Structural Mechanics, vol. AFFDL-TR-66-80, Wright Patterson Air Force Base, Ohio, October 1966, pp. 863–886.
[72] B. Fraeijs de Veubeke, O.C. Zienkiewicz, Strain energy bounds in finite element analysis by slab analogy, J. Strain Anal. 2 (1967) 265–271.
[73] M. Ainsworth, D. Kelly, S. Sloan, S. Wang, Post processing with computable error bounds for finite element approximation of a non-linear heat conduction problem, IMA J. Numer. Anal. 17 (1997) 547–561.
[74] P. Ladevèze, G. Coffignal, J.P. Pelle, Accuracy of elastoplastic and dynamic analysis, in: I. Babuška, O.C. Zienkiewicz, J. Gago, E.R. de A. Oliviera (Eds.), Accuracy Estimates and Adaptivity for Finite Elements, John Wiley & Sons, New York, 1986 (Chapter 11).
[75] R. Cottereau, P. Díez, A. Huerta, Strict error bounds for linear solid mechanics problems using a subdomain-based flux-free method, Comput. Mech. 44 (2009) 533–547.

[76] I. Babuška, A. Miller, The post-processing approach in the finite element method. Part 1: Calculation of displacements, stresses and other higher derivatives of displacements, Int. J. Numer. Methods Eng. 20 (1984) 1085–1109.
[77] I. Babuška, A. Miller, The post-processing approach in the finite element method. Part 2: The calculation of stress intensity factors, Int. J. Numer. Methods Eng. 20 (1984) 1111–1129.
[78] I. Babuška, A. Miller, The post-processing approach in the finite element method. Part 3: A posteriori error estimates and adaptive mesh selection, Int. J. Numer. Methods Eng. 20 (1984) 2311–2324.
[79] R. Becker, R. Rannacher, A feed-back approach to error control in finite element methods: basic analysis and examples, East-West J. Numer. Math. 4 (1996) 237–264.
[80] R. Rannacher, F.-T. Stuttmeier, A feed-back approach to error control in finite element methods: application to linear elasticity, Comput. Mech. 19 (1997) 434–446.
[81] M. Paraschivoiu, J. Peraire, A. Patera, A posteriori finite element bounds for linear-functional outputs of elliptic partial differential equations, Comput. Methods Appl. Mech. Eng. 150 (1–4) (1997) 289–312 (Symposium on Advances in Computational Mechanics, vol. 2, Austin, TX, 1997).
[82] R. Rannacher, F.-T. Stuttmeier, A posteriori error control in finite element methods via duality techniques: application to perfect plasticity, Comput. Mech. 21 (1998) 123–133.
[83] S. Prudhomme, J.T. Oden, On goal-oriented error estimation for elliptic problems: application to the control of pointwise errors, Comput. Methods Appl. Mech. Eng. 176 (1999) 313–331.
[84] J. Peraire, A.T. Patera, Asymptotic a posteriori finite element bounds for the outputs of noncoercive problems: the Helmholtz and Burgers equations, Comput. Methods Appl. Mech. Eng. 171 (1–2) (1999) 77–86.
[85] Y. Maday, A.T. Patera, J. Peraire, A general formulation for a posteriori bounds for output functionals of partial differential equations; application to the eigenvalue problem, C.R. Acad. Sci. Paris Sér. I Math. 328 (9) (1999) 823–828.
[86] L. Machiels, J. Peraire, A.T. Patera, Output bound approximations for partial differential equations; application to the incompressible Navier-Stokes equations, in: Industrial and Environmental Applications of Direct and Large-eddy Simulation (Istanbul, 1998), vol. 529, Lecture Notes in Physics, Springer, Berlin, 1999, pp. 93–108.
[87] J.T. Oden, S. Prudhomme, Goal-oriented error estimation and adaptivity for the finite element method, Comput. Math. Appl. 41 (5–6) (2001) 735–756.
[88] L. Machiels, J. Peraire, A.T. Patera, A posteriori finite-element output bounds for the incompressible Navier-Stokes equations: application to a natural convection problem, J. Comput. Phys. 172 (2) (2001) 401–425.
[89] A.T. Patera, J. Peraire, A general Lagrangian formulation for the computation of a posteriori finite element bounds, in: Error estimation and adaptive

discretization methods in computational fluid dynamics, vol. 25, Lecture Notes in Computer Science and Engineering, Springer, Berlin, 2003, pp. 159–206.
[90] S. Prudhomme, J.T. Oden, T. Westerman, J. Bass, M.E. Botkin, Practical methods for a posteriori error estimation in engineering applications, Int. J. Numer. Methods Eng. 56 (2003) 1193–1224.
[91] N. Parés, J. Bonet, A. Huerta, J. Peraire, The computation of bounds for linear-functional outputs of weak solutions to the two-dimensional elasticity equations, Comput. Methods Appl. Mech. Eng. 195 (2006) 406–429.
[92] L. Chamoin, P. Ladevèze, A non-intrusive method for the calculation of strict and efficient bounds of calculated output of interest in linear viscoelasticity problems, Comput. Methods Appl. Mech. Eng. 197 (2008) 994–1014.
[93] N. Parés, P. Díez, A. Huerta, Exact bounds for linear output of advect-diffucion-reaction equation using flux-free error estimates, SIAM J. Sci. Comput. 31 (2009) 3064–3089.
[94] R.H. Nochetto, A. Veeser, M. Verani, A safeguarded dual weighted residual method, IMA J. Numer. Anal. 29 (2009) 126–140.
[95] P. Ladevéze, L. Chamoin, Calculation of strict error bounds for finite element approximations of non-linear pointwise quantities of interest, Int. J. Numer. Methods Eng. 84 (2010) 1638–1664.
[96] T. Richter, A posteriori error estimation and anisotropy detection with dual-weighted residual method, Int. J. Numer. Methods Fluids 62 (2010) 90–118.
[97] M. Ainsworth, R. Rankin, Guaranteed computational bounds on quantities of interest in finite element computations, Int. J. Numer. Methods Eng. 89 (2012) 1605–1634.
[98] R. Becker, R. Rannacher, An optimal control approach to a posteriori error estimation in finite element methods, Acta Numer. 10 (2001) 1–102.
[99] M.B. Giles, E. Süli, Adjoint methods for PDEs: a posteriori error analysis and postprocessing by duality, Acta Numer. 11 (2002) 145–236.

This page is intentionally left blank

CHAPTER

# 16 Adaptive Finite Element Refinement

## 16.1 Introduction

In the previous chapter we discussed at some length various methods of recovery by which the finite element solution results could be made more accurate and this led us to devise various procedures for error estimation. In this chapter we are concerned with methods which can be used to reduce the errors once a finite element solution has been obtained. As the process depends on previous results at all stages it is called adaptive. Such adaptive methods were first introduced to finite element calculations by Babuška and Rheinboldt in the late 1970s [1,2].

Before proceeding further it is necessary to clarify the objectives of refinement and specify "permissible error magnitudes" and here the engineer or user must have very clear aims. For instance the naive requirement that all displacements or all stresses should be given within a specified tolerance is not acceptable. The reasons for this are obvious as at singularities, for example, stresses will always be infinite and therefore no finite tolerance could be specified. The same difficulty is true for displacements if point or knife edge loads are considered.

The most common criterion in general engineering use is that of prescribing a total limit of the error computed in the energy norm. Often this error is required not to exceed a specified percentage of the total energy norm of the solution and in the many examples presented later, we shall use this simple criterion. However, using a recovery type of error estimator it is possible to adaptively refine the mesh so that the accuracy of a certain quantity of interest, such as the RMS error in displacement and/or RMS error in stress [see Chapter 15, Eqs. (15.9) and (15.10)], satisfy some user-specified criterion. We should recognize that mesh refinement based on reducing the RMS error in displacement is in effect reducing the average displacement error in a user-specified region (e.g., in each element); similarly mesh refinement based on reducing the RMS error in stress is the same as reducing the average stress error in a user-specified region. Here we could, for instance, specify directly the permissible error in stresses or displacements at any location. Some investigators (e.g., Zienkiewicz and Zhu [3]) have used RMS error in stress in the adaptive mesh refinement to obtain more accurate stress solutions. Others (e.g., Oñate and Bugeda [4]) have used the requirement of constant energy norm density in the adaptive analysis, which is in fact equivalent to specifying a uniform distribution of RMS error in stress in each element. We note that the recovery type of error estimators is particularly useful and

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00016-2

convenient in designing adaptive analysis procedures for the quantities of interest. The methodologies of designing adaptive analysis procedures based on error estimation of the quantities of interest are in principle the same as described in this chapter; for additional details we refer the reader to Refs. [5–7].

As we have already remarked in the previous chapter we will at all times consider the error in the actual finite element solution rather than the error in the recovered solution. It may indeed be possible in special problems for the error in the recovered solution to be zero, even if the error in the finite element solution itself is quite substantial. (Consider here for instance a problem with a linear stress distribution being solved by linear elements which results in constant element stresses. Obviously the element error will be quite large. But if recovered stresses are used, exact results can be obtained and no errors will exist.) The problem of which errors to consider still needs to be answered. At the present time we shall consider the question of recovery as that of providing a very substantial margin of safety in the definition of errors.

Various procedures exist for the refinement of finite element solutions. Broadly these fall into two categories:

1. The $h$-refinement in which the same class of elements continue to be used but are changed in size, in some locations made larger and in others made smaller, to provide maximum economy in reaching the desired solution.
2. The $p$-refinement in which we continue to use the same element size and simply increase, generally hierarchically, the order of the polynomial used in their definition.

It is occasionally useful to divide the above categories into subclasses, as the $h$-refinement can be applied and thought of in different ways. In Fig. 16.1 we illustrate three typical methods of $h$-refinement.

1. The first of these $h$-refinement methods is *element subdivision* (enrichment) (Fig. 16.1b). Here refinement can be conveniently implemented and existing elements, if they show too much error, are simply divided into smaller ones keeping the original element boundaries intact. Such a process is cumbersome as many *hanging points* are created where an element with mid-side nodes is joined to a linear element with no such nodes. On such occasions it is necessary to provide local constraints at the hanging points and the calculations become more involved. In addition, the implementation of de-refinement requires rather complex data management which may reduce the efficiency of the method. Nevertheless, the method of element subdivision is quite widely used.
2. The second method is that of a complete *mesh regeneration* or *remeshing* (Fig. 16.1c). Here, on the basis of a given solution, a new element size is predicted in all the domain and a totally new mesh is generated. Thus a refinement and de-refinement are simultaneously allowed. This of course can be expensive, especially in three dimensions where mesh generation is difficult for certain types of elements, and it also presents a problem of transferring data from one mesh

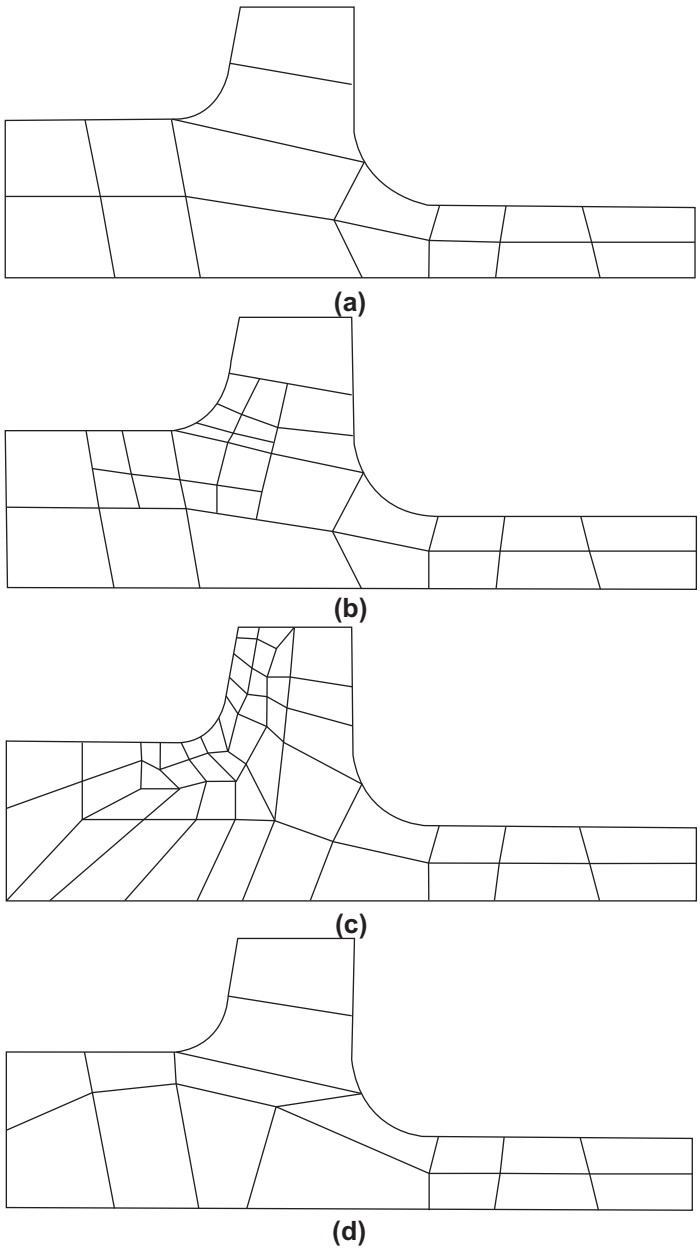



**FIGURE 16.1**
Various procedures by *h*-refinement: (a) original mesh, (b) mesh enhancement by subdivision (enrichment), (c) mesh enhancement by remeshing, and (d) *h*-refinement of original mesh by reposition of nodes.

to another. However, the results are generally much superior and this method will be used in most of the examples shown in this chapter. For many practical engineering problems, particularly of those for which the element shape will be severely distorted during the analysis, adaptive mesh regeneration is a natural choice.

3. The final method, sometimes known as *r-refinement* (Fig. 16.1d), keeps the total number of nodes constant and adjusts their position to obtain an optimal approximation [8–10]. While this procedure is theoretically of interest it is difficult to use in practice and there is little to recommend it. Further it is not a true refinement procedure as a prespecified accuracy cannot generally be reached.

We shall see that with energy norms specified as the criterion, it is a fairly simple matter to predict the element size required for a given degree of approximation. Thus very few resolutions are generally necessary to reach the objective.

With $p$-refinement the situation is different. Here two subclasses exist:

1. One in which the polynomial order is increased uniformly throughout the whole domain.
2. One in which the polynomial order is increased locally using hierarchical refinement.

In neither of these has a direct procedure been developed which allows the prediction of the best refinement to be used to obtain a given error. Here the procedures generally require more resolutions and tend to be more costly. However, the convergence for a given number of variables is more rapid with $p$-refinement and it has much to recommend it.

On occasion it is possible to combine efficiently the $h$- and $p$-refinements and call it the $hp$-refinement. In this procedure both the size of elements $h$ and their degree of polynomial $p$ are altered. Much work has been reported in the literature by Babuška, Oden, and others and the interested reader is referred to Refs. [11–21].

In Sections 16.2 and 16.3 we shall discuss both the $h$- and the $p$-refinement methods. In Section 16.3 we also include some details of the very simple and yet efficient $hp$-refinement process introduced by Zienkiewicz et al. [22].

## 16.2 Adaptive *h*-refinement

### 16.2.1 Predicting the required element size in *h*-adaptivity

In the introduction to this chapter we have mentioned several alternative processes of $h$-adaptivity and we suggested that the process in which the complete mesh is regenerated is in general the most efficient. Such a procedure allows elements to be de-refined (or enlarged) as well as refined (made smaller) and invariably starts at each stage of the analysis from a specification of the mesh size $h$ defined at each nodal point of the previous mesh. Standard interpolation is used to find the size of elements

required at any point in the domain. As the refinement process proceeds for each subsequent stage of analysis the computed mesh sizes $h$ are based on a prescribed accuracy at the nodes of the previous mesh.

The error estimators discussed in the previous chapter allow the global energy (or similar) norm of the error to be determined and the errors occurring locally (at the element level) are usually also well represented. If these errors are within the limits prescribed by the analyst then clearly the work is completed. More frequently these limits are exceeded and refinement is necessary. The question which this section addresses is how best to effect this refinement. Here obviously many strategies are possible and much depends on the *objectives* or *goals* to be achieved.

In the simplest case we shall seek, for instance, to make the relative energy norm error $\eta$ [viz. Eq. (15.12)] less than some specified value $\bar{\eta}$ (say 5% for many engineering applications). Thus

$$\eta \leq \bar{\eta} \tag{16.1}$$

is to be achieved.

In an "optimal mesh" it is desirable that the distribution of element energy norm error (i.e., $\|\mathbf{e}\|_K$) should be equal for all elements. Thus if the total permissible error is determined (assuming that it is given by the result of the approximate analysis) as

$$\text{Permissible error} \equiv \bar{\eta}\|\mathbf{u}\| = \bar{\eta}\left(\|\hat{\mathbf{u}}\|^2 + \|\mathbf{e}\|^2\right)^{1/2} \tag{16.2}$$

here we have used [23]

$$\|\mathbf{e}\|^2 = \|\mathbf{u}\|^2 - \|\hat{\mathbf{u}}\|^2 \tag{16.3}$$

We could pose a requirement that the error in any element $k$ should be

$$\|\mathbf{e}\|_K < \bar{\eta}\left(\frac{\|\hat{\mathbf{u}}\|^2 + \|\mathbf{e}\|^2}{m}\right)^{1/2} \equiv \bar{e}_m \tag{16.4}$$

where $m$ is the number of elements involved.

Elements in which the above is not satisfied are obvious candidates for refinement. Thus if we define a refinement ratio by

$$\xi_K = \frac{\|\mathbf{e}\|_K}{\bar{e}_m} \tag{16.5}$$

we shall refine whenever[1]

$$\xi_K > 1 \tag{16.6}$$

The refinement ratio $\xi_K$ can be approximated, of course, by replacing the true error in Eqs. (16.4) and (16.5) with the error estimators.

[1] We can indeed "de-refine" or use a larger element spacing where $\xi_K < 1$ if computational economy is desired.

The refinement could be carried out progressively by refining only a certain number of elements in which $\xi$ is higher than a specified limit. This type of element subdivision process is also known as *mesh enrichment* as depicted in Fig. 16.1b. This process of refinement, though ultimately leading to a satisfactory solution being obtained with a relatively small number of total degrees of freedom, is in general not economical as the total number of trial solutions is usually excessive.

It is more efficient to try to design a completely new mesh which satisfies the requirement that

$$\xi_K \leq 1 \tag{16.7}$$

in all elements.

One possibility here is to invoke the asymptotic convergence rate criteria to predict the element size distribution. For instance, if we assume

$$\|\mathbf{e}\|_K \propto h_K^p \tag{16.8}$$

where $h_K$ is the current element size and $p$ is the polynomial order of approximation, then to satisfy the requirement of Eq. (16.4) the new generated element size should be no larger than

$$h_{\text{new}} = \xi_K^{-1/p} h_K \tag{16.9}$$

Mesh generation programs in which the local element size can be specified are available now as we will discuss in Chapter 17 and these can be used to design a new mesh for which the reanalysis is carried out [24,25]. In the figures we show how starting from a relatively coarse solution a single mesh prediction often allows a solution (almost) satisfying the specified accuracy requirement to be achieved.

The reason for the success of the mesh regeneration based on the simple assumption of asymptotic convergence rate implied in Eq. (16.8) is the fact that with refinement the mesh tends to be "optimal" and the localized singularity influence no longer affects the overall convergence.

Of course the effects of any singularity will still remain present in the elements adjacent to it. An improved mesh results if in such elements we use the appropriate convergence and replace $p$ by $\lambda$ in Eqs. (16.8) and (16.9), to obtain

$$h_{\text{new}} = \xi_K^{-1/\lambda} h_K \tag{16.10}$$

in which $\lambda$ is the singularity strength, see Chapter 15, Eq. (15.14). A conservative number to use here is $\lambda = 0.5$ as most singularity parameters lie in the range 0.5–1.0. With this procedure added to the refinement strategy, we frequently achieve accuracies better than the prescribed limit in one remeshing.

### 16.2.2 Numerical examples

In the examples which follow we will show in general a process of refinement in which the total number of degrees of freedom increases with each stage, even though the mesh is redesigned. This need not necessarily be the case as a fine but badly

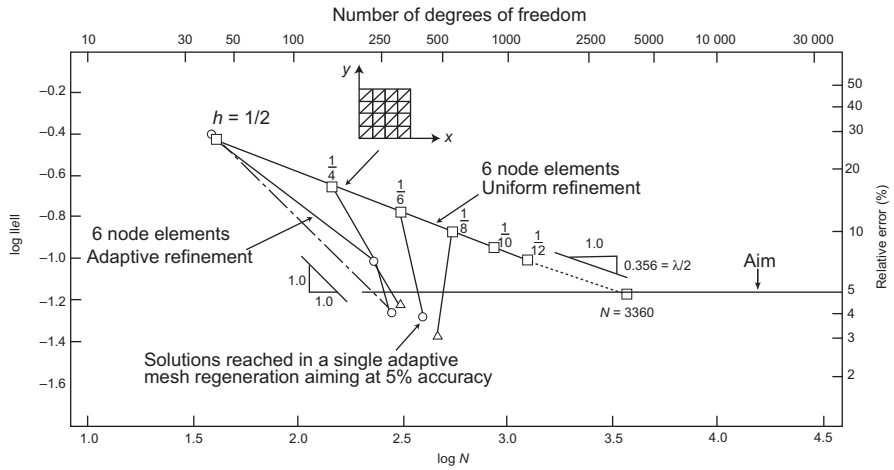



**FIGURE 16.2**

The influence of initial mesh on convergence rates in *h* version. Adaptive refinement using quadratic triangular elements. Problem of Fig. 16.3. Note that if initial mesh is finer than $h = 1/8$ adaptive refinement reduces the number of equations.

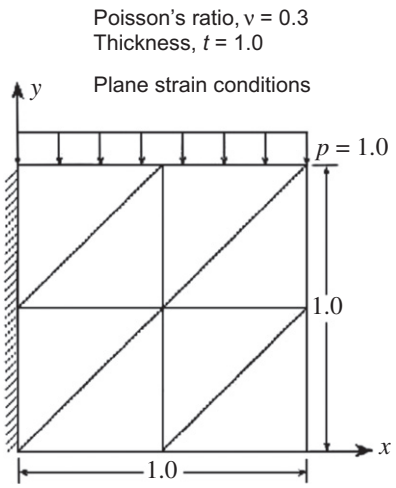
structured mesh can show much greater error than a near-optimal one. To illustrate this point we show in Fig. 16.2 a refinement designed to reach 5% accuracy in one step starting from uniform mesh subdivisions. We note that now, in at least one refinement, a decrease of total error occurs with a reduction of total degrees of freedom (starting from a uniform $8 \times 8$ subdivision with 544 equations and $\eta = 9.8\%$ to $\eta = 3.1\%$ with 460 equations).

We shall now present further typical examples of *h*-refinement with mesh adaptivity. In all of these, full mesh regeneration is used at every step.

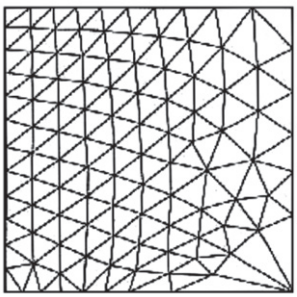
**Example 16.1.** Short cantilever beam

This problem refers to a short cantilever beam in which two very high singularities exist at the corners attached to a rigid wall. The beam is loaded by a uniform load along the top boundary as illustrated in Figs. 16.3 and 16.4. In the refinement process we use both the mesh criteria of Eqs. (16.9) and (16.10) [26]. In Figs. 16.3 and 16.4 we show three stages of an adaptive solution and in Fig. 16.5 we indicate how rapidly these converge, although all uniform refinements converge at a very slow rate (due to the singularities).

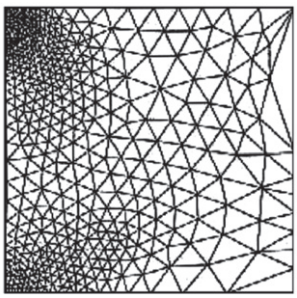
The same problem is also solved by both mesh enrichment and mesh regeneration using linear quadrilateral elements to achieve 5% accuracy. The prescribed accuracy is obtained with optimal rate of convergence being reached by both adaptive refinement processes (Fig. 16.6). However, the mesh enrichment method requires seven refinements, as shown in Fig. 16.7, while mesh regeneration requires only three (see Fig. 16.8). Here the refinement criterion, Eq. (16.6), is used for the mesh enrichment process.




Mesh 1



Mesh 2



Mesh 3

**FIGURE 16.3**
Short cantilever beam and adaptive meshes of linear triangular elements.

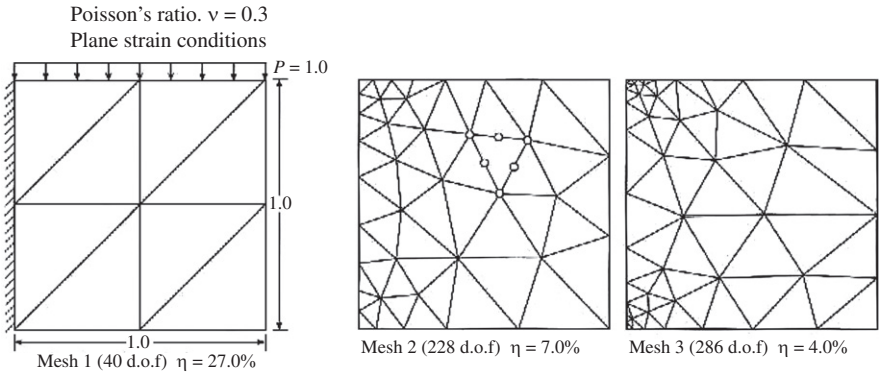
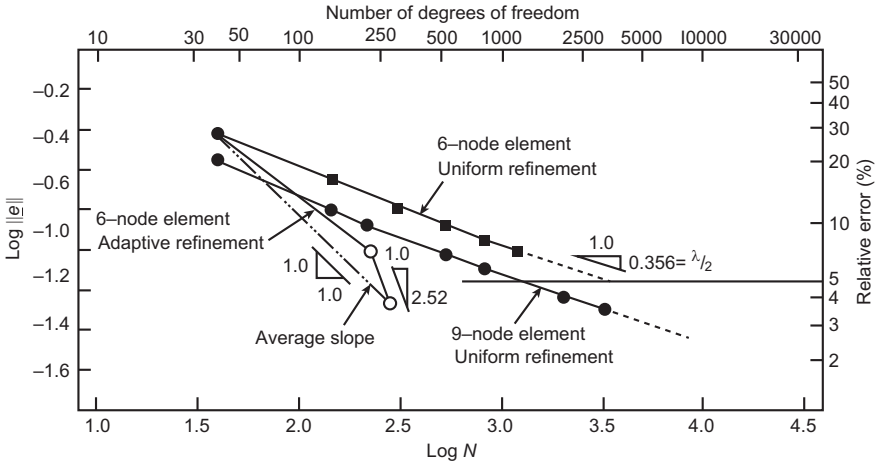



**FIGURE 16.4**

Adaptive mesh of quadratic triangular elements for short cantilever beam.




$\lambda/2 = 0.356$, theoretical rate of convergence for uniform refinement

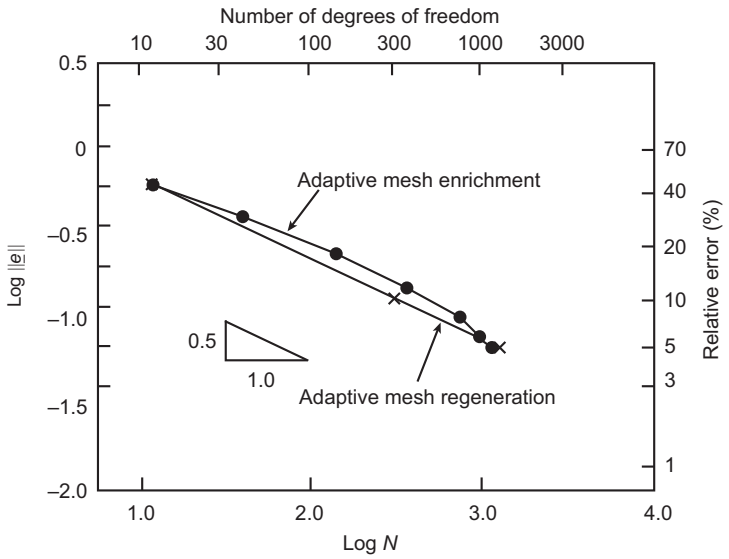
$P/2 = 1.0$, maximum rate of convergence

**FIGURE 16.5**

Experimental rates of convergence for short cantilever beam.

**Example 16.2.** Stressed cylinder

As we mentioned earlier, the value of the energy norm error is not necessarily the best criterion for practical refinement. Limits on the local stress error can be used effectively. Such errors are quite simply obtained by the recovery processes described in the previous chapter (SPR in Section 15.4 and REP in Section 15.5). In Fig. 16.9 we show a simple exercise conducted by Oñate and Bugeda [4] in which a refinement of a stressed cylinder is made using various criteria as described in the caption of Fig. 16.9. It will be observed that the stress tolerance method generally needs a much finer mesh.




$P/2 = 0.5$, optimal rate of convergence

**FIGURE 16.6**

Short cantilever beam. Mesh enrichment versus mesh regeneration using linear quadrilateral elements.

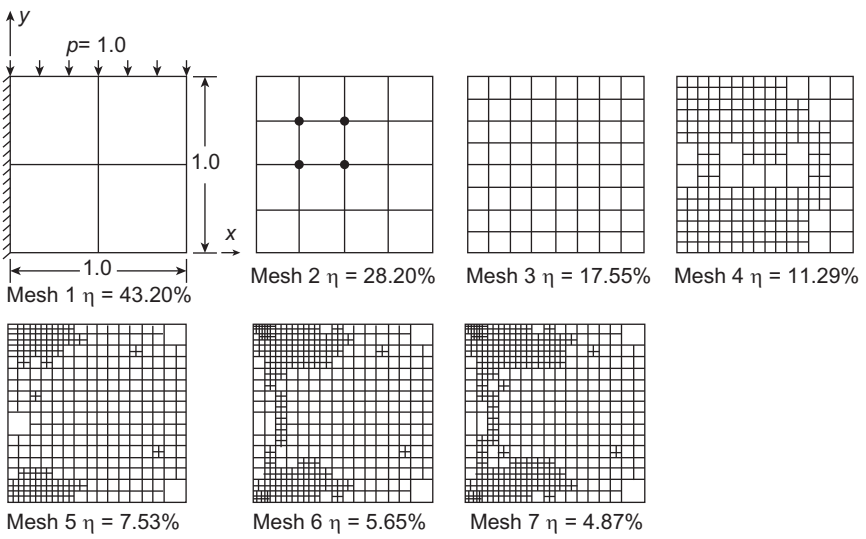



**FIGURE 16.7**

Short cantilever solved by mesh enrichment. Linear quadrilateral elements.

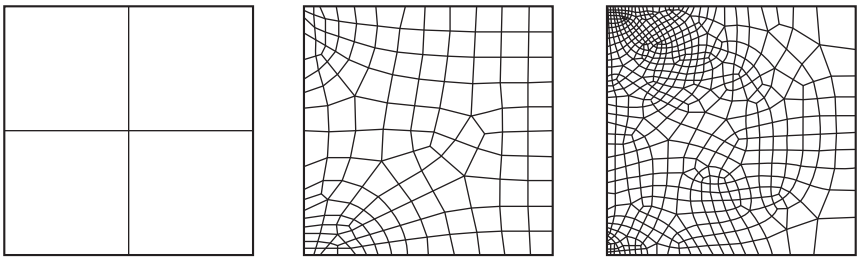



**FIGURE 16.8**
Short cantilever solved by mesh regeneration. Linear quadrilateral elements.

**Example 16.3.** A Poisson equation in a square domain

This example is fairly straightforward and starts from a simple square domain in which suitable loading terms exist in a Poisson equation to give the solution shown in Fig. 16.10 [15]. In Fig. 16.11 we show the first subdivision of this domain into regular linear and quadratic elements and the subsequent refinements. The elements are of both triangular and quadrilateral shape and for the linear ones a target error of 10% in total energy has been set, while for quadratic elements the target error is 1% of total energy. In practically all cases three refinements suffice to reach a very accurate solution satisfying the requirements despite the fact that the original mesh cannot capture in any way the high intensity region illustrated in the previous figure. It is of interest to note that the effectivity indices in all cases are very close to one—this is true even for the original refinement. Figure 16.12 shows the convergence for various elements with the error plotted against the total number of degrees of freedom. The reader should note that the asymptotic rate of convergence is exceeded when the refinement gets closer to its final objective.

**Example 16.4.** An L-shaped domain

It is of interest to note the results in Fig. 16.13 which come from an analysis of a re-entrant corner using isoparametric quadratic quadrilaterals. Here two meshes are shown together with the convergence data of the solution.

**Example 16.5.** A machine part

For this machine part problem plane strain conditions are assumed. A prescribed accuracy of 5% relative error is achieved in one adaptive refinement (see Fig. 16.14) with linear quadrilateral elements. The convergence of the shear stress $\tau_{xy}$ is shown in Fig. 16.15.

**Example 16.6.** A perforated gravity dam

The final example of this section shows a more practical engineering problem of a perforated dam. This dam was analyzed in the late 1960s during its construction. The problem was revisited to choose a suitable mesh of quadratic triangles. Figure

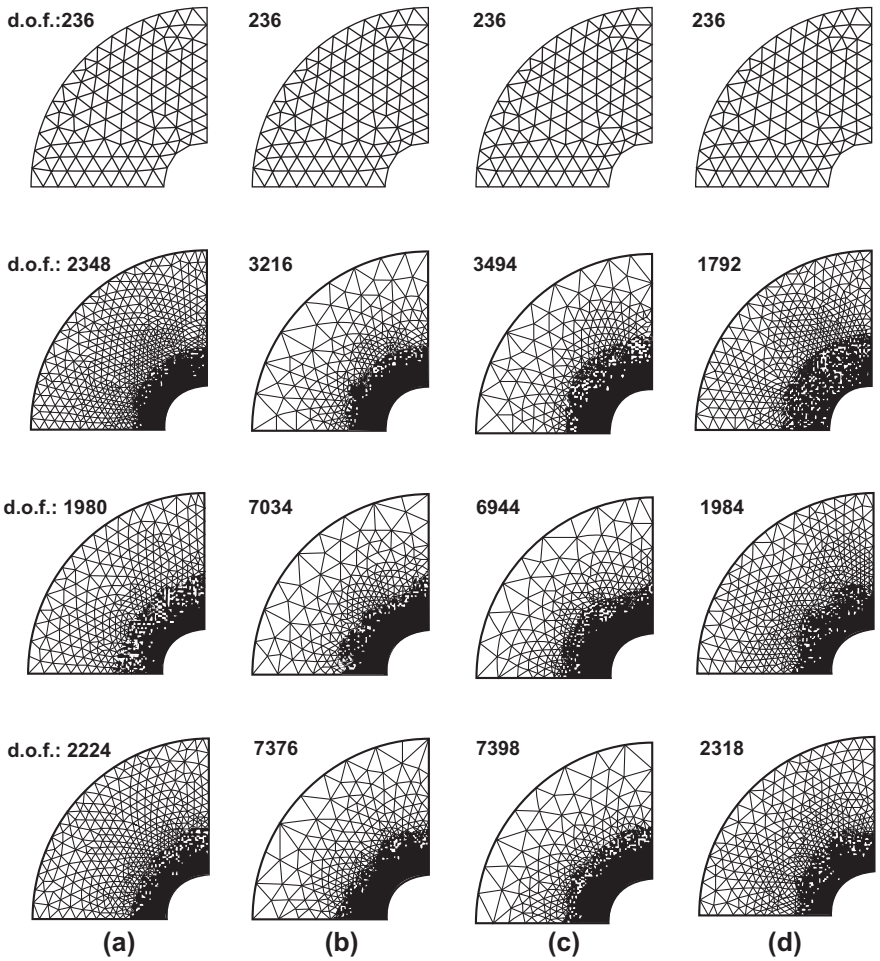



**FIGURE 16.9**

Sequence of adaptive mesh refinement strategies based on (a) equal distribution of the global energy error between all the elements, (b) equal distribution of the density of energy error, (c) equal distribution of the maximum error in stresses at each point, and (d) equal distribution of the maximum percentage of the error in stresses at each point. All final meshes have less than 5% energy norm error.

16.16a shows the mesh chosen. Despite the high order of elements the error is quite high, around 17%. One stage of adaptive refinement reaches the specified value of 5% error in energy norm. As we have seen in previous examples such convergence is not always possible but it is achieved here. We believe this typical example shows the advantages of adaptivity and the ease with which a final good mesh can be arrived at automatically.

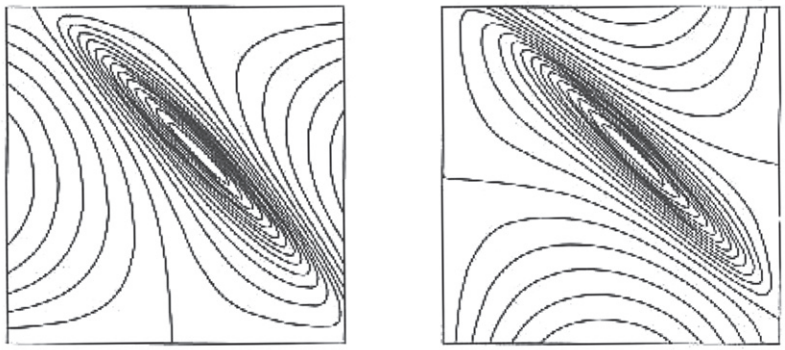



**FIGURE 16.10**
Poisson equation "exact" solutions: (a) $\partial u/\partial x$ contours (b) $\partial u/\partial y$ contours.

## 16.3 *p*-refinement and *hp*-refinement

The use of nonuniform $p$-refinement is of course possible if done hierarchically and many attempts have been made to do this efficiently. Some of this was done as early as 1983 [27,28]. However, the general process is difficult and necessitates many assumptions about the decrease of error. Certainly, the desired accuracy can seldom be obtained in a single step and most of the work on this requires a sequence of steps. We illustrate such a refinement process in Fig. 16.17 for the perforated dam problem presented in the previous section.

The same applies to $hp$-processes in which much work has been done during the last two decades [11–21]. We shall quote here only one particular attempt at $hp$-refinement which seems to be particularly efficient and where the number of resolutions is quite small. The methodology was introduced by Zienkiewicz et al. [22] and we shall quote here some of the procedures suggested.

The first procedure is that of pursuing an $h$-refinement with lower order elements (e.g., linear or quadratic elements) to obtain, say, a 5% accuracy, at which stage the energy norm error is nearly uniformly distributed throughout all elements. From there a $p$-refinement is applied in a uniform manner (i.e., the same $p$ is used in all elements). This has very substantial computational advantages as programming is easy and can be readily accomplished, especially if hierarchical functions are used. The uniform $p$-refinement also allows the global energy norm error to be approximately extrapolated by three consecutive solutions [29].

The convergence of the $p$-refinement finite element solution can be written as [30]

$$\|\mathbf{e}\| \leq CN^{-\beta} \tag{16.11}$$

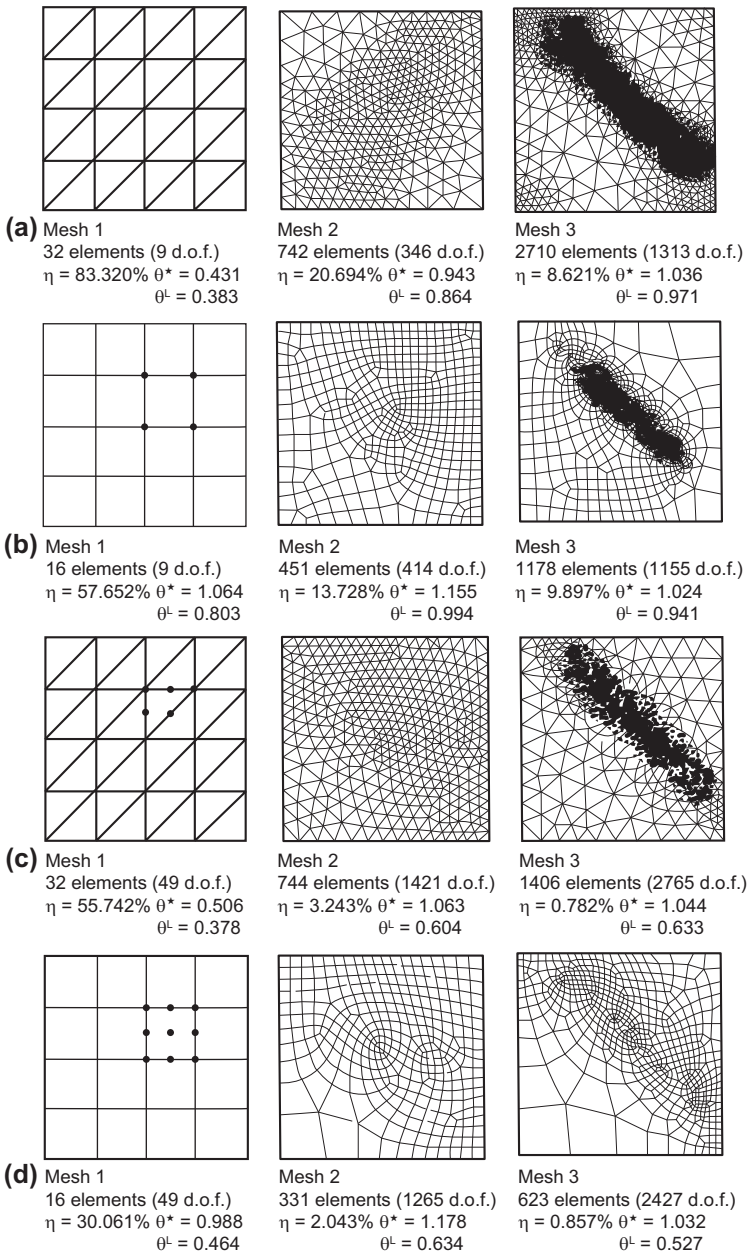



**FIGURE 16.11**

Poisson problem of Fig. 16.10. Adaptive solutions for (a) linear triangles, (b) linear quadrilaterals, (c) quadratic triangles, and (d) quadratic quadrilaterals. $\theta^\star$ based on SPR, $\theta^L$ based on $L_2$ projection. Target error 10% for linear elements and 1% for quadratic elements.

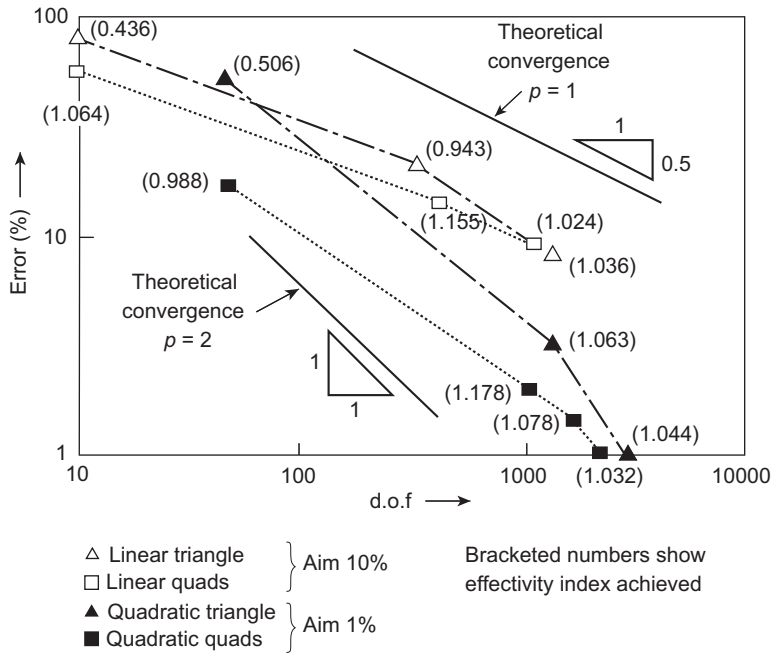



**FIGURE 16.12**
Adaptive refinement for Poisson problem of Fig. 16.10.

where $C$ and $\beta$ are positive constants depending on the solution of the problem and $N$ is the number of degrees of freedom. We assume that for each refinement the error is, observing Eq. (16.3),

$$\|\mathbf{u}\|^2 - \|\hat{\mathbf{u}}_q\|^2 = CN_q^{-2\beta} \tag{16.12}$$

with $q = p-2, p-1, p$ for the three solutions. Eliminating the two constants $C$ and $\beta$ from the above three equations, $\|\mathbf{u}\|^2$ can be solved by

$$\frac{\|\mathbf{u}\|^2 - \|\hat{\mathbf{u}}_p\|^2}{\|\mathbf{u}\|^2 - \|\hat{\mathbf{u}}_{p-1}\|^2} = \left(\frac{\|\mathbf{u}\|^2 - \|\hat{\mathbf{u}}_{p-1}\|^2}{\|\mathbf{u}\|^2 - \|\hat{\mathbf{u}}_{p-2}\|^2}\right)^{\frac{\log(N_{p-1}/N_p)}{\log(N_{p-2}/N_{p-1})}} \tag{16.13}$$

The global energy norm error for the final solution and indeed the error at any stage of the $p$-refinement can be determined using

$$\|\mathbf{e}\|^2 = \|\mathbf{u}\|^2 - \|\hat{\mathbf{u}}_q\|^2 \tag{16.14}$$

$q = 1, 2, \ldots, p$.

**Example 16.7.** *h*-*p*-refinement of L-shaped domain and short cantilever beam

Generally the high accuracy is gained rapidly by refinement, at least from examples performed to date. In Figs. 16.18 and 16.19 we show two examples for which we have

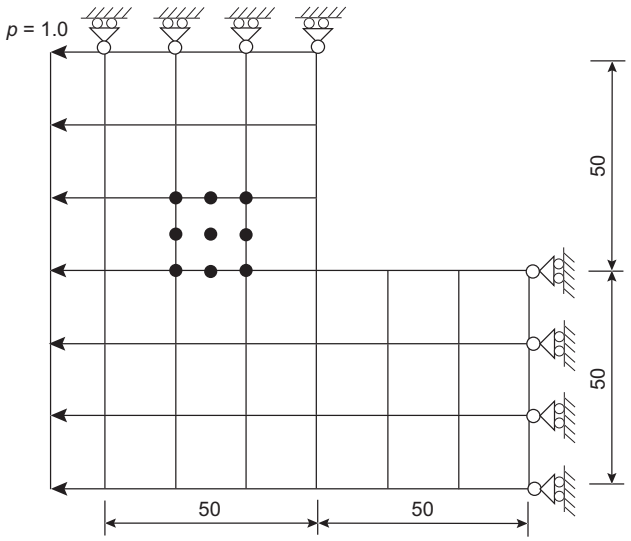



**FIGURE 16.13**
Adaptive refinement of an L-shaped domain in plane stress with prescribed error of 1%.

previously used an $h$-refinement. The first illustrates the L-shaped domain with one singularity and the second the short cantilever beam with two strong singularities. Both problems are solved first using $h$-refinement until the target 5% accuracy is reached using quadratic triangles. At this stage $p$ is increased to third and fourth

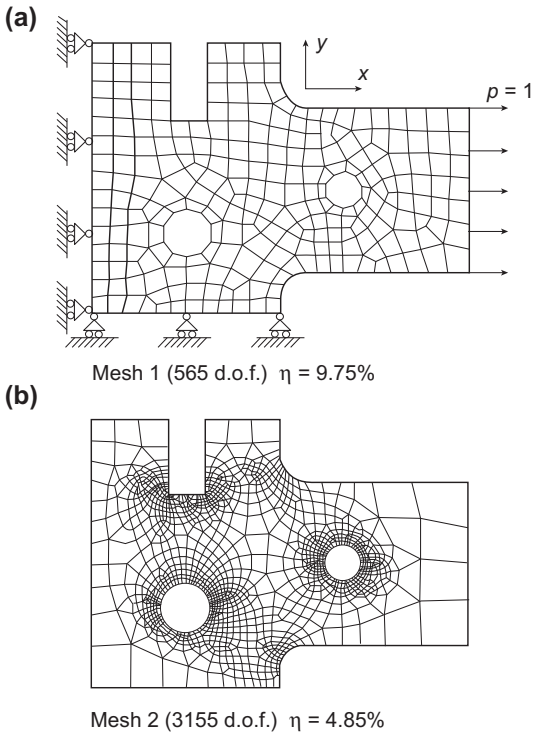



**FIGURE 16.14**

Adaptive refinement of a machine part using linear quadrilateral elements. Target error 5%.

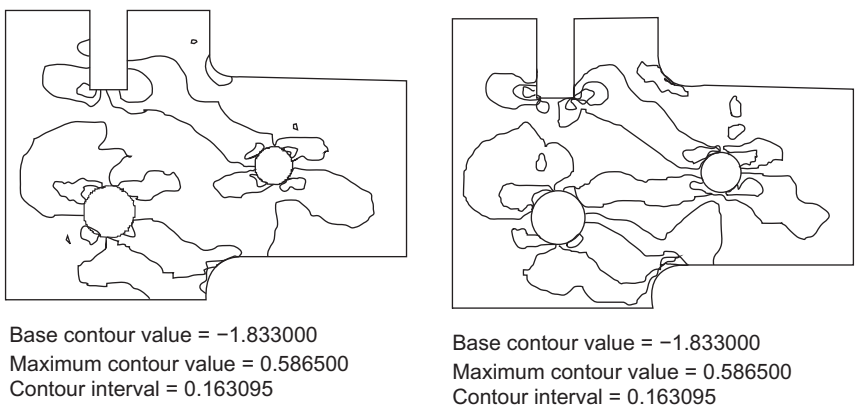
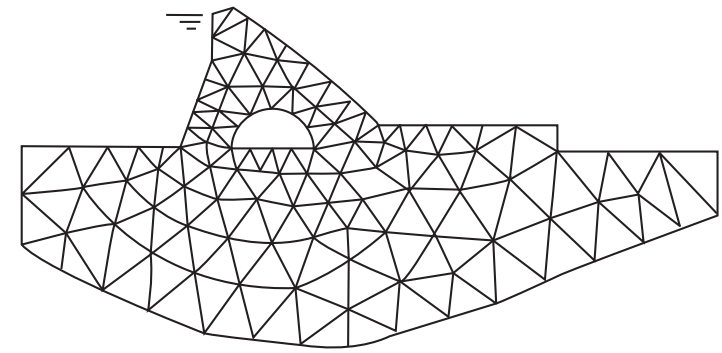


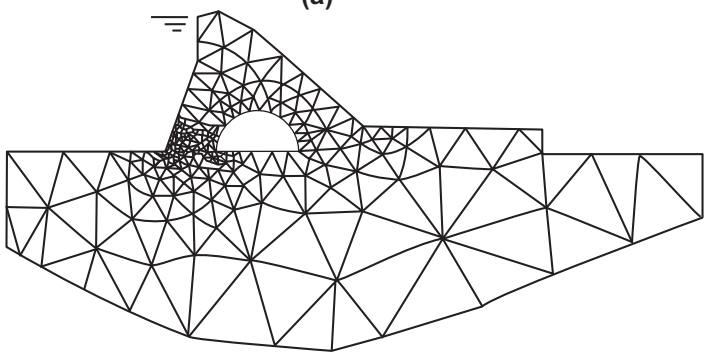

**FIGURE 16.15**

Adaptive refinement of a machine part. Contours of shear stress for original and final mesh.



Mesh 1 ($\eta$ = 16.5%, $\theta$ = 1.05, 728 DOF)

**(a)**



Mesh 2 ($\eta$ = 4.9%, $\theta$ = 1.06, 1764 DOF)

**(b)**

**FIGURE 16.16**

Quadratic triangle. Automatic mesh generation to achieve 5% accuracy. Plane strain analysis of a dam with perforation, water loading only: (a) original mesh and (b) refined mesh.

order so that three solutions are available. At the end of the third solution the error is less than 1%.

In the same paper [22] an alternative procedure is suggested. This uses a very coarse mesh at the outset followed by $p$-refinement. In this case the error at the element level is estimated at the last stage of the $p$-refinement as the difference between the last two refinements (e.g., the third and fourth order when the maximum $p$ is 4). The global error estimator is calculated by the extrapolation procedure used in the previous example. The element error estimator is for order $p-1$ rather than the highest order $p$. It is, however, very accurate. The element error estimator is subsequently used to compute the optimal mesh size as described in Section 16.2.1. A nearly optimal rate of convergence is expected to be achieved because the optimal mesh is designed for $p-1$ order elements. Details of this process will be found again in the reference and will not be discussed further.

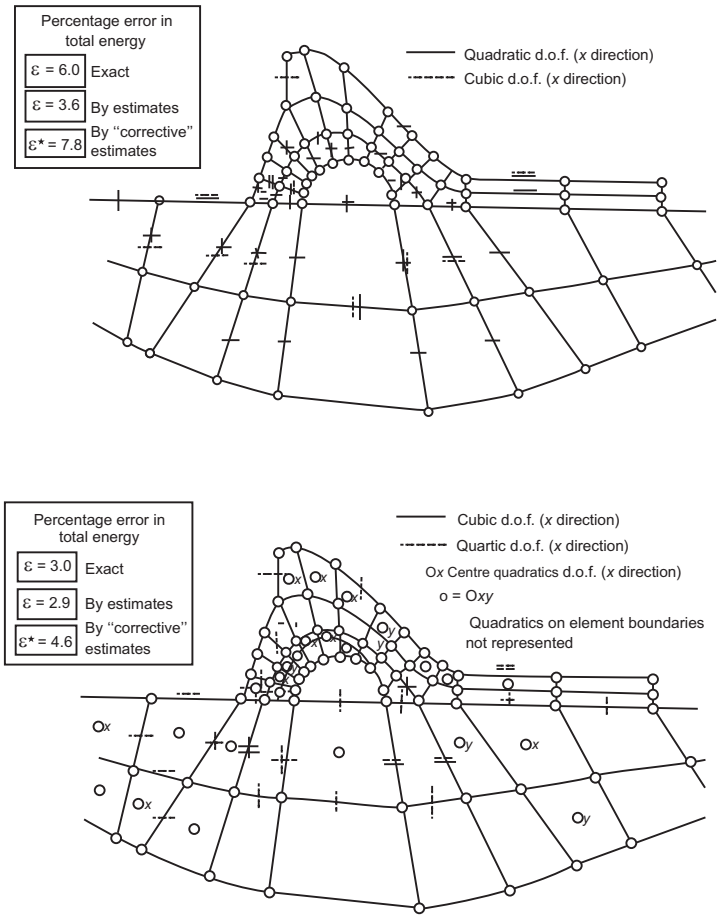



**FIGURE 16.17**

Adaptive solution of perforated dam by *p*-refinement: (a) stage three, 206 degrees of freedom and (b) stage five, 365 degrees of freedom.

At no stage of the *hp*-refinements have we used any of the estimators quoted in the previous chapter. However, their use would make the optimal mesh design at order $p$ possible, because the element error can be accurately estimated at order $p$. It will result in an optimal *hp*-refinement.

The two examples we have quoted above are reanalyzed using the alternative process described above and presented in Figs. 16.20 and 16.21. In both cases the final accuracy shows an error of less than 1% but it is noteworthy that the total number of degrees of freedom used with the second method is considerably less than that in the first and still achieves a nearly optimal rate of convergence.

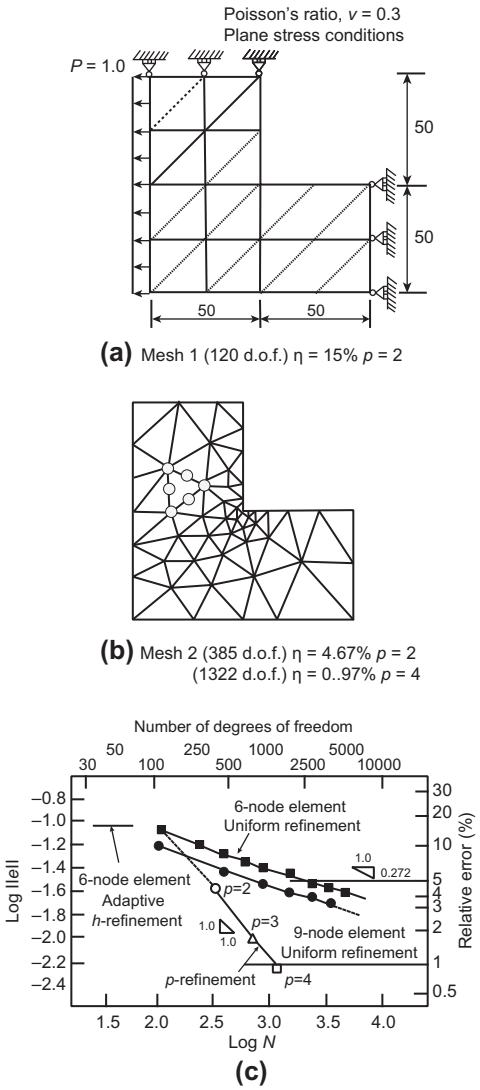



**FIGURE 16.18**

Solution of L-shaped domain by refinement (as defined in Example 16.4 of the previous section) using *h*-refinement and lower order elements followed by *p*-refinement. (a) Original mesh; (b) Quadratic triangles for 5% error and (c) *h-p* refinement. 1% accuracy reached with 1322 d.o.f.

We can conclude this section on *hp*-refinement with a final example where a highly singular crack domain is studied. Once again the second procedure is used showing in Fig. 16.22 a remarkable rate of convergence.

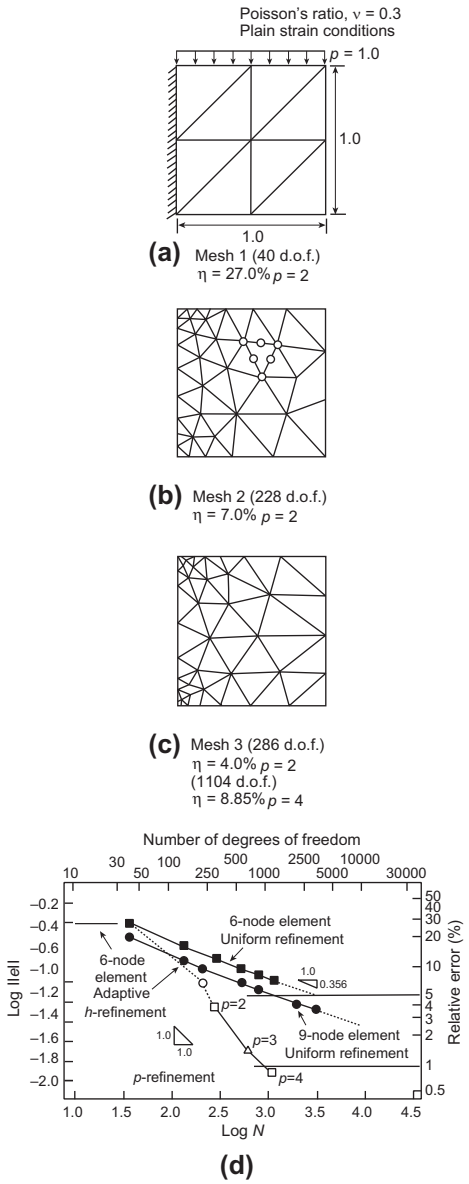
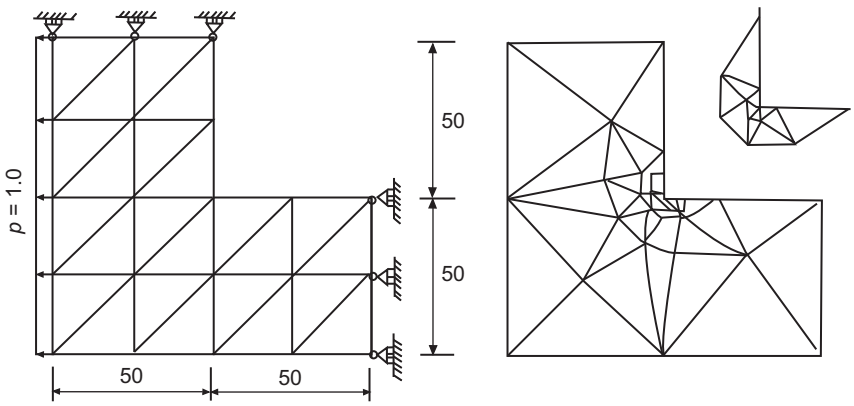



**FIGURE 16.19**

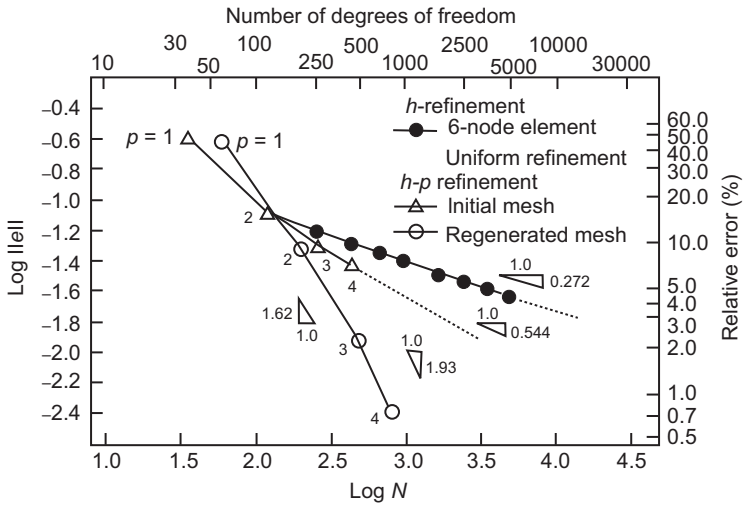
Solution of short cantilever by adaptive refinement using *h*-refinement and lower order elements followed by *p*-refinement: (a) Original mesh; (b) First refinement; (c) Second refinement and (d) *p*-refinement. 1% accuracy reached with 1104 d.o.f.




Mesh 1, for $p = 4$ d.o.f = 430 η = 6.72%

Mesh 2, for $p = 4$ d.o.f = 846 η = 0.76%

**(a)** **(b)**




Note: 1% accuracy reached with 846 d.o.f

**(c)**

**FIGURE 16.20**

Solution of L-shaped domain by *h-p* adaptive refinement using alternative procedure.

## 16.4 Concluding remarks

The methods of estimating errors and adaptive refinement which are described in this and the previous chapter constitute a very important tool for practical application of

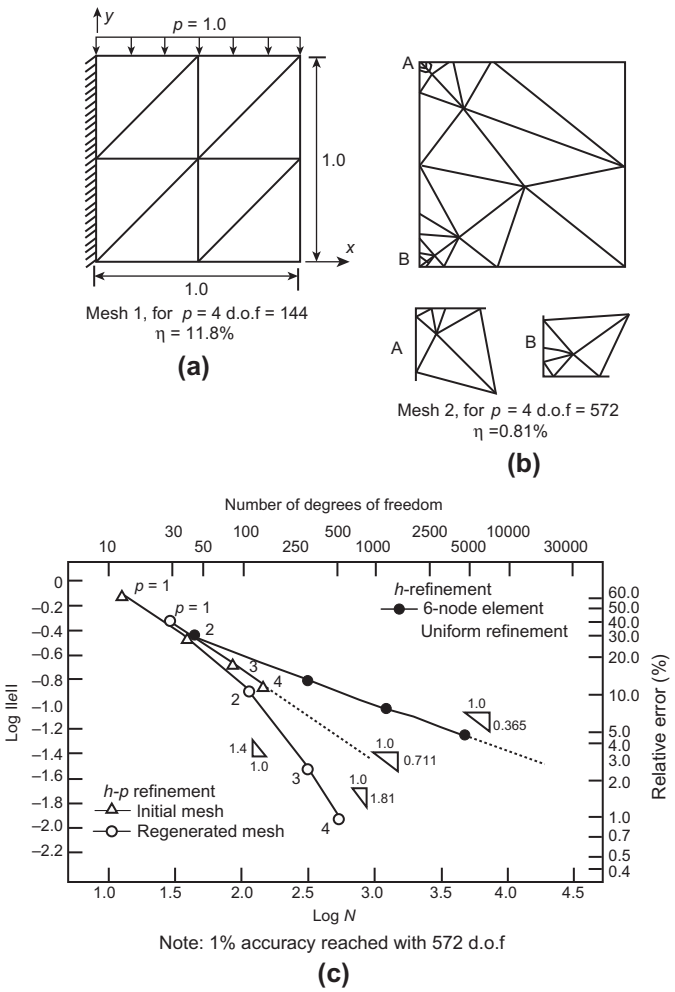



**FIGURE 16.21**
Solution of short cantilever by *h-p* adaptive refinement using alternative procedure.

finite element methods. The range of applications is large and we have only touched here upon the relatively simple range of linear elasticity and similar self-adjoint problems. We would like to reiterate that in other classes of problems many different norms or measures of error can be used and that for some problems the energy norm is not in fact "natural." A good example of this is given by problems of high-speed gas flow, where very steep gradients (shocks) can develop. The formulation of such problems is complex [31], but this is not necessary for the present argument. Applications of error estimates and adaptive refinement in this and many more areas are reported in Refs. [32, 33] and the reader is referred to these publications for additional details.

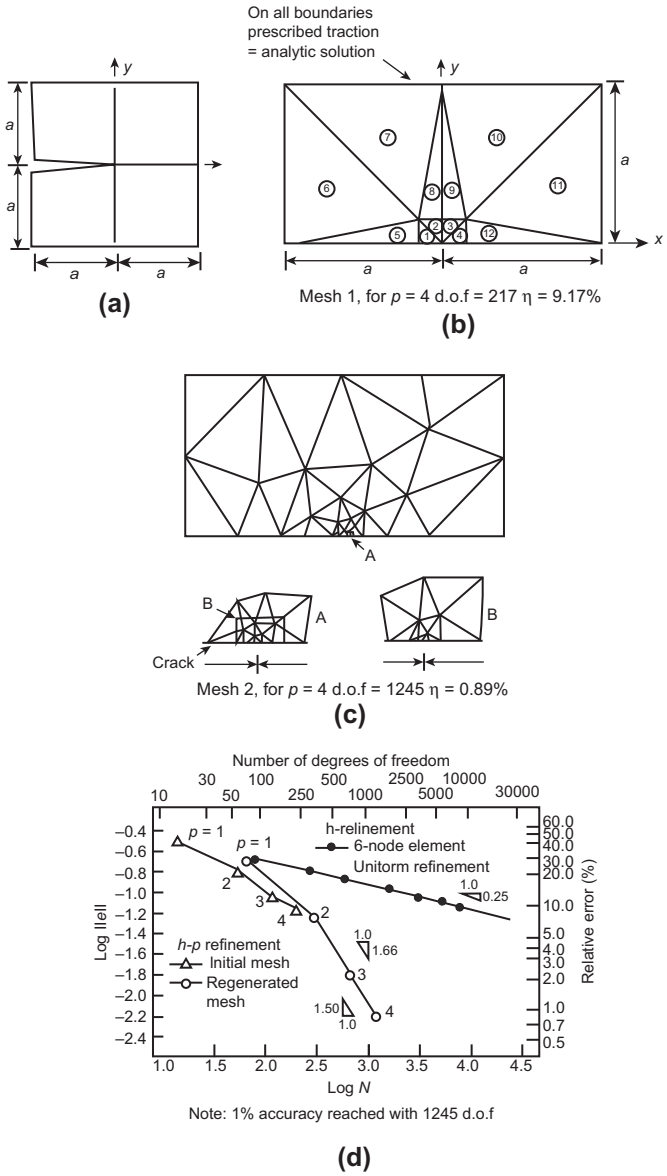
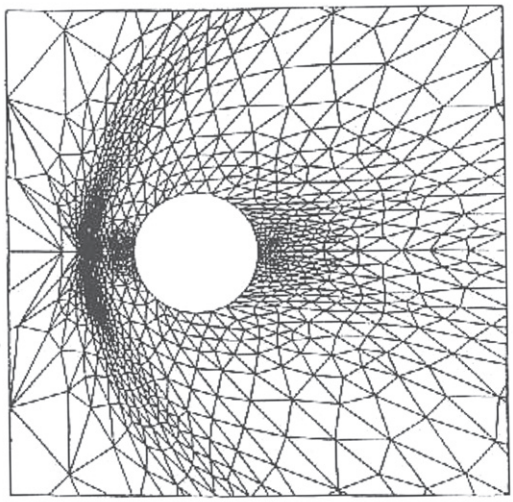
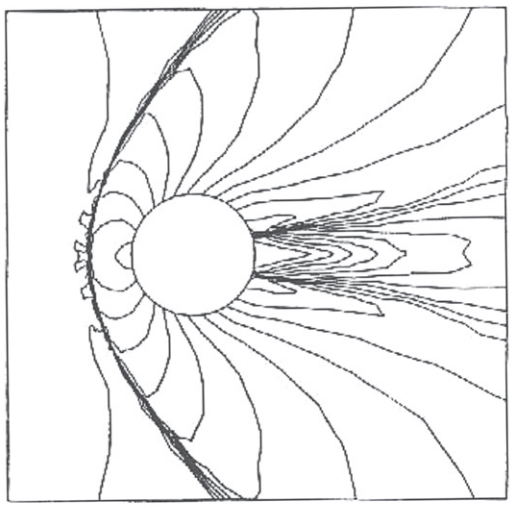



**FIGURE 16.22**
Adaptive *h-p* refinement for a singular crack using alternative procedure.

For problems in fluid mechanics discussed in Ref. [31] and similarly for problems of strain localization in plastic softening discussed in Ref. [34] no global norms can be used effectively. In such situations it is convenient to base the refinement on



(a)



(b)

**FIGURE 16.23**

Directional mesh refinement. Gas flow past a circular cylinder: Mach number 3. Third refinement mesh 709 nodes (1348 elements): (a) Local mesh and (b) Pressure coefficients.

the value of the maximum curvatures developed by the solution of **u**. On occasion an elongation of the elements will be used to refine the mesh appropriately. Such anisotropic mesh adaptation is especially effective when it is based on the recovery

type of error estimators [35–38]. Figure 16.23 shows a typical problem of shock capturing solved adaptively.

## 16.5 Problems

**16.1** Program development project: Implement a *mesh enrichment* algorithm (as described in Sections 16.2.1) in the solution system started in Problem 7.20 and extended in subsequent exercises. Assume the problem is given by the quasi-harmonic equation and modeled using three-node triangular elements. (Hint: Adapt the mesh generation program developed in Problems 6.32 and 6.33 to generate the mesh using the new coordinates resulting from enrichment.)

**16.2** Program development project: Implement an *adaptive mesh regeneration* algorithm (as described in Section 16.2 in the solution system started in Problem 7.20 and extended in subsequent exercises. Assume the problem is given by the quasi-harmonic equation and modeled using three-node triangular elements.

**16.3** Solve Example 7.5 using linear (three-node) elements. Follow the mesh refinement procedure described in Section 16.2 for adaptive $h$-refinement and show that the optimal rate of convergence of the finite element method can be attained when a prescribed accuracy is achieved.

**16.4** Solve Example 7.5 using quadratic (6-node) elements. Follow the mesh refinement procedure described in Section 16.2 for adaptive $h$-refinement and show that the optimal rate of convergence of the finite element method can be attained when prescribed accuracy is achieved.

**16.5** Program development project: Devise and implement in the solution system started in Problem 7.20 an $hp$-refinement strategy (see Section 16.3) to attain a prescribed accuracy. Assume the problem is given by the quasi-harmonic equation and hierarchical triangular elements are used to define the finite element $p$-models.

## References

[1] I. Babuška, C. Rheinboldt, *A-posteriori* error estimates for the finite element method, Int. J. Numer. Methods Eng. 12 (1978) 1597–1615.

[2] I. Babuška, C. Rheinboldt, Adaptive approaches and reliability estimates in finite element analysis, Comput. Methods Appl. Mech. Eng. 17 (18) (1979) 519–540.

[3] O.C. Zienkiewicz, J.Z. Zhu, A simple error estimator and adaptive procedure for practical engineering analysis, Int. J. Numer. Methods Eng. 24 (1987) 337–357.

[4] E. Oñate, G. Bugeda, A study of mesh optimality criteria in adaptive finite element analysis, Eng. Comput. 10 (1993) 307–321.

[5] J. Peraire, A.T. Patera, Bounds for linear-functional outputs of coercive partial differential equations: local indicators and adaptive refinement, in: Studies in Applied Mechanics, vol. 47, Elsevier, New York, 1998, pp. 129–147.
[6] J.T. Oden, S. Prudhomme, T.A. Westermann, J.M. Bass, Progress on practical methods of error estimation for engineering calculations, in: Proceedings of the Second European Conference on Computational Mechanics, Cracow, Poland, 2001 (ECCOMAS & IACM).
[7] T. Richter, A posteriori error estimation and anisotropy detection with dual-weighted residual method, Int. J. Numer. Methods Fluids 62 (2010) 90–118.
[8] K. Miller, R.N. Miller, Moving finite elements: I. SIAM J. Numer. Anal. 18 (1981) 1019–1032.
[9] B.J. Lucier, A moving mesh numerical method for hyperbolic conservation laws, Math. Comput. 46 (1986) 59–69.
[10] I.W. Johnson, A.J. Wathan, M.J. Baines, Moving finite element methods for evolutionary problems. II: Applications, J. Comput. Phys. 79 (1988) 270–297.
[11] W. Gui, I. Babuška, The *h*, *p* and *h-p* version of the finite element method in 1 dimension. Part 1: The error analysis of the *p*-version. Part 2: The error analysis of the *h*- and *h-p* version. Part 3: The adaptive *h-p* version, Numer. Math. 48 (1986) 557–683.
[12] B. Guo, I. Babuška, The *h-p* version of the finite element method. Part 1: The basic approximation results. Part 2: General results and applications, Comput. Mech. 1 (21–41) (1986) 203–226.
[13] I. Babuška, B. Guo, The *h-p* version of the finite element method for domains with curved boundaries, SIAM J. Numer. Anal. 25 (1988) 837–861.
[14] I. Babuška, B.Q. Guo, Approximation properties of the *hp* version of the finite element method, Comput. Methods Appl. Mech. Eng. 133 (1996) 319–349.
[15] L. Demkowicz, J.T. Oden, W. Rachowicz, O. Hardy, Toward a universal *h-p* adaptive finite element strategy. Part 1: Constrained approximation and data structure, Comput. Methods Appl. Mech. Eng. 77 (1989) 79–112.
[16] W. Rachowicz, J.T. Oden, L. Demkowicz, Toward a universal *h-p* adaptive finite element strategy. Part 3: Design of *h-p* meshes, Comput. Methods Appl. Mech. Eng. 77 (1989) 181–211.
[17] K.S. Bey, J.T. Oden, *hp*-version discontinuous Galerkin methods for hyperbolic conservation laws, Comput. Methods Appl. Mech. Eng. 133 (1996) 259–286.
[18] C.E. Baumann, J.T. Oden, A discontinuous *hp* finite element method for convection-diffusion problems, Comput. Methods Appl. Mech. Eng. 175 (1999) 311–341.
[19] P. Monk, On the *p* and *hp* extension of Nédélec's curl-conforming elements, J. Comput. Appl. Math. 53 (1994) 117–137.
[20] L.K. Chilton, M. Suri, On the selection of a locking-free *hp* element for elasticity problems, Int. J. Numer. Methods Eng. 40 (1997) 2045–2062.
[21] L. Vardapetyan, L. Demkowicz, *hp*-adaptive finite elements in electromagnetics, Comput. Methods Appl. Mech. Eng. 169 (1999) 331–344.

[22] O.C. Zienkiewicz, J.Z. Zhu, N.G. Gong, Effective and practical *h-p*-version adaptive analysis procedures for the finite element method, Int. J. Numer. Methods Eng. 28 (1989) 879–891.
[23] P.G. Ciarlet, The Finite Element Method for Elliptic Problems, North-Holland, Amsterdam, 1978.
[24] J. Peraire, M. Vahdati, K. Morgan, O.C. Zienkiewicz, Adaptive remeshing for compressible flow computations, J. Comput. Phys. 72 (1987) 449–466.
[25] J.Z. Zhu, O.C. Zienkiewicz, E. Hinton, J. Wu, A new approach to the development of automatic quadrilateral mesh generation, Int. J. Numer. Methods Eng. 32 (1991) 849–866.
[26] J.Z. Zhu, O.C. Zienkiewicz, Adaptive techniques in the finite element method, Commun. Appl. Numer. Math. 4 (1988) 197–204.
[27] D.W. Kelly, J.P. De, S.R. Gago, O.C. Zienkiewicz, I. Babuška, A posteriori error analysis and adaptive processes in the finite element method. Part I—Error analysis, Int. J. Numer. Methods Eng. 19 (1983) 1593–1619.
[28] J.P. De, S.R. Gago, D.W. Kelly, O.C. Zienkiewicz, I. Babuška, A posteriori error analysis and adaptive processes in the finite element method. Part II—Adaptive mesh refinement, Int. J. Numer. Methods Eng. 19 (1983) 1621–1656.
[29] B.A. Szabo, Mesh design for the *p* version of the finite element, Comput. Methods Appl. Mech. Eng. 55 (1986) 181–197.
[30] I. Babuška, B.A. Szabo, I.N. Katz, The *p* version of the finite element method, SIAM J. Numer. Anal. 18 (1981) 512–545.
[31] O.C. Zienkiewicz, R.L. Taylor, P. Nithiarasu, The Finite Element Method for Fluid Dynamics, seventh ed., Elsevier, Oxford, 2013.
[32] P. Ladevèze, J.T. Oden (Eds.), Advances in Adaptive Computational Methods in Mechanics, Studies in Applied Mechanics, vol. 47, Elsevier, 1998.
[33] S. Prudhomme, J.T. Oden (Eds.), Special Issue on Modeling Error Estimation and Adaptive Modeling, Computer Methods in Applied Mechanics and Engineering, Elsevier, 2011.
[34] O.C. Zienkiewicz, R.L. Taylor, D. Fox, The Finite Element Method for Solid and Structural Mechanics, seventh ed., Elsevier, Oxford, 2013.
[35] M. Picasso, Adaptive finite elements with large aspect ratio based on an anisotropic error estimator involving first order derivatives, Comput. Methods Appl. Mech. Eng. 196 (2006) 14–23.
[36] P. Sun, J. Xu, R.D. Russell, A new adaptive local mesh refinement algorithm and its application on fourth order thin film flow problem, J. Comput. Phys. 224 (2007) 1021–1048.
[37] P.E. Farrell, S. Micheletti, S. perotto, An anisotropic Zienkiewicz-Zhu type error estimator for 3D applications, Int. J. Numer. Methods Eng. 85 (2011) 671–692.
[38] R. Bois, M. Fortin, A. Fortin, A fully optimal anisotropic mesh adaptation method based on a hierarchical error estimator, Comput. Methods Appl. Mech. Eng. 209–212 (2012) 12–27.

CHAPTER

# 17 Automatic Mesh Generation

## 17.1 Introduction

In the previous chapters we have introduced various forms of elements and the procedures of using these elements in the computation of approximate solutions of a wide range of engineering problems. It is now obvious that the first step in the finite element computation is to discretize the problem domain into a union of elements. These elements could be of any one type or a combination of different types of those described in Chapters 5 and 6. The union of these elements is the so-called finite element mesh. The process of creating a finite element mesh is often termed as *mesh generation*.

Mesh generation has always been a time-consuming and error-prone process. This is especially true in the practical science and engineering computations, where meshes have to be generated for three-dimensional geometries of various levels of complexity. The attempt to create a fully automatic mesh generator, which is a particular mesh generation algorithm that is capable of generating valid finite element meshes over arbitrary domains, and needs only the information of the specified geometric boundary of the domain and the required distribution of the element size, started from the work of Zienkiewicz and Phillips [1] in the early 1970s. Since then many methodologies have been proposed and different algorithms have been devised in the development of automatic mesh generators.

The early proposed mesh generation methods, such as the isoparametric mapping method by Zienkiewicz and Phillips [1] and the transfinite mapping method by Gordon and Hall [2] discussed in Section 6.10, and the method of generating a mesh by solving various types of partial different equations as described by Thompson et al. [3,4], are often regarded as semi-automatic mesh generation methods. This is because in the mesh generation process the model domain has to be subdivided manually into simple subregions, i.e., multi-blocks, which are then mapped onto regular grids to produce a mesh. This manual process is tedious and occasionally difficult, particularly in the case of three-dimensional complex geometries. Such mesh generation procedures are complicated further by the requirement of varying mesh size distributions since the element sizes are controlled by the subdivision of the simple subregions. Thus, more

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00017-4

subregions are needed to generate a mesh which can accommodate changes in the desired element sizes from region to region. However, one of the main features of these mapping techniques is that, once the domain is decomposed into mappable subregions, the generation of the elements is much easier than any other methods. In addition, the elements generated by mapping methods usually have good shape and regular orientation. Mapping methods are often used to generate quadrilateral elements in two dimensions and hexahedral elements in three dimensions. Triangles, tetrahedra, and all other types of elements can be obtained by dividing quadrilaterals and hexahedra accordingly. These generated meshes are sometimes called *structured meshes*. Over the years, continuous efforts have been made to automate the mapping methods [5–10], although automatic decomposition of a complex domain into subregions seems to be a nontrivial task. Today, no fully automatic mesh generator using a mapping method has been achieved.

In contrast with mapping methods, in recent years concrete achievements have been made in the development of various algorithms for the automatic generation of the so-called *unstructured meshes*. Most of the unstructured mesh generation methods are designed for generating triangular elements in two dimensions and tetrahedral elements in three dimensions (known as simplex forms). These simplex forms lead to the simplest discretization of two- and three-dimensional domains of any shape, especially when meshes with varying element sizes in different regions of the domain are requested. A large number of automatic unstructured mesh generation algorithms have been proposed in the literature, but the most widely used algorithms are based on one or some kind of combination of the three fundamentally distinctive methods, which are the Delaunay triangulation method [11–20], the advancing front method [21–28], and tree methods [29–31] (the finite quadtree method in two dimensions and the finite octree method in three dimensions). By observing the fact that a quadrilateral can be formed by two triangles which share a common edge, the above-mentioned methods can be extended to automatically generate unstructured quadrilateral meshes in two dimensions. Although various algorithms have been proposed for unstructured hexahedral mesh generation [10,32–36], only octree-based approaches are currently automatic [37–43] consequently much research is still needed in this direction.

The automatic mesh generation process has been an active research subject since the early 1970s. The research literature on the subject is vast and different methodologies and techniques have been proposed. In this chapter, we are mainly concerned with the automatic mesh generation methods based on the advancing front method and the Delaunay triangulation method. These are the basis of many existing mesh generation programs and the basis of current research. We shall discuss the algorithmic procedures of the advancing front method in two dimensions and the Delaunay triangulation method in three dimensions. We shall also discuss curve and surface mesh generations. The reader is referred to Thompson et al. [5,44] for discussions on the development of semi-automatic multi-block mesh generation methods and to Frey and George [45] for other existing methods.

Before proceeding further on mesh generation schemes it is necessary to specify the kind of mesh we desire. Here we should give the following information.

1. The type of element and the number of nodes required on each.
2. The size of the desired element; here the minimum size of each element generally is specified.
3. Specification of regions of different material types or characteristics to be attached to a given element.
4. In some cases, the so-called stretch ratio if we wish to present elements which are elongated in some preferential direction. This is often needed for problems in fluid mechanics in regions where boundary layers and shocks are encountered.

Any and all of the above information has to be available at all points of the space in which elements are to be generated. It is often convenient to present this information as numbers attached to a *background mesh*, consisting say of elements of a linear kind, from which these values can be interpolated to any point in space. The procedure is important particularly if the use of adaptive refinement is considered—and here all mesh generation schemes must ensure that the input data contains this information. In adaptive refinement, in fact in general analysis, the background mesh will simply be the last mesh used for analysis of the problem and the refinement will proceed from there as this is the starting point for any new mesh to be developed.

## 17.2 Geometrical representation of the domain

In modern engineering design, nearly all three-dimensional geometries are created by computer aided design (CAD) systems. In the boundary representation (B-rep) of CAD systems for three-dimensional solids, the surfaces and curves are usually given in parametric forms represented by NURBS (non-uniform rational B-splines), which is currently a standard curve and surface representation in CAD systems. The faces of a solid are trimmed NURBS surfaces on which they are defined. The edges that connect faces are trimmed NURBS curves and are the boundary of the faces. Figure 17.1 illustrates a machine part, its NURBS boundary faces, and their connecting edges.

Before proceeding to the detailed discussion of mesh generation algorithms, we present a brief overview of the B-rep for the solid geometry in the form of NURBS curves and surfaces. The reader is referred to Piegl and Tiller [46], Rogers [47], and Farin [48] for more detailed discussion of the geometric characteristics of NURBS curves and surfaces.

### 17.2.1 Curve representation

The NURBS curves generated by CAD systems for each surface boundary edge of the solid are piecewise rational functions expressed by a vector valued function in parametric form, using a parameter $u$, as

$$\begin{aligned}\mathbf{x}(u) &= \begin{bmatrix} x(u) & y(u) & z(u) \end{bmatrix}^{\mathrm{T}} \\ &= \frac{\sum_{a=0}^{n} \mathrm{w}_a \, N_a^{(p)}(u) \, \tilde{\mathbf{x}}_a}{\sum_{a=0}^{n} \mathrm{w}_a N_a^{(p)}(u)} = \sum_{a=0}^{n} R_a^{(p)}(u) \, \tilde{\mathbf{x}}_a \end{aligned} \tag{17.1}$$

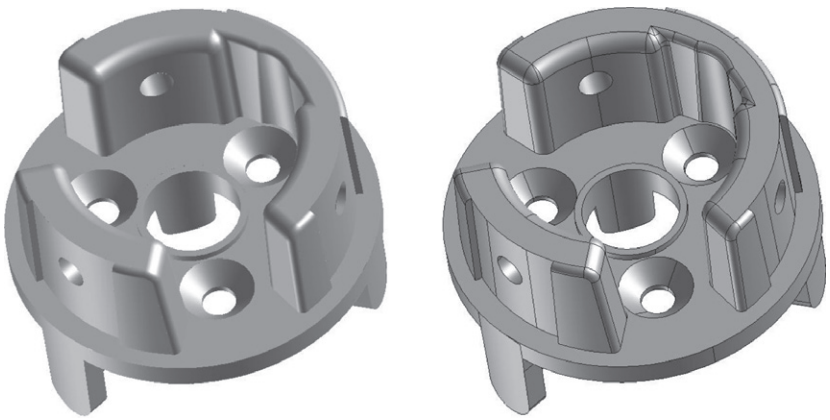



**FIGURE 17.1**
(a) A machine part. (b) Boundary faces and boundary edges of the machine part.

where

$$R_a^{(p)}(u) = \frac{\mathrm{w}_a\, N_a^{(p)}(u)}{\sum_{a=0}^{n} \mathrm{w}_a N_a^{(p)}(u)} \tag{17.2}$$

are the $n+1$ NURBS basis functions, $p$ is the order of the NURBS, $\tilde{\mathbf{x}}_a$ are the *control points* of the control polygon that defines the geometry of the NURBS, $N_a^{(p)}(u)$ are B-spline basis functions, and $\mathrm{w}_a$ are weights corresponding to the B-spline basis functions. It is observed that the construction of the NURBS basis function is based on B-spline basis functions.

The B-spline basis functions are defined by the *knot vector*

$$\mathbf{U} = \{u_0, u_1, \ldots, u_m\} \tag{17.3}$$

that consists of a nondecreasing sequence of coordinates (*knots*) $u_a \leq u_{a+1}$ in the parametric space. When knots are equally distributed in the parametric space, the knot vector is termed *uniform*, otherwise it is referred to as *nonuniform*. Multiple knots can be placed at the same location in the parametric space to control the continuity of the basis functions. Each repeated knot lowers continuity at the location by one order.

The $p$th order B-spline basis functions $N_a^{(p)}(u)$ are generated by the Cox-de Boor recursion formula [49,50], as

$$N_a^{(0)}(u) = \begin{cases} 1, & \text{if } u_a \leq u \leq u_{a+1} \\ 0, & \text{otherwise} \end{cases} \tag{17.4a}$$

and

$$N_a^{(p)}(u) = \left(\frac{u - u_a}{u_{a+p} - u_a}\right) N_a^{(p-1)}(u) + \left(\frac{u_{a+p+1} - u}{u_{a+p+1} - u_{a+1}}\right) N_{a+1}^{(p-1)}(u) \tag{17.4b}$$

in which any quotient $0/0$ is assumed to be zero. For open knot vectors in which the first and last knot values have $p+1$ multiplicity, the number of knots $m+1$, and the number of B-spline basis functions $n+1$ are related by

$$m = n + p + 1 \tag{17.5}$$

Figure 17.2 shows quadratic B-spline basis functions for a nonuniform open knot vector which has multiple knot values at $u_a = 4$. The curve is $C_1$ continuous except at $u_a = 4$ where continuity is $C_0$.

The NURBS basis functions $R_a^{(p)}(u)$ possess important properties such as partition of unity, nonnegativity, local support, linear independence, and controllable continuity. These properties determine the geometric characteristics of NURBS curves [46].

It is easy to verify that the Bèzier curves introduced in Section 6.10 are special cases of NURBS curves. An example of a cubic NURBS curve with its control polygon is illustrated in Fig. 17.3.

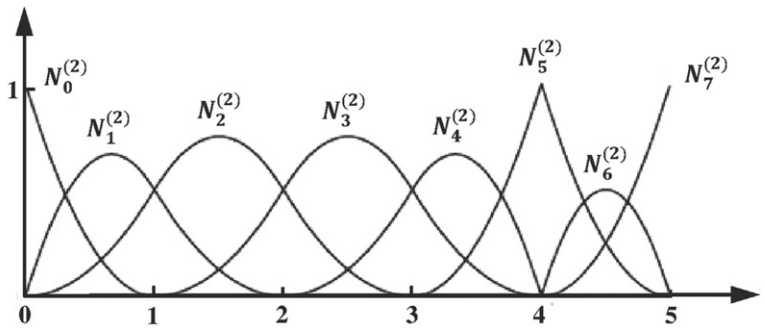



**FIGURE 17.2**
Quadratic B-spline basis functions with nonuniform, open knot vector $\mathbf{U} = \{0,0,0,1,2,3,4,4,5,5,5\}$.

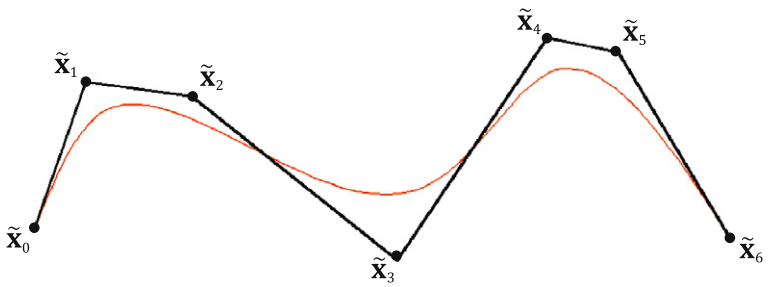



**FIGURE 17.3**
A cubic NURBS curve with control points and control polygon.

### 17.2.2 Surface representation

NURBS surfaces exported by CAD systems to represent the surface of the solid are bivariate vector valued piecewise rational functions that are defined by two parameters $v$ and $w$. The functions are of order $p$ in the $v$ direction and order $q$ in the $w$ direction of parametric space. The tensor product NURBS surface has the form of

$$\begin{aligned}\mathbf{x}(v, w) &= \frac{\sum_{a=0}^{n}\sum_{b=0}^{m} \mathrm{w}_{ab}\, N_a^{(p)}(v) N_b^{(q)}(w)\, \tilde{\mathbf{x}}_{ab}}{\sum_{a=0}^{n}\sum_{b=0}^{m} \mathrm{w}_{ab} N_a^{(p)}(v) N_b^{(q)}(w)} \\ &= \sum_{a=0}^{n}\sum_{b=0}^{m} R_{ab}(v, w)\, \tilde{\mathbf{x}}_{ab}\end{aligned} \tag{17.6}$$

where

$$R_{ab}(v, w) = \frac{\mathrm{w}_{ab}\, N_a^{(p)}(v) N_b^{(q)}(w)}{\sum_{a=0}^{n}\sum_{b=0}^{m} \mathrm{w}_{ab} N_a^{(p)}(v) N_b^{(q)}(w)} \tag{17.7}$$

are the $(n+1)(m+1)$ NURBS surface basis functions, $\tilde{\mathbf{x}}_{ab}$ are the control points of the three-dimensional control net, $\mathrm{w}_{ab}$ are the weights, and $N_a^{(p)}(v) N_b^{(q)}(w)$ are the tensor product B-spline basis functions defined on knot vectors formed in the $v$ and $w$ directions, respectively.

It is noted that $R_{ab}(v, w)$ share all the important analytic and geometric properties of $R_a^{(p)}(u)$. It is also easy to show that the Bèzier surfaces discussed in Section 6.10 are special cases of NURBS surfaces. Figure 17.4 depicts one of the NURBS surfaces and its control net for the machine part shown in Fig. 17.1. An illustration of a trimmed NURBS surface with curved boundary defined on the parametric plane $[v_0, v_m] \times [w_0, w_n]$ is shown in Fig. 17.5.

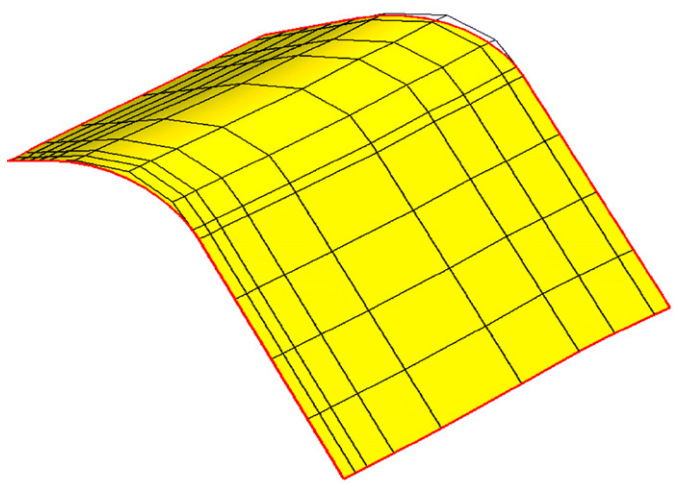


**FIGURE 17.4**
NURBS surface with control net.

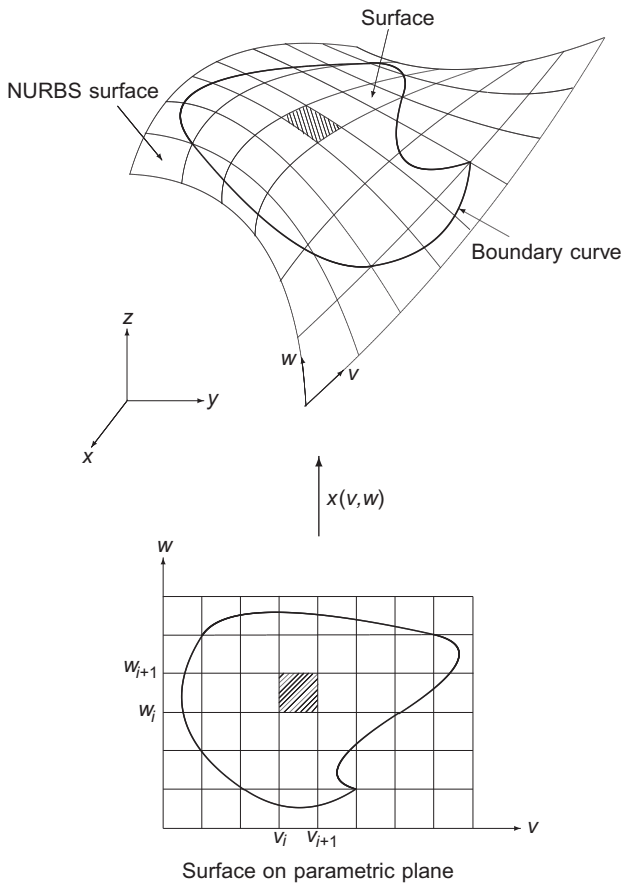



**FIGURE 17.5**
A trimmed surface on a parametric plane and its image on a NURBS surface.

All the geometric data used to define NURBS curves and surfaces in the B-rep of CAD systems for engineering design, such as knot vectors, control points, weights, etc., will be utilized as input data for automatic mesh generation to produce meshes according to user-specified distribution of element sizes and directional orientation of the elements.

## 17.3 Two-dimensional mesh generation: Advancing front method

Conceptually, the advancing front method is one of the simplest mesh generation processes. The element generation algorithm, starting from an initial "front" formed

from the specified boundary of the domain, generates elements, one by one, as the front advances into the region to be discretized until the whole domain is completely covered by elements.

The representative element generation algorithms of the advancing front method include the procedure introduced by Lo [21], which constructs a triangulation over a set of *a priori* generated points inside of the domain, and the methodology developed be Peraire et al. [22], which generates points and triangular elements at the same time.

One of the main distinctions of the mesh generation algorithm of Peraire et al. is that the geometrical characteristics of the mesh, such as the location of the newly generated point, the shape of the element and the size of the element, can be controlled during the mesh generation process, due to the fact that individual points and elements are generated simultaneously. With the assistance of a background mesh, which is utilized to define the geometrical characteristics of the mesh, nonuniform distribution of element sizes, often required in highly graded meshes, can be achieved throughout the domain according to particular specifications. Any directional orientation of the elements can also be realized by introducing stretches in certain specified directions. These features are particularly desirable for the nearly optimal mesh design in adaptive analysis (viz. Chapter 16) and adaptive computations of fluid dynamics as discussed in Ref. [51].

The mesh generation procedure includes three main steps:

- Node generation along boundary edges to form a discretized boundary of the domain
- Element (and node) generation within the discretized boundary
- Element shape enhancement to improve the quality of the mesh

Before we proceed to the discussion of mesh generation procedures, the geometrical representation of the two-dimensional domain and the geometrical characteristics of the two-dimensional mesh are introduced.

### 17.3.1 Geometrical representation of two-dimensional domain

A general two-dimensional domain is defined by its boundary which consists of a closed loop of curved boundary segments (viz. Fig. 17.6). A curved boundary segment in two dimensions is represented by the same NURBS curve described in Section 17.2.1 except that the curve is now planar and the vector valued function is in the form of

$$\mathbf{x}(u) = \begin{bmatrix} x(u) & y(u) \end{bmatrix}^{\mathrm{T}} \tag{17.8}$$

The collection of all boundary edges, following a specific sequence that is convenient for mesh generation, forms the complete boundary of the domain. For the advancing front method, the sequence of exterior boundary edges is usually in a counterclockwise order, but, for interior boundary edges, is set in a clockwise order, i.e., the domain to be discretized always has an interior area situated to the left of the boundary edges. Figure 17.6 shows the direction of the boundary edges of a typical domain.

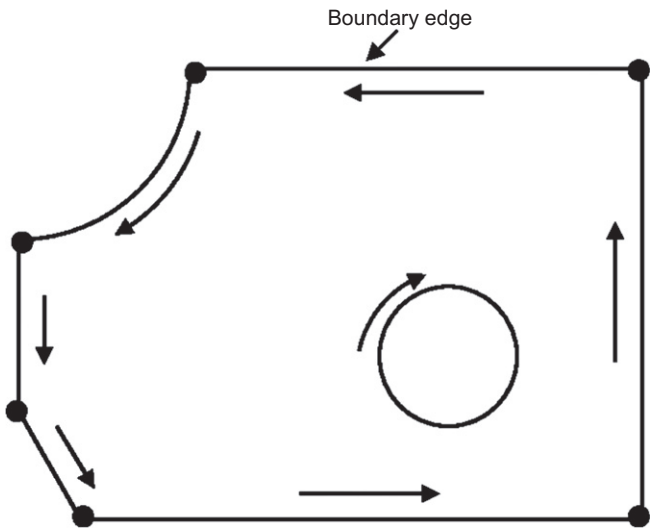



**FIGURE 17.6**
Boundary edge and orientation of a typical domain.

### 17.3.2 Geometrical characteristics of the mesh

The geometrical characteristics of the mesh such as element size, element shape, and element orientation are represented by means of mesh parameters which are spatial functions. The mesh parameters include two orthogonal directions defined by unit vectors $\boldsymbol{\alpha}_i$ $(i = 1, 2)$ and the associated element sizes $h_i$ $(i = 1, 2)$ as illustrated in Fig. 17.7. The orthogonal directions $\boldsymbol{\alpha}_i$ $(i = 1, 2)$ describe the directions of element stretching. A mesh with stretched elements in certain directions is only necessary when an anisotropic mesh is desired, otherwise the stretching directions are set to be constant unit vectors in the coordinate directions and the related element sizes are set to be equal, i.e., $h_1 = h_2$. In this case, the generated elements will not be stretched in any direction and an isotropic mesh type will be generated.

#### 17.3.2.1 *Background mesh*

The background mesh may be represented by simple triangular elements and is employed to accurately control the distribution of the geometrical characteristics on the new mesh. A piecewise linear distribution of the mesh parameters (mainly element sizes and stretching as discussed above) is represented by data assigned to nodes of the background mesh. Values of the mesh parameters at any point inside the domain or on the boundary of the domain can be obtained by linear interpolation. There is no requirement that the background mesh precisely represent the geometry, but it should completely cover the domain to be meshed. The number of the elements and the position of the nodes in the background mesh are chosen so that the mesh parameters can be

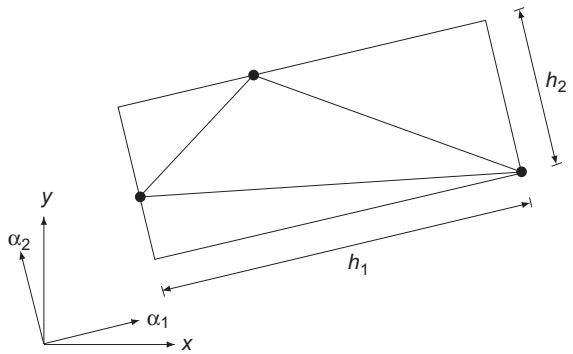



**FIGURE 17.7**
Mesh parameters in two dimensions.

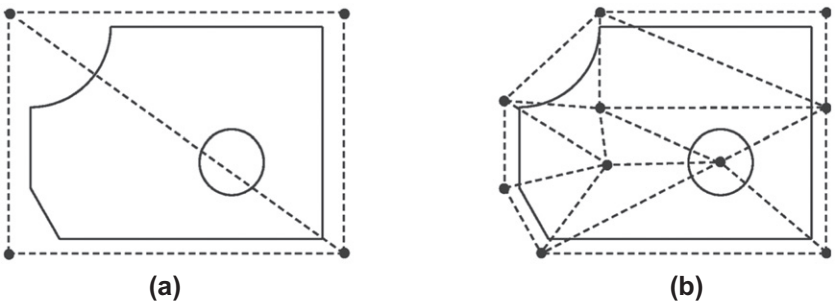



**FIGURE 17.8**
Background meshes for a typical domain. (a) Two triangles are used in the background mesh to represent a uniform distribution of the mesh parameters. (b) Eleven triangles are used in the background mesh to generate a graded mesh.

approximated in a satisfactory manner. One or two background elements will be sufficient if a uniform (isotropic) distribution of the element sizes $h_i\,(i = 1, 2)$ is required. Examples of background meshes for a given domain are illustrated in Fig. 17.8.

### 17.3.2.2 *Geometrical transformation of the mesh*

In order to simplify the mesh generation process, a symmetric transformation matrix $\mathbf{T}$, defined by the mesh parameters, is introduced and has the form

$$\mathbf{T}(\mathbf{x}) = \sum_{i=1}^{2} \frac{1}{h_i} \boldsymbol{\alpha}_i \boldsymbol{\alpha}_i^{\mathrm{T}}, \quad \boldsymbol{\alpha}_i = \begin{Bmatrix} \alpha_{1i} \\ \alpha_{2i} \end{Bmatrix} \tag{17.9}$$

It is easy to verified that the local transformation

$$\mathbf{x}' = \mathbf{T}\mathbf{x} \tag{17.10}$$

is in fact imposing two scaling operations with factor $1/h_i$ in each of the corresponding directions $\boldsymbol{\alpha}_i$. Figure 17.9 illustrates the effect of the transformation $\mathbf{T}$ on a triangle formed by nodes $abc$ in the coordinate system $(x, y)$ to form the triangle $a'b'c'$ in the coordinate system $(x', y')$. This demonstrates that, at a particular point, the transformation $\mathbf{T}$ maps a triangle with element size $h_i$ $(i = 1, 2)$, formed in the

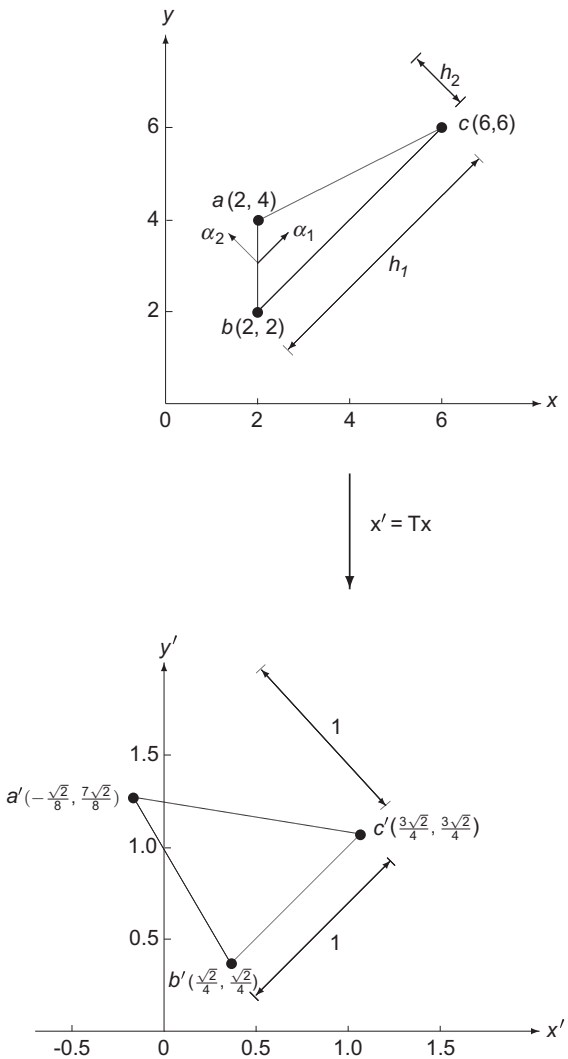



**FIGURE 17.9**

An irregular triangle $abc$ in coordinate system $(x, y)$ mapped to a regular triangle $a'b'c'$ in coordinate system $(x', y')$.

neighborhood of the point into a *normalized space* $(x', y')$, in which the triangular elements are approximately equilateral.

**Example 17.1.** Transformation of a triangle

As an example, the details of the coordinates transformation shown in Fig. 17.9 are given as follows.
From the coordinates of nodes $a$, $b$, and $c$, we can easily find that at node $b$

$$\boldsymbol{\alpha}_1 = \begin{Bmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{Bmatrix} \quad \text{and} \quad \boldsymbol{\alpha}_2 = \begin{Bmatrix} -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{Bmatrix}$$

with the associated element sizes $h_1 = 4\sqrt{2}$ and $h_2 = \sqrt{2}$.
The transformation matrix $\mathbf{T}$ at node $b$ is computed, using Eq. (17.9), as

$$\mathbf{T} = \frac{\sqrt{2}}{16}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} + \frac{\sqrt{2}}{4}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

Applying $\mathbf{T}$ to node $a$, $b$, $c$ results in

$$\mathbf{x}_{a'} = \mathbf{T}\mathbf{x}_a = \frac{\sqrt{2}}{8}\begin{Bmatrix} -1 \\ 7 \end{Bmatrix}, \quad \mathbf{x}_{b'} = \mathbf{T}\mathbf{x}_b = \frac{\sqrt{2}}{4}\begin{Bmatrix} 1 \\ 1 \end{Bmatrix}, \quad \text{and}$$
$$\mathbf{x}_{c'} = \mathbf{T}\mathbf{x}_c = \frac{\sqrt{2}}{4}\begin{Bmatrix} 3 \\ 3 \end{Bmatrix}$$

These are the nodal positions for the triangle $a'b'c'$ in the coordinate system $(x', y')$.

### 17.3.3 Triangular mesh generation

Among all the steps in the mesh generation process, we are particularly concerned with the procedure for element generation, which include node and side generation on the boundary curve and triangular element generation inside of the two-dimensional domain [22].

#### *17.3.3.1 Boundary node generation*

The boundary of the domain will be discretized into a polygon which will form the element edges, herein called "sides." The sides are defined by nodes generated on the NURBS curves that represent the boundary edges. The nodes will be generated along the curve edge and expressed by their parametric positions. The coordinates of the nodes in the two-dimensional domain are determined using Eq. (17.8). The algorithmic procedure of the boundary node generation is described in the following.

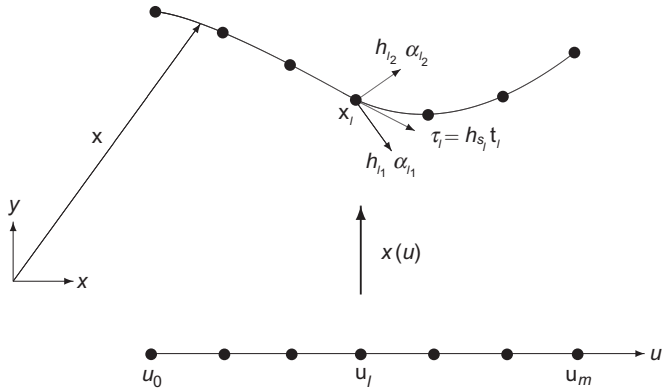



**FIGURE 17.10**
Sampling points and mesh parameters on the curve.

1. For a curved edge with typical length $L$, a set of sampling points $\mathbf{x}_l = \mathbf{x}(u_l)$ $(l = 0, 1, 2, \ldots, m)$ is first placed along the curve with parameters $u_l$ uniformly distributed as shown in Fig. 17.10. The unit tangent vector of the curve is determined at each of the sampling point as

$$\mathbf{t}_l = [t_{1_l} \quad t_{2_l}]^{\mathrm{T}} \tag{17.11}$$

where

$$t_{1_l} = \frac{x_{,u_l}}{\sqrt{x_{,u_l}^2 + y_{,u_l}^2}} \quad \text{and} \quad t_{2_l} = \frac{y_{,u_l}}{\sqrt{x_{,u_l}^2 + y_{,u_l}^2}} \tag{17.12}$$

and

$$\mathbf{x}_{,u_l} = [x_{,u_l} \quad y_{,u_l}]^{\mathrm{T}}, \quad x_{,u_l} = \left.\frac{\mathrm{d}x}{\mathrm{d}u}\right|_{u_l} \tag{17.13}$$

2. The mesh parameters $\boldsymbol{\alpha}_{l_i}$ and $h_{l_i}$ $(i = 1, 2)$ are computed at each sampling point by interpolation from their values assigned on the background mesh. The transformation matrix $\mathbf{T}_l$ is formed accordingly at each of the sampling point.
3. In order to find the position of the new nodes on the curve, an element size distribution function needs to be determined along the curve edge.
Let $h_{s_l}$ denote the element size at the arc length $s_l$ corresponding to the sampling point $\mathbf{x}_l$; a vector of length $h_{s_l}$ in the tangent direction is defined as

$$\boldsymbol{\tau}_l = h_{s_l} \mathbf{t}_l \tag{17.14}$$

Apply transformation $\mathbf{T}_l$ and assume that $\mathbf{T}_l$ maps $\boldsymbol{\tau}_l$ to a vector $\mathbf{T}\boldsymbol{\tau}_l$ in the normalized space with unit length, i.e.,

$$\mathbf{T}_l \boldsymbol{\tau}_l = \mathbf{T}_l (h_{s_l} \boldsymbol{\tau}_l) = h_{s_l} \mathbf{T}_l \boldsymbol{\tau}_l \tag{17.15}$$

with

$$\sqrt{(\mathbf{T}_l\boldsymbol{\tau}_l)^{\mathrm{T}}(\mathbf{T}_l\boldsymbol{\tau}_l)} = h_{s_l}\sqrt{(\mathbf{T}_l\mathbf{t}_l)^{\mathrm{T}}(\mathbf{T}_l\mathbf{t}_l)} = 1 \tag{17.16}$$

Thus, the curvilinear element size $h_{s_l}$ $(l = 0, 1, 2, \ldots, m)$ at the sampling points along the curve in the direction of the tangent is

$$h_{s_l} = \frac{1}{\sqrt{(\mathbf{T}_l\mathbf{t}_l)^{\mathrm{T}}(\mathbf{T}_l\mathbf{t}_l)}} = \frac{1}{\sqrt{\mathbf{t}_l^{\mathrm{T}}\mathbf{C}_l\mathbf{t}_l}} \tag{17.17}$$

where

$$\mathbf{C} = \mathbf{T}^{\mathrm{T}}\mathbf{T} = \sum_{i=1}^{2}\frac{1}{h_i^2}\boldsymbol{\alpha}_i\boldsymbol{\alpha}_i^{\mathrm{T}} \tag{17.18}$$

is the two-dimensional matrix of *Euclidean metric tensor* in the normalized space.

**4.** Assume that $\mathbf{T}_l$ is a constant matrix in the neighborhood of $\mathbf{x}_l$; it is observed, from Eq. (17.12), that $h_{s_l}$ is a function of the parameter $u$. A continuous element size distribution function may be achieved by a piecewise linear interpolation of the nodal values $h_{s_l}$ using (Lagrangian) finite element shape functions along the arc length of the curve

$$h(u) = \sum_{l=0}^{m} h_{s_l}N_l(u) \tag{17.19}$$

The element density function, i.e., the number of elements per length scale along the curve, is defined as $1/h(u)$.

**5.** The total number of the sides $N$ to be generated along the curved edge need to be consistent with the specified element size, which is now represented by the element density function. Therefore, $N$ is taken to be the nearest integer to

$$A = \int_0^L \frac{1}{h(u)}\mathrm{d}s = \int_{u_0}^{u_m}\frac{1}{h(u)}\sqrt{(x_{,u})^2 + (y_{,u})^2}\mathrm{d}u \tag{17.20}$$

where $A$ is the ideal number of the sides that should have been created on the boundary curve edge. However, $A$ is in general not an integer. To measure how close $N$ is to $A$, a *consistency index* is defined as

$$\theta = \frac{A}{N} = \frac{1}{N}\int_{u_0}^{u_m}\frac{1}{h(u)}\sqrt{(x_{,u})^2 + (y_{,u})^2}\mathrm{d}u \tag{17.21}$$

Because the positions of the nodes at the end of the edge $\mathbf{x}_0 = \mathbf{x}(u_0)$ and $\mathbf{x}_m = \mathbf{x}(u_m)$ are already known, $(N - 1)$ new nodes generated.

**6.** Assume that every node on the boundary edge is generated with the same consistency index $\theta$. The position of a particular new node $n_k$ $[k = 1, 2, \ldots, (N - 1)]$, represented by its parametric position $u_k$ on the boundary curve can be computed as

$$\theta = \theta_k = \frac{1}{k}\int_{u_0}^{u_k}\frac{1}{h(u)}\sqrt{(x_{,u})^2 + (y_{,u})^2}\mathrm{d}u \tag{17.22}$$

and similarly, the position of new node $n_{k+1}$ is given by

$$\theta = \theta_{k+1} = \frac{1}{k+1}\int_{u_0}^{u_{k+1}} \frac{1}{h(u)}\sqrt{(x_{,u})^2 + (y_{,u})^2}\mathrm{d}u \tag{17.23}$$

From $\theta_k = \theta_{k+1} = \theta$, we obtain the parametric position of node $n_{k+1}$ computed consecutively as

$$\theta = \int_{u_k}^{u_{k+1}} \frac{1}{h(u)}\sqrt{(x_{,u})^2 + (y_{,u})^2}\mathrm{d}u \tag{17.24}$$

where $k = 0, 1, 2, \ldots, (N-2)$. In general, Eq. (17.24) can be solved iteratively for $u_{k+1}$. For example, writing Eq. (17.24) in the form of

$$F(u_{k+1}) = \int_{u_k}^{u_{k+1}} \frac{1}{h(u)}\sqrt{(x_{,u})^2 + (y_{,u})^2}\mathrm{d}u - \theta = 0 \tag{17.25}$$

a Newton iterative process results in

$$u_{k+1}^{j+1} = u_{k+1}^{j} - \frac{h\left(u_{k+1}^{j}\right)}{\sqrt{\left(x_{,u_{k+1}^{j}}\right)^2 + \left(y_{,u_{k+1}^{j}}\right)^2}}\left[F\left(u_{k+1}^{j}\right)\right] \tag{17.26}$$

for $j = 0, 1, 2, \ldots$ with initial value $u_{k+1}^{0} = u_k$.

**7.** Finally, the position of the new points are mapped onto the boundary curve using Eq. (17.8).

The generation of the boundary nodes will be performed edge by edge following the above procedure. The boundary of the domain is finally discretized and transformed to a union of straight line sides formed by connecting the consecutive boundary nodes.

**Example 17.2.** Quarter circle

To verify the above procedure, the node generation technique is applied to a curve edge representing a quarter of a circle shown in Fig. 17.11a. To simplify the presentation and to demonstrate that the node generation procedure is independent of the curve representation of the boundary edge, we chose not to use NURBS to represent the curve, but use its curve, parametric expression in the form of

$$x = 4\cos(u), \quad y = 4\sin(u), \quad 0 \le u \le \frac{\pi}{2} \tag{17.27}$$

The mesh parameters $\boldsymbol{\alpha}_i$ and $h_i (i = 1, 2)$ are chosen to be independent of their spatial positions; the background mesh is therefore not required. We shall follow each step of the boundary node generation procedure to create nodes in accordance with the specified element size.

**1.** Three sampling points $\mathbf{x}(u_l)$ $(l = 0, 1, 2)$ are located with $u_l$ equally distributed at

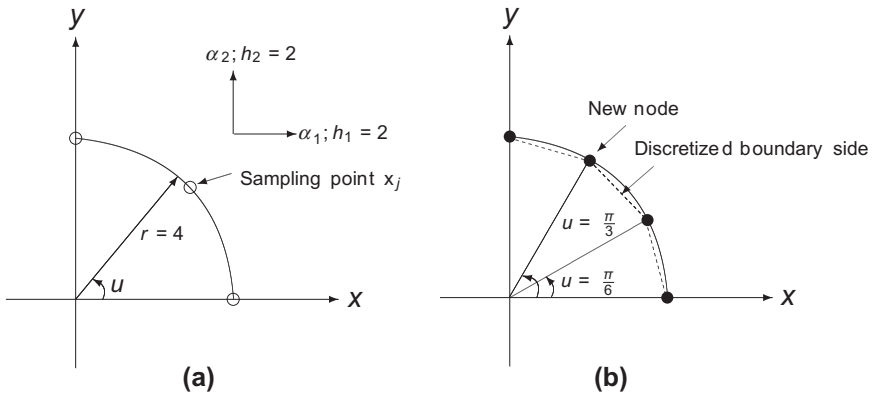



**FIGURE 17.11**
Node generation on a curve. (a) Description of the curve and mesh parameters. (b) Nodes generated on the curve and sides formed by nodes.

$$u_0 = 0, \quad u_1 = \frac{\pi}{4}, \quad u_2 = \frac{\pi}{2}$$

The unit tangent vector to the curve is, by noting Eq. (17.12),

$$\mathbf{t} = \begin{Bmatrix} -\sin(u) \\ \cos(u) \end{Bmatrix}$$

and

$$\mathbf{t}(u_0) = \begin{Bmatrix} 0 \\ 1 \end{Bmatrix}, \quad \mathbf{t}(u_1) = \begin{Bmatrix} -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{Bmatrix}, \quad \mathbf{t}(u_2) = \begin{Bmatrix} 1 \\ 0 \end{Bmatrix}$$

**2.** The stretching directions are chosen as depicted in Fig. 17.11a, i.e.,

$$\boldsymbol{\alpha}_1 = \begin{Bmatrix} 1 \\ 0 \end{Bmatrix}, \quad \boldsymbol{\alpha}_2 = \begin{Bmatrix} 0 \\ 1 \end{Bmatrix}$$

The corresponding element sizes are set to be constant and have the values of $h_1 = h_2 = 2$. That is, we are looking for a uniform discretization on the edge. As a result of our choice of mesh parameters, the transformation matrix $\mathbf{T}$ is a constant matrix and at all sampling points

$$\mathbf{T} = \frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \frac{1}{2}\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

**3.** Applying $\mathbf{T}$ to tangent vector $\mathbf{t}(u_0)$, we have

$$\mathbf{T}\,\mathbf{t}(u_0) = \frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{Bmatrix} 0 \\ 1 \end{Bmatrix} = \frac{1}{2}\begin{Bmatrix} 0 \\ 1 \end{Bmatrix}$$

The mesh size at $\mathbf{x}(u_0)$ along the curve is, using Eq. (17.17), $h_{s_0} = 2$. Similarly, as expected, we have

$$\mathbf{T}\,\mathbf{t}(u_1) = \frac{\sqrt{2}}{4}\begin{Bmatrix} -1 \\ 1 \end{Bmatrix}, \quad h_{s_1} = 2, \quad \text{and}$$
$$\mathbf{T}\,\mathbf{t}(u_2) = -\frac{1}{2}\begin{Bmatrix} 1 \\ 0 \end{Bmatrix}, \quad h_{s_2} = 2$$

**4.** Each element size is obtained by linear interpolation

$$h(u) = \sum_{l=0}^{2} h_{s_l} N_l(u) = 2$$

because $h_{s_l}$ $(l = 0, 1, 2)$ are constant. The element density function is

$$\frac{1}{h(u)} = \frac{1}{2}$$

**5.** The integration of the element density function, using Eq. (17.20), gives the ideal number of sides, as

$$A = \int_0^{\pi/2} \frac{1}{2}\sqrt{16\sin^2(u) + 16\cos^2(u)}\mathrm{d}u = \pi$$

The nearest integer $N$ to $\pi$ is 3, i.e., there should be three sides being generated along the curve. In addition to the nodes at the end of the curve, two new nodes are required. The consistency index has the value of

$$\theta = \frac{A}{N} = \frac{\pi}{3}$$

**6.** The parametric position of the first new node is computed using Eq. (17.24):

$$\theta_1 = \frac{\pi}{3} = \int_0^{u_1} \frac{1}{2}\sqrt{16\sin^2(u) + 16\cos^2(u)}\mathrm{d}u = \int_0^{u_1} 2\mathrm{d}u = 2\,u_1$$

which gives $u_1 = \pi/6$. Here Eq. (17.24) can be solved exactly; no iterative scheme needs to be invoked. Using the above result, the parametric position $u_2$ of the second node is calculated as

$$\theta_2 = \frac{\pi}{3} = \int_{\pi/6}^{u_2} 2\mathrm{d}u = 2\left(u_2 - \frac{\pi}{6}\right)$$

We have $u_2 = \pi/3$.

**7.** Finally, the coordinates of the new nodes are obtained from the parametric equations of the circle. Substituting $u_1$ and $u_2$ into Eq. (17.27), we have for node $n_1$,

$$x_1 = 4\,\cos\left(\frac{\pi}{6}\right) = 2\sqrt{3}, \quad y_1 = 4\,\sin\left(\frac{\pi}{6}\right) = 2$$

and for node $n_2$,

$$x_2 = 4\cos\left(\frac{\pi}{3}\right) = 2, \quad y_2 = 4\sin\left(\frac{\pi}{3}\right) = 2\sqrt{3}$$

The curved edge is discretized by three sides after linking each of the nodes as shown in Fig. 17.11b.

#### 17.3.3.2 Generation front

A generation front is established prior to starting the triangular element generation. The initial generation front is a collection of all of the sides which form the discretized boundary edges of the domain. Thus, it consists of a set of closed loops of boundary sides. If the domain is composed of multiple connected regions, such as regions with different material properties, an initial generation front will be formed for each of the regions.

Each side in the generation front is defined by its two end points. The sequence of the sides is also arranged such that the regions to be meshed are always situated to the left of the generation front. The initial generation front for a simple rectangular domain is shown in Fig. 17.12a. At any stage of the element generation process, the generation front always forms the boundary of the region to be discretized as depicted in Fig. 17.12b. In the process of element generation, a side from the generation front is chosen as a base to form a new element with either a newly generated node or an existing node from the generation front. Once a new element is formed, the generation front is updated. Any side that has been used to create a new element is removed from the generation front and the newly created sides are added, as illustrated in Figs. 17.12c and 17.12d. The updating procedure ensures that the generation front always forms the boundary of the region to be meshed. The sides and the nodes in the generation front are referred to as *active sides* and *active nodes*, respectively. Processing of the generation front continues until the entire region is filled with elements and nodes.

#### 17.3.3.3 Element generation

The process of generating a triangular element is illustrated in Fig. 17.13 and includes the following steps:

1. An active side $ab$ connecting nodes $a$ and $b$ is selected from the generation front as a base to form the new element. To produce a mesh with smooth transition of the element size, the smallest side is considered first. The sides in the generation front are sorted and updated according to their length during the element generation process to increase the efficiency of the mesh generation algorithm.
2. At the middle point $m$ of side $ab$ compute the local mesh parameters $\boldsymbol{\alpha}_{m_i}$ and $h_{m_i}$ $(i = 1, 2)$ for the new element by interpolating from the background mesh.
3. The element creation process can be significantly simplified when point $m$ and all the nodes in the generation front with respect to coordinates $(x, y)$ are mapped to the normalized coordinate system $(x', y')$ by $\mathbf{x}' = \mathbf{T}\mathbf{x}$. The element generation process will be conducted in the coordinate system $(x', y')$ to construct a triangle that is as regular as possible.

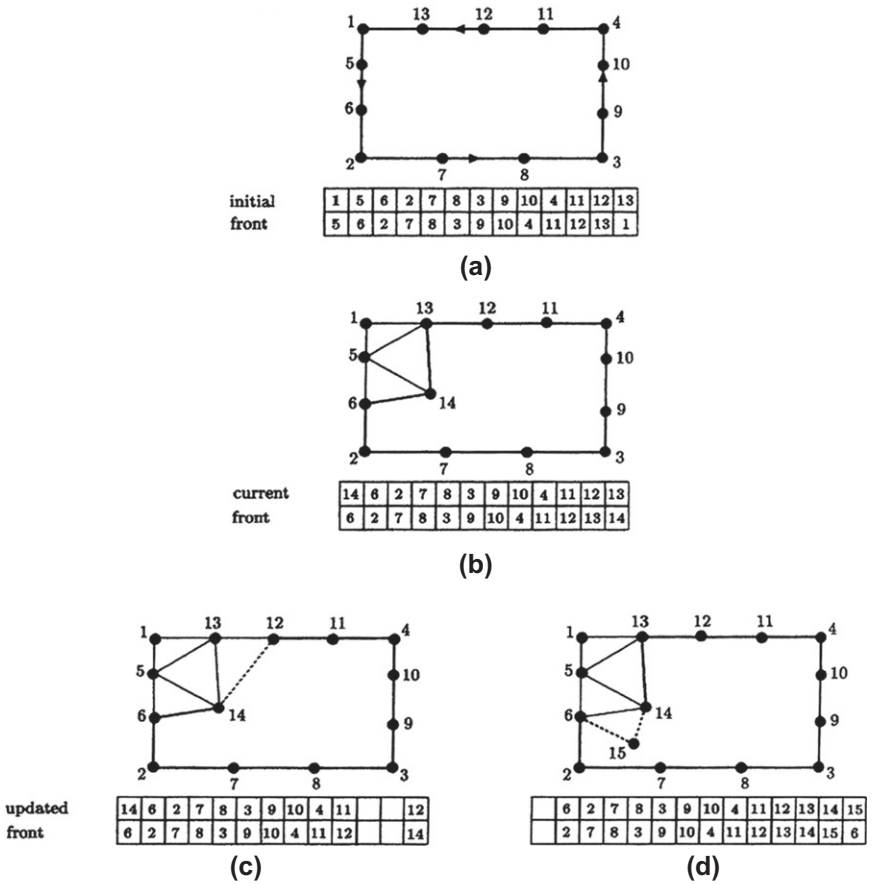



**FIGURE 17.12**

Generation front and its updating during the element generation process. (a) The initial generation front. (b) Generation front at a certain stage. (c) and (d) Updated generation front after the creation of a new element.

**4.** Determine the ideal position of new node $c'$ to form the new triangle. Node $c'$ is constructed in the direction normal to side $a'b'$ and located at a distance $h'_l$ from node $a'$ and node $b'$ as shown in Fig. 17.13. Here the normal of side $a'b'$ is pointing to the region to be meshed and $h'_l$ is chosen as

$$h'_l = \begin{cases} 0.55 l_{a'b'} & 1 \le 0.55 l_{a'b'} \\ 1 & 0.55 l_{a'b'} < 1 < 2 l_{a'b'} \\ 2 l_{a'b'} & 2 l_{a'b'} \le 1 \end{cases} \tag{17.28}$$

to ensure that an element with excessive distortion is not created. The constants appearing in the expression are empirical but have been shown to work well in practice.

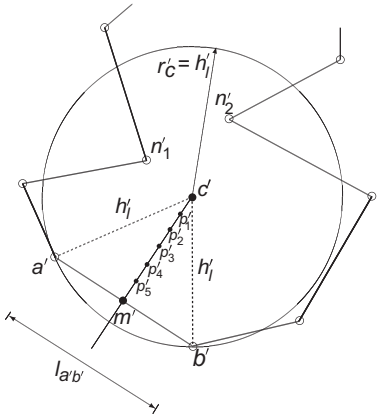



**FIGURE 17.13**
The ideal position of the new node and locations of the potential forming nodes.

**5.** Additional points are located to create a list of the potential nodes to generate the new element. These include

- All the nodes from the generation front that fall into the circle centered at $c'$ with radius

$$r_{c'} = N\, h'_l \tag{17.29}$$

  These nodes are ordered by their distance from $c'$ and denoted by $n'_1, n'_2, \ldots, n'_M$ with $n'_1$ being the closest to $c'$. The value of constant $N$ is given here as $N = 1$. The inclusion of such nodes creates the opportunity for the new element to be formed from existing nodes on the generation front.
- A collection of $L$ points are generated along the straight line between points $c'$ and $m'$. These points, denoted by $p'_1, p'_2, \ldots, p'_L$ are also ordered according to how close they are to node $c'$. Their addition ensures that a new element can always be generated. Here $L = 5$ is commonly used.

It is noted that the empirical values given to $N$ and $L$ can be automatically modified during the mesh generation process to include a sufficient number of nodes.

**6.** Nodes $n'_j (j = 1, 2, \ldots, M)$ from the potential node list and node $c'$ are considered sequentially to form the new triangle with side $a'b'$. Such ordering allows the existing nodes to be considered first when they are not too far away from $c'$. The point that forms the new triangle with side $a'b'$ is taken to be the first node that satisfies the criterion (17.28) such that the newly formed sides of the triangle do not intersect any of the existing sides in the generation front.
If neither nodes $n'_j (j = 1, 2, \ldots, M)$ nor node $c'$ can form the new triangle, nodes $p'_j (j = 1, 2, \ldots, L)$ are tested. The first $p'_j$ that verifies the criterion is taken to be the new node.

7. A new element is formed. If the new node $c'$ is adopted to generate the new triangle, its coordinates are transformed back to the original coordinate system by

$$\mathbf{x}_c = \mathbf{T}^{-1}\mathbf{x}'_c \tag{17.30}$$

8. The generation front list is updated after each new element is added and any new node is added to the node list.
The element generation process continues until the number of the sides in the generation front reduces to zero. The domain is then discretized completely by triangular elements.

### 17.3.4 Mesh quality enhancement for triangles

Mesh quality enhancement is indispensable to all mesh generation algorithms, because the shape of the triangles generated directly are not always optimal, particularly for a strongly graded mesh with element size varying rapidly. To improve the shape of the elements, at the final stage of the mesh generation, various mesh quality enhancement techniques, such as mesh smoothing and mesh modification, are employed.

#### *17.3.4.1 Mesh smoothing*

In the process of mesh smoothing, the topological structure of the mesh is fixed, i.e., the nodal connections of the elements will not be altered, but the interior nodes are repositioned to produce triangles with somewhat improved shapes. The computationally most efficient smoothing algorithm is the well-known Laplacian smoothing [52] which repositions the internal node at the centroid of the polygon formed by its neighboring nodes. The new position of an internal node $i$ is computed as

$$x_i = \frac{1}{N}\sum_{j=1}^{N} x_j \quad \text{and} \quad y_i = \frac{1}{N}\sum_{j=1}^{N} y_j \tag{17.31}$$

where $N$ is the number of the nodes linked to node $i$.

The mesh smoothing process consists of several (usually three to five) iterations. The technique has proved to be effective and generally adjusts the mesh into one with better shaped elements as shown in Fig. 17.14. However, the algorithm may fail if some of the neighboring nodes of interior node $i$ are boundary nodes and the polygon formed by these nodes is concave. The following example demonstrates this possibility.

**Example 17.3.** Concave domain

A simple triangulated concave domain with boundary nodes $a, b, c, d, e, f$, and their coordinates is shown in Fig. 17.15. The new position of the only interior node $i$, calculated by Eq. (17.31), is at

$$\begin{aligned} x_{i'} &= \frac{1}{6}\left(-1+4+4+0+0-1\right) = 1 \\ y_{i'} &= \frac{1}{6}\left(-1-1+0+0+4+4\right) = 1 \end{aligned}$$

which clearly is outside the domain.

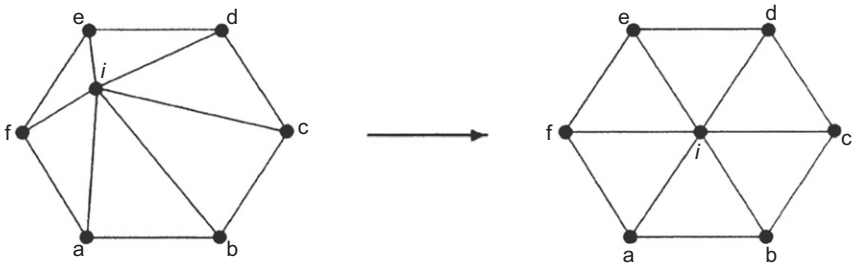



**FIGURE 17.14**
Laplacian smoothing. Node $i$ is repositioned.

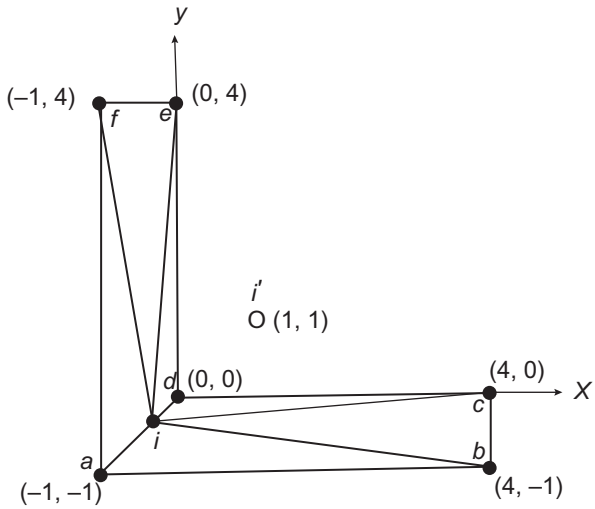



**FIGURE 17.15**
A pathological case of Laplacian smoothing.

To prevent such failure, various constraints can be added to Laplacian smoothing. One such constraint is to reposition node $i$ only if the maximum interior angle of all the elements linked by node $i$ is decreasing. This constraint is in fact only necessary for those interior nodes which have neighboring nodes being boundary nodes.

### *17.3.4.2 Mesh modification*

The topological structures of the mesh, such as the node-element relation, the side-element relation, and the node-node relation, are established once the process of element generation is completed. These relations, to some extent, reflect the regularity of the mesh. For instance, the optimal value of node element adjacency number NE,

which shows how many elements are connected to a node in a triangular mesh, is defined as

$$NE_{op} = \begin{cases} 6 & \text{for interior nodes} \\ max\left(\left\lfloor \frac{3\theta}{\pi} + \frac{1}{2} \right\rfloor, 1\right) & \text{for boundary nodes} \end{cases} \tag{17.32}$$

where $\theta$ is the internal angle formed by boundary edges joined at the boundary node and $\lfloor c \rfloor$ is the integer part of the value $c$. When $NE_{op}$ is attained for all the nodes, most of the elements in the resulting mesh are approximately equilateral. However, if an interior node has a node element adjacency number far bigger or smaller than $NE_{op}$, the surrounding elements of the node may be very distorted.

Distorted elements are, in general, inevitable for a mesh with varying element sizes as less regular transition elements are created during the element generation process. Distorted elements are also produced because the element size $h$ is considered only locally when each element is formed. The distortion of the element caused by nonoptimal topological structure of the mesh cannot be corrected by smoothing alone. In the following, techniques that alter the topology of the mesh in order to reduce the element distortion to a minimum are described.

*Node elimination.* A node elimination process consists of a loop over all the interior nodes. A node $i$ will be eliminated if:

1. $i$ is linked with three elements, i.e., $NE_i = 3$. $i$ is removed together with the three elements connected to it and replaced by a single element $e_i'$ as illustrated in Fig. 17.16.
2. $i$ is shared by four elements, i.e., $NE_i = 4$. $i$ is deleted from the mesh and its four related elements are reduced to two.

The possibility of such operations are depicted in Fig. 17.17.

It is noted that the element size distribution is almost unchanged in the process of node elimination.

*Diagonal swapping.* The process of diagonal swapping examines all the element sides common to two elements. Element sides that are part of a material interface

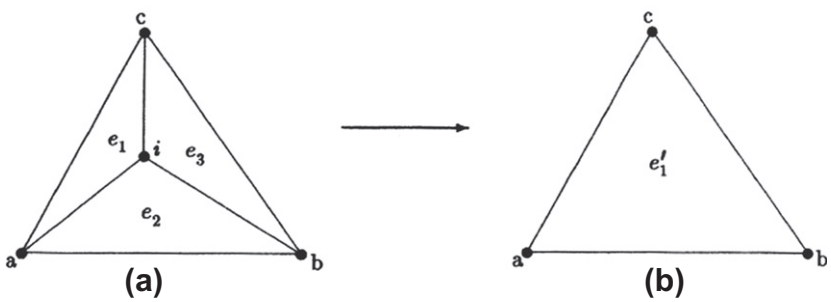



**FIGURE 17.16**
Elimination of node $i$ with $NE_i = 3$.

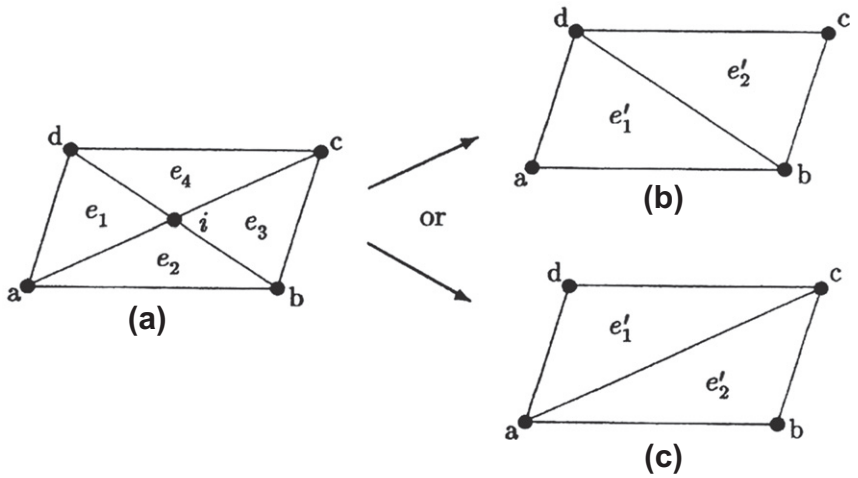



**FIGURE 17.17**
Elimination of node $i$ with $NE_e = 4$.


**FIGURE 17.18**
Diagonal swapping. Diagonal *ac* replaced by diagonal *bd*. Elements $e_1$ and $e_2$ changed to elements $e'_1$ and $e'_2$.

should not be altered. Considering a side shared by two triangles $e_1$ and $e_2$ shown in Fig. 17.18, the edge *ac* will be replaced by edge *bd* together with elements $e_1$ and $e_2$ being substituted for elements $e'_1$ and $e'_2$ when one of the following frequently used criteria is satisfied.

- The maximum internal angle of the new elements $e'_1$ and $e'_2$ should be smaller than that of elements $e_1$ and $e_2$.
  or
- The node element adjacency number improves to be closer to the optimal value after swapping.

Both criteria work well in practice. Figure 17.19 shows two diagonal swapping steps that satisfy the criterion of reducing the maximum angle in a region with seven elements. The quality of the elements is obviously improved.

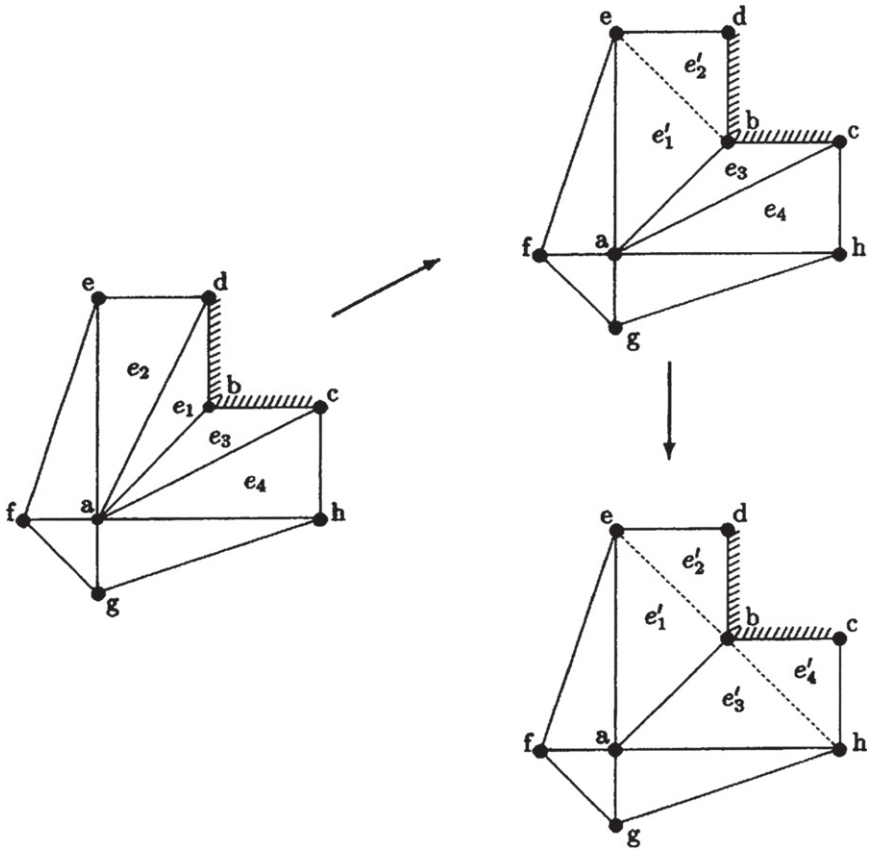



**FIGURE 17.19**
Diagonal swapping for elements in a boundary region.

Swapping of diagonals is not allowed if it results in a negative area for one of the newly created elements.

The mesh quality, such as the smoothness of the mesh and the regularity of the elements, can be significantly improved after combined applications of mesh quality enhancement techniques.

**Example 17.4.** Mesh for dam and vortex shedding

Two triangular meshes generated by a mesh generator using the advancing front method are shown in Figs. 17.20 and 17.21. The mesh plotted in Fig. 17.20 is used for finite element analysis of a dam. The mesh parameters are arbitrarily given. The mesh of Fig. 17.21 was generated in the adaptive analysis of fluid dynamics. The mesh reflects the distribution of the specified mesh parameters which are computed from an *a posteriori* error estimator [53].

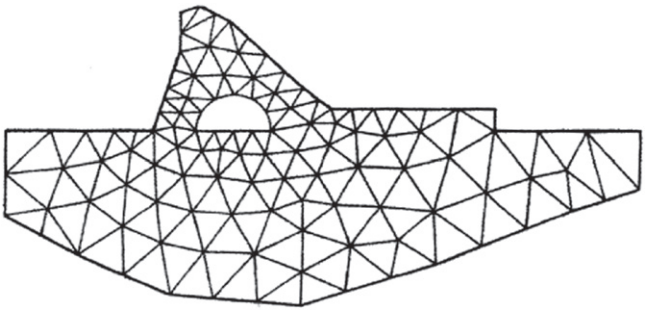


**FIGURE 17.20**
Triangular mesh for a dam.

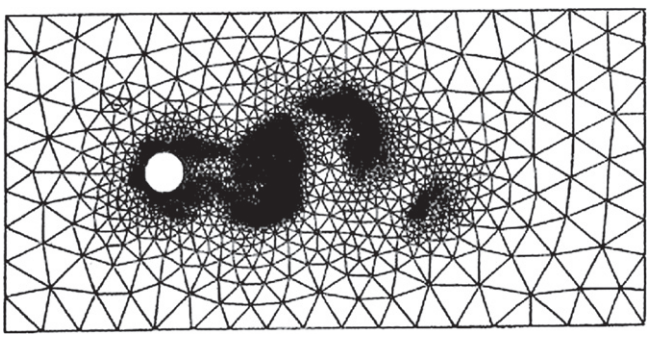


**FIGURE 17.21**
Triangular mesh with mesh size distribution given by an error estimator.

### 17.3.5 Higher order elements

Higher order elements can be created easily by adding additional intermediate nodes to each element edge. For an interior edge, the position of an intermediate node is determined directly by interpolation using the positions of the nodes at each end of the edge. For a boundary edge, however, the parametric position of the intermediate node between nodes $n_k$ and $n_{k+1}$ may be computed either by Eq. (17.24) or by interpolating the parametric position of $n_k$ and $n_{k+1}$ directly, so that the position of the intermediate node can be mapped onto the curvilinear boundary. These nodes are generated at the boundary node generation stage and placed on the boundary curve after the completion of the element generation process. The position of any interior nodes can be interpolated by the position of the element perimeter nodes. Figure 17.22 shows the locations of vertex nodes, edge nodes, and interior nodes for some quadratic elements.

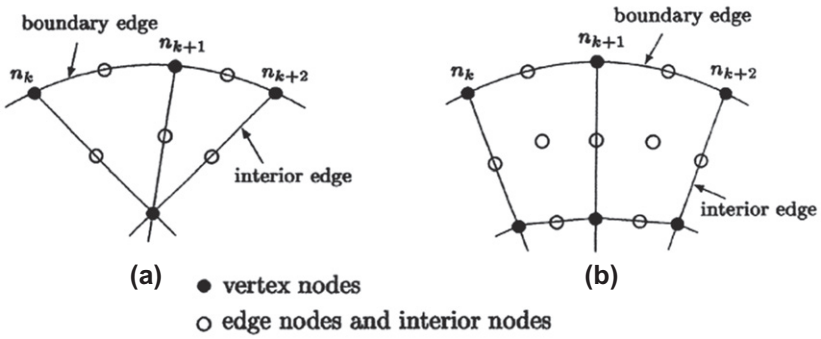



**FIGURE 17.22**
Quadratic elements.

### 17.3.6 Remarks

Before we proceed to discussion of surface mesh generation, several remarks are given on issues related to topics discussed in this section.

**Remark 1.** In order to implement the search algorithms required in the element generation process efficiently, use of special data structures such as those proposed in Refs. [54–56] for the advancing front method are advantageous.

**Remark 2.** The algorithm for the advancing front method has been shown to be robust in two-dimensional mesh generation of triangles and, although not discussed here, can easily be extended to generate quadrilaterals [57]. However, in the process of generating triangles, several empirical constants have been adopted, e.g., those used in Eqs. (17.28) and (17.29). The optimal values of these constants are still unknown. In addition, the correlation between the value of these constants and the structure of the resulting mesh is also an open question. Because of the lack of mathematical rigor, a robust element generation process for the advancing front method in three dimensions is much more complicated than that described here for two dimensions [23–28].

**Remark 3.** Other methodologies for generating quadrilateral mesh can be found in Refs. [30,58–64]. The various algorithms that convert an existing triangular mesh to a quadrilateral mesh [60–62,64] are all robust, although the element size and element orientation of the resulting mesh are influenced by the preexisting triangular mesh. Among these algorithms, the one proposed by Owen et al. [64] uses the advancing front method in the process of converting triangles to quadrilaterals.

## 17.4 Surface mesh generation

Surface mesh generation is a prerequisite for many three-dimensional mesh generation algorithms, such as those based on the advancing front method and the Delaunay triangulation method, but it is not required for the finite octree method. However,

generating a surface mesh prior to three-dimensional mesh generation has its advantages:

**(a)** The quality of three-dimensional meshes is strongly dependent on the quality of the surface mesh. A distorted triangle in the surface mesh will almost certainly result in a tetrahedron with poor quality.
**(b)** For many applications, an accurate description of the surface of a three-dimensional domain is essential. This can be readily realized by increasing the accuracy of the parametric representation of the surface and by assigning proper surface mesh element size distribution during the generation of the surface mesh.
**(c)** The B-rep of the three-dimensional geometry exported by CAD systems often contains defects, e.g., gaps between connecting surfaces and discontinuities of boundary edges of the surface. These defects can be corrected before and during the surface mesh generation process. Consequently, accurate surface mesh generation prevents three-dimensional mesh generation algorithms from failing due to presence of defects.

Although essential to many three-dimensional mesh generation algorithms, surface mesh generation also has its own application in finite element analysis; this is especially true in the solution of shell problems. In the process of generating a surface mesh for a three-dimensional geometry, each face of the geometry is discretized individually; the complete surface mesh of the geometry will be formed by a final assembly of the faces. Since we are mainly concerned with the algorithmic procedure of the surface mesh generation, we shall discuss the mesh generation algorithm for an individual face. The basic idea of the algorithm described below, proposed by Peraire et al. [65] and Peiró [66], is to perform mesh generation, according to the prescribed element size distribution, in the two-dimensional parametric plane and map the two-dimensional mesh onto the three-dimensional surface. From a computational standpoint, generating triangles (or quadrilaterals) on a plane is much simpler than that on a three-dimensional surface. The two-dimensional element generation algorithm described in the previous section can be readily applied. Nevertheless, in order to obtain a surface mesh that respects the prescribed geometrical characteristics, such as element size and element shape, the mesh parameters given to the three-dimensional surface mesh need to be transformed to the parametric plane. The mesh generation procedure includes four major steps:

1. Perform node generation along the curved boundary edge to form the discretized boundary of the surface.
2. Transform boundary nodes, therefore the discretized surface boundary, to the parametric plane.
3. Perform element generation in the parametric space within the discretized boundary.
4. Map the mesh in parametric space onto the surface using its parametric representation.

In the following we will mainly be concerned with the processes of boundary node generation and element generation in the parametric space, starting with the geometrical representation of the surface mesh.

## 17.4.1 Geometrical characteristics of the surface mesh

A surface mesh generally is three dimensional; thus, the spatial distribution of the shape and size of the surface elements needs to be specified in three dimensions. Because the mesh generation is performed in the two-dimensional parametric plane, proper spatial distribution of the element shape and size in the parametric space also needs to be defined.

### 17.4.1.1 Mesh control function in three dimensions

As was adopted in two-dimensional mesh generation, the geometrical characteristics of the surface mesh such as the distribution of the element shapes and sizes on the surface are controlled by mesh parameters. For a surface mesh, mesh parameters include a set of three mutually orthogonal directions $\boldsymbol{\alpha}_i$ $(i = 1, 2, 3)$, and their associated element sizes $h_i$ $(i = 1, 2, 3)$ as shown in Fig. 17.23.

Mesh parameters are defined at the nodes of a three-dimensional background mesh, which usually consists of a small number of tetrahedral elements. The background mesh can be constructed to cover the entire surface of the three-dimensional geometry or to cover each face of the geometry individually. In either case, the background mesh will be created automatically by dividing one or several hexahedra (tetrahedra or prisms) into tetrahedra. Figure 17.24 shows a tetrahedral background mesh generated from a single hexahedron for a single surface. The spatial distribution of the mesh parameters is furnished by the background mesh. At a particular point on the surface, the mesh parameters are computed by a linear interpolation of the values

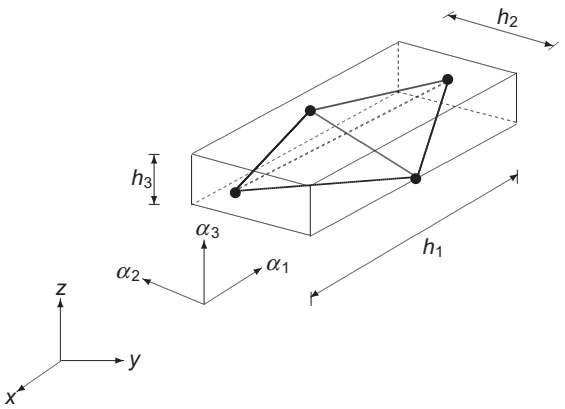



**FIGURE 17.23**
Mesh parameters defined in three dimensions.

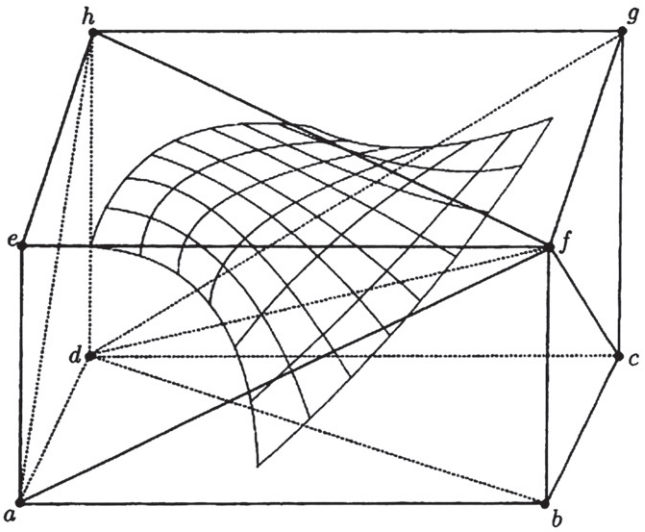



**FIGURE 17.24**
A six tetrahedra background mesh derived from a hexahedron. The six tetrahedra are (*e*,*f*,*h*,*a*), (*a*,*b*,*f*,*d*), (*f*,*h*,*d*,*a*), (*b*,*f*,*d*,*c*), (*d*,*g*,*h*,*f*), and (*d*,*c*,*g*,*f*).

assigned at the nodes of the background mesh. The geometrical characteristics at a point of the surface mesh at the point is therefore attained. For instance, when all three element sizes are found to be equal at the point, the tetrahedral elements in the surrounding area of the point will be approximately equilateral. Since the faces of the tetrahedral elements form the surface elements they are therefore approximately equilateral triangles.

A three-dimensional geometrical transformation matrix $\mathbf{T}$, similar to that defined in two-dimensional mesh generation, is in the form of

$$\mathbf{T}(\mathbf{x}) = \sum_{i=1}^{3} \frac{1}{h_i} \boldsymbol{\alpha}_i \boldsymbol{\alpha}_i^{\mathrm{T}} \tag{17.33}$$

and constitutes a scaling factor $1/h_i$ in each of the $\boldsymbol{\alpha}_i$ directions ($i = 1, 2, 3$). When it is applied to tetrahedra with element sizes $h_i$ in directions $\boldsymbol{\alpha}_i$ at a given point in the coordinate system $(x, y, z)$, the tetrahedra will be mapped to equilateral tetrahedra in the normalized space $(x', y', z')$. Consequently, the surface element will be mapped to equilateral triangles in the normalized space. Indeed, transformation $\mathbf{T}$ also provides a mapping relationship between parametric coordinates and the three-dimensional normalized coordinates

$$\mathbf{x}'(v, w) = \mathbf{T}\,\mathbf{x}(v, w) \tag{17.34}$$

As indicated, surface mesh generation will be performed in the parametric space $(v, w)$ and mapped onto the surface in three dimensions. In order to accomplish the

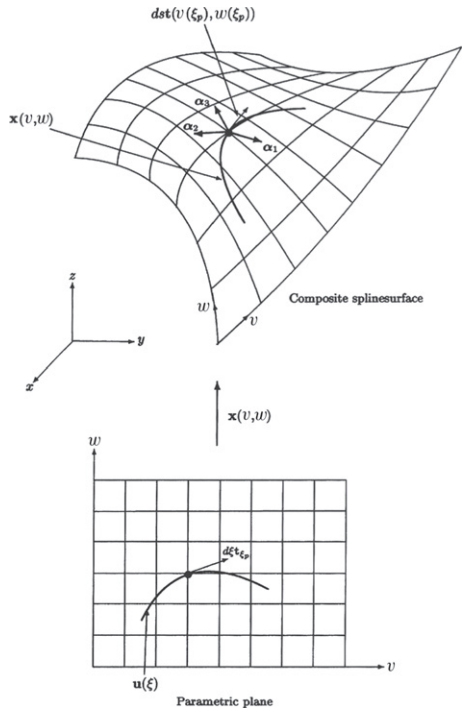



**FIGURE 17.25**
Arc length element $d\xi$ in parametric plane and its image on the surface.

planar mesh generation on the parametric plane, appropriate planar mesh parameters have to be assigned. These mesh parameters must be given in such a way that, after being mapped onto the surface, the final surface mesh respects the specified geometrical characteristics. Such a requirement can be achieved by deriving the planar mesh parameters from the three-dimensional surface mesh parameters.

### 17.4.1.2 Mesh parameters in parametric plane

We start by examining a curve in the parametric plane expressed by parameter $\xi$ and illustrated in Fig. 17.25, i.e.,

$$v = v(\xi), \quad w = w(\xi) \tag{17.35}$$

or in vector form

$$\mathbf{u}(\xi) = [v(\xi), w(\xi)]^{\mathrm{T}} \tag{17.36}$$

At a particular point $\mathbf{u}(\xi_p)$, the square of arc length element $\zeta$ along the curve is expressed as

$$(\mathrm{d}\zeta)^2 = (\mathrm{d}v)^2 + (\mathrm{d}w)^2 = \left(\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\xi}\mathrm{d}\xi\right)^{\mathrm{T}}\left(\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\xi}\mathrm{d}\xi\right) = \mathbf{t}_\xi^{\mathrm{T}}\mathbf{t}_\xi (l_\xi d\xi)^2 = (l_\xi \mathrm{d}\xi)^2 \tag{17.37}$$

where $l_\xi$ is the length of the tangent vector

$$l_\xi = \sqrt{\left(\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\xi}\right)^{\mathrm{T}}\left(\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\xi}\right)} \tag{17.38}$$

and $\mathbf{t}_\xi$ is the unit tangent vector

$$\mathbf{t}_\xi = \frac{1}{l_\xi}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\xi} \tag{17.39}$$

This shows that the arc length element in the direction of a unit tangent along the curve in the parametric plane can be expressed by

$$\mathrm{d}\zeta = l_\xi \mathrm{d}\xi \tag{17.40}$$

We now consider the image of the planar curve at point $\mathbf{x}(v(\xi_p), w(\xi_p))$ on the surface represented by

$$\mathbf{x}(v, w) = [x(v, w) \quad y(v, w) \quad z(v, w)]^{\mathrm{T}} \tag{17.41}$$

The square of the arc length element in the direction of the unit tangent $\mathbf{t}$ along the curve on the surface is given by

$$(\mathrm{d}s)^2 = (\mathrm{d}x)^2 + (\mathrm{d}y)^2 + (\mathrm{d}z)^2 = (\mathrm{d}\mathbf{x})^{\mathrm{T}}(\mathrm{d}\mathbf{x}) \tag{17.42}$$

where

$$\mathrm{d}\mathbf{x} = \begin{Bmatrix} \mathrm{d}x \\ \mathrm{d}y \\ \mathrm{d}z \end{Bmatrix} = \begin{bmatrix} \dfrac{\partial x}{\partial v} & \dfrac{\partial x}{\partial w} \\ \dfrac{\partial y}{\partial v} & \dfrac{\partial y}{\partial w} \\ \dfrac{\partial z}{\partial v} & \dfrac{\partial z}{\partial w} \end{bmatrix} \begin{Bmatrix} \mathrm{d}v \\ \mathrm{d}w \end{Bmatrix} = [\,\mathbf{x}_{,v} \quad \mathbf{x}_{,w}\,]\,\mathrm{d}\mathbf{u} \tag{17.43}$$

Assume that the transformation matrix $\mathbf{T}_{\mathrm{d}s}$, correlated to $\mathrm{d}s$ at $\mathbf{x}(v(\xi_p), w(\xi_p))$, maps $\mathrm{d}\mathbf{x}$ to the normalized space with a unit length, i.e.,

$$(\mathrm{d}s')^2 = (\mathrm{d}x')^2 + (\mathrm{d}y')^2 + (\mathrm{d}z')^2 = (\mathrm{d}\mathbf{x})^{\mathrm{T}}\mathbf{C}_{\mathrm{d}s}(\mathrm{d}\mathbf{x}) = 1 \tag{17.44}$$

where

$$\mathbf{C}_{\mathrm{d}s} = \mathbf{T}_{\mathrm{d}s}^{\mathrm{T}}\mathbf{T}_{\mathrm{d}s} = \sum_{i=1}^{3} \frac{1}{h_{\mathrm{d}s_i}^2}\boldsymbol{\alpha}_i\boldsymbol{\alpha}_i^{\mathrm{T}} \tag{17.45}$$

corresponding to the *Euclidean metric tensor* in the normalized space, and $h_{\mathrm{d}s_i}$ can be viewed as element size associated with $\mathrm{d}s$ in the direction of $\boldsymbol{\alpha}_i$.

Substitute d$\mathbf{x}$ into Eq. (17.44), we have

$$\left(\left[\mathbf{x}_{,v} \quad \mathbf{x}_{,w}\right] \mathrm{d}\mathbf{u}\right)^{\mathrm{T}} \mathbf{C}_{\mathrm{d}s} \left(\left[\mathbf{x}_{,v} \quad \mathbf{x}_{,w}\right] \mathrm{d}\mathbf{u}\right) = 1 \tag{17.46}$$

From Eqs. (17.39) and (17.40), we know that

$$\mathrm{d}\mathbf{u} = l_{\xi}\, \mathrm{d}\xi\, \mathbf{t}_{\xi} = \mathrm{d}\zeta\, \mathbf{t}_{\xi} \tag{17.47}$$

which when substituted into Eq. (17.46) gives

$$\left(\left[\mathbf{x}_{,v} \quad \mathbf{x}_{,w}\right] \mathbf{t}_{\xi}\right)^{\mathrm{T}} \mathbf{C}_{\mathrm{d}s} \left(\left[\mathbf{x}_{,v} \quad \mathbf{x}_{,w}\right] \mathbf{t}_{\xi}\right) (\mathrm{d}\zeta)^2 = 1 \tag{17.48}$$

This shows that for an arc length element d$s$ along a curve on the surface, the arc length along its curve in the parametric plane is

$$\mathrm{d}\zeta = \frac{1}{\sqrt{\left(\left[\mathbf{x}_{,v} \quad \mathbf{x}_{,w}\right] \mathbf{t}_{\xi}\right)^{\mathrm{T}} \mathbf{C}_{\mathrm{d}s} \left(\left[\mathbf{x}_{,v} \quad \mathbf{x}_{,w}\right] \mathbf{t}_{\xi}\right)}} \tag{17.49}$$

Replacing d$s$ by the curvilinear element size $h_s$ and $\mathbf{T}_{\mathrm{d}s}$ by $\mathbf{T}$ on the surface, and using $h_{\xi}$, the element size along the planar curve on the parametric plane, in place of d$\zeta$, we have

$$\left(\left[\mathbf{x}_{,v} \quad \mathbf{x}_{,w}\right] \mathbf{t}_{\xi}\right)^{\mathrm{T}} \mathbf{C} \left(\left[\mathbf{x}_{,v} \quad \mathbf{x}_{,w}\right] \mathbf{t}_{\xi}\right) (h_{\xi})^2 = 1 \tag{17.50}$$

or

$$h_{\xi} = \frac{1}{\sqrt{\left(\left[\mathbf{x}_{,v} \quad \mathbf{x}_{,w}\right] \mathbf{t}_{\xi}\right)^{\mathrm{T}} \mathbf{C} \left(\left[\mathbf{x}_{,v} \quad \mathbf{x}_{,w}\right] \mathbf{t}_{\xi}\right)}} = \frac{1}{\sqrt{\mathbf{t}_{\xi}^{\mathrm{T}} \mathbf{G}\, \mathbf{t}_{\xi}}} \tag{17.51}$$

The matrix of the metric tensor is now expressed as

$$\mathbf{C} = \mathbf{T}^{\mathrm{T}}\mathbf{T} = \sum_{i=1}^{3} \frac{1}{h_i^2} \boldsymbol{\alpha}_i \boldsymbol{\alpha}_i^{\mathrm{T}} \tag{17.52}$$

and

$$\mathbf{G} = [\mathbf{x}_{,v} \ \mathbf{x}_{,w}]^{\mathrm{T}} \mathbf{C} [\mathbf{x}_{,v} \ \mathbf{x}_{,w}] \tag{17.53}$$

is the matrix of the metric tensor in the parametric space.

If we assume that $\mathbf{T}$ is a constant matrix in the neighborhood of point $\mathbf{x}(v(\xi_p), w(\xi_p))$, substitute Eq. (17.52) into Eq. (17.51), and note the mapping relationship of Eq. (17.34), we obtain $h_\xi$ in a somewhat different form

$$h_\xi = \frac{1}{\sqrt{\mathbf{t}_\xi^{\mathrm{T}} \mathbf{g} \mathbf{t}_\xi}} \tag{17.54}$$

where

$$\mathbf{g} = \begin{bmatrix} g_{vv} & g_{vw} \\ g_{wv} & g_{ww} \end{bmatrix} \tag{17.55}$$

is the *first fundamental matrix* of the surface in the normalized space [67], and

$$\begin{aligned} g_{vv} &= \left(\frac{\partial x'}{\partial v}\right)^2 + \left(\frac{\partial y'}{\partial v}\right)^2 + \left(\frac{\partial z'}{\partial v}\right)^2 \\ g_{vw} &= 2\left(\frac{\partial x'}{\partial v}\frac{\partial x'}{\partial w} + \frac{\partial y'}{\partial v}\frac{\partial y'}{\partial w} + \frac{\partial z'}{\partial v}\frac{\partial z'}{\partial w}\right) = g_{wv} \\ g_{ww} &= \left(\frac{\partial x'}{\partial w}\right)^2 + \left(\frac{\partial y'}{\partial w}\right)^2 + \left(\frac{\partial z'}{\partial w}\right)^2 \end{aligned} \tag{17.56}$$

Consequently, we have established the relationship between $\mathbf{G}$ and $\mathbf{g}$. When transformation $\mathbf{T}$ is a constant matrix,

$$\mathbf{G} = \mathbf{g} \tag{17.57}$$

This shows that the matrix of the metric tensor in parametric space is the same as the first fundamental matrix of the surface in the normalized space.

The two-dimensional mesh parameters $\boldsymbol{\alpha}_i(v(\xi_p), w(\xi_p))$ and $h_i(v(\xi_p), w(\xi_p))$ for the planar mesh on the parametric plane are computed from the directions in which $h_\xi$ attains an extremum. To this end, Eq. (17.51) is rewritten in the form

$$\mathbf{t}_\xi^{\mathrm{T}} \mathbf{G}\, \mathbf{t}_\xi = \frac{1}{h_\xi^2} \tag{17.58}$$

Let

$$\mathbf{G} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \tag{17.59}$$

where

$$G_{11} = \mathbf{x}_{,v}^{\mathrm{T}} \mathbf{C}\, \mathbf{x}_{,v}, \quad G_{12} = \mathbf{x}_{,v}^{\mathrm{T}} \mathbf{C}\, \mathbf{x}_{,w}, \quad G_{22} = \mathbf{x}_{,w}^{\mathrm{T}} \mathbf{C}\, \mathbf{x}_{,w} \tag{17.60}$$

and $G_{21} = G_{12}$.

From Eq. (17.58), we know that finding the direction in which $1/h_\xi^2$ reaches an extremum involves the solution of an eigen-problem for the symmetric matrix $\mathbf{G}$. To this end we let $\lambda_1, \lambda_2$ with $\lambda_1 \geq \lambda_2$ denote the eigenvalues and $\mathbf{a}_1, \mathbf{a}_2$ the eigenvectors of $\mathbf{G}$. The mesh parameters in the parametric plane at point $(v(\xi_p), w(\xi_p))$ are given as

$$h_1(v, w) = \frac{1}{\sqrt{\lambda_1}}, \quad h_2(v, w) = \frac{1}{\sqrt{\lambda_2}} \tag{17.61}$$

and

$$\boldsymbol{\alpha}_1(v, w) = \mathbf{a}_1, \quad \boldsymbol{\alpha}_2(v, w) = \mathbf{a}_2 \tag{17.62}$$

where the eigenvalues are computed as

$$\begin{aligned} \lambda_1 &= \frac{G_{11} + G_{22}}{2} + \sqrt{\frac{(G_{11} - G_{22})^2}{4} + G_{12}^2} \\ \lambda_2 &= \frac{G_{11} + G_{22}}{2} - \sqrt{\frac{(G_{11} - G_{22})^2}{4} + G_{12}^2} \end{aligned} \tag{17.63}$$

and the eigenvectors are

$$\mathbf{a}_1 = [\cos\theta \quad \sin\theta]^{\mathrm{T}}, \quad \mathbf{a}_2 = [-\sin\theta \quad \cos\theta]^{\mathrm{T}} \tag{17.64}$$

with

$$\theta = \frac{1}{2}\tan^{-1}\left(\frac{2G_{12}}{G_{11} - G_{22}}\right) \tag{17.65}$$

### 17.4.2 Discretization of three-dimensional curves

In order to perform mesh generation on the parametric plane, the boundary curves of the surface will be discretized in three-dimensional space and then projected to the parametric plane by inverse mapping.

#### 17.4.2.1 Node generation on the curves

The algorithmic procedure for node generation on three-dimensional curves is identical to that described in Section 17.3.3 for the two-dimensional curve. The curves are of course now represented by Eq. (17.1). The procedure listed below is the same as that discussed in Section 17.3.3 but now expressed in its three-dimensional form.

1. A set of sampling points $\mathbf{x}_l = \mathbf{x}(u_l)$ $(l = 0, 1, 2, \ldots, m)$ is first placed along the curve with parameters $u_l$ uniformly distributed as shown in Fig. 17.26. The unit tangent vector for the three-dimensional curve is computed at the sampling point as

$$\mathbf{t}_l = [t_{1_l} \quad t_{2_l} \quad t_{3_l}]^{\mathrm{T}} \tag{17.66}$$

where

$$t_{1_l} = \frac{x_{,u_l}}{l_{u_l}}, \quad t_{2_l} = \frac{y_{,u_l}}{l_{u_l}}, \quad t_{3_l} = \frac{z_{,u_l}}{l_{u_l}} \tag{17.67}$$

with $l_{u_l} = \sqrt{x_{,u_l}^2 + y_{,u_l}^2 + z_{,u_l}^2}$ and $\mathbf{x}_{u_l} = (x_{,u_l}\, y_{,u_l}\, z_{,u_l})$ the tangent vector.

2. Using the interpolated values from the background mesh, the mesh parameters $\boldsymbol{\alpha}_{l_i}, h_{l_i}$ $(i = 1, 2, 3)$, and the associated transformation matrix $\mathbf{T}_l$ are computed at each sampling point.

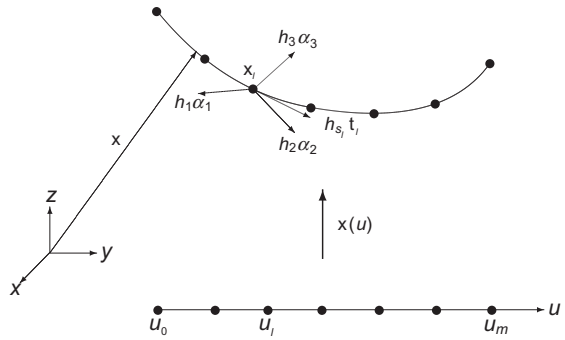



**FIGURE 17.26**
Sampling points and mesh parameters on a three-dimensional curve.

**3.** The element size $h_{s_l}$ at the sampling point $\mathbf{x}_l$ $(l = 0, 1, 2, \ldots, m)$ in the tangent direction is calculated by

$$h_{s_l} = \frac{1}{\sqrt{(\mathbf{T}_l\mathbf{t}_l)^{\mathrm{T}}(\mathbf{T}_l\mathbf{t}_l)}} = \frac{1}{\sqrt{\mathbf{t}_l^{\mathrm{T}}\mathbf{C}_l\mathbf{t}_l}} \tag{17.68}$$

where the metric tensor $\mathbf{C}$ is defined in Eq. (17.52).

**4.** A continuous element size distribution function is obtained by linear interpolation of the nodal values $h_{s_l}$ at sampling points using finite element shape functions

$$h(u) = \sum_{l=0}^{m} h_{s_l} N_l(u) \tag{17.69}$$

The element density function is set to $1/h(u)$.

**5.** The total number of sides $N$ to be generated along the curve is taken to be the nearest integer to

$$A = \int_0^L \frac{1}{h(u)}\mathrm{d}s = \int_{u_0}^{u_m} \frac{1}{h(u)}\sqrt{(x_{,u})^2 + (y_{,u})^2 + (z_{,u})^2}\mathrm{d}u \tag{17.70}$$

where $L$ is the length of the curve. The consistency index is calculated as

$$\theta = \frac{A}{N} \tag{17.71}$$

In addition to the already known end points $\mathbf{x}(u_0)$ and $\mathbf{x}(u_m)$ of the curve, there will be $(N - 1)$ new nodes created along the curve.

**6.** The parametric positions $u_{k+1}$ $(k = 0, 1, 2, \ldots, (N - 2))$ of the new nodes are computed consecutively from

$$\theta = \int_{u_k}^{u_{k+1}} \frac{1}{h(u)}\sqrt{(x_{,u})^2 + (y_{,u})^2 + (z_{,u})^2}\mathrm{d}u \tag{17.72}$$

by an iterative method, such as a Newton method.

7. Finally, the position of all the new points are mapped onto the boundary curve using Eq. (17.1).

#### 17.4.2.2 Place boundary nodes on parametric plane

In order to perform surface mesh generation in the parametric plane, the discretized boundary curves must be placed on the parametric plane to form the boundary of the region to be meshed. However, the inverse mapping

$$\mathbf{u}(\mathbf{x}) = [v(\mathbf{x}) \quad w(\mathbf{x})]^{\mathrm{T}} \tag{17.73}$$

is in general not expressed explicitly. When a NURBS surface of Eq. (17.6) is used to represent the surface of the geometry, the above inverse mapping is clearly nonlinear.

The parametric position of a boundary node on the surface can be found by assuming that its parametric coordinates are the same as its closest point on the surface, which is to find a point $\mathbf{x}(v, w)$ on the surface that is the closest point to boundary node $\mathbf{x}(u_k)$. The problem of finding the closest point can be formulated as follows: find the parametric coordinates $(v, w)$ of a surface point such that

$$D = \|\mathbf{x}(v, w) - \mathbf{x}(u_k))\| = \text{minimum} \tag{17.74}$$

where $\|\cdot\|$ denotes the Euclidean norm.

This problem is usually nonlinear and may be solved by various iterative methods [68,69]. The initial approximation of the parametric coordinates of a boundary node is taken as the computed position of a previous boundary point.

After all the boundary nodes are placed on the parametric plane, they are linked by straight lines to form the discretized boundary of the region to be meshed.

### 17.4.3 Element generation in the parametric plane

The element generation procedure in the parametric plane is the same as that described in Section 17.3.3 except that when creating a new element, the mesh parameters $\boldsymbol{\alpha}_i(v, w)$ and $h_i(v, w)$ computed from the specified mesh parameters for the surface mesh have to be utilized. After the completion of the element generation, quality enhancement techniques need to be applied which include a constraint of preserving the curvature of the surface to improve the quality of the mesh [66,70]. The final mesh is then mapped onto the surface by $\mathbf{x}(v, w)$ to obtain the required surface mesh.

The complete surface of a three-dimensional solid can be achieved after assembling the surface mesh of all the faces. Such a surface mesh may be used as the discretized boundary for three-dimensional mesh generation, which we shall discuss in the next section.

**Example 17.5.** Faces of machine part

A simple face from the machine part shown in Fig. 17.1 is discretized by a surface mesh generator using the algorithms discussed in this section. Figure 17.27a shows a coarse mesh. A finer mesh is illustrated in Fig. 17.27b. Complete surface meshes with different element size distributions are shown in Figs. 17.28a and 17.28b.

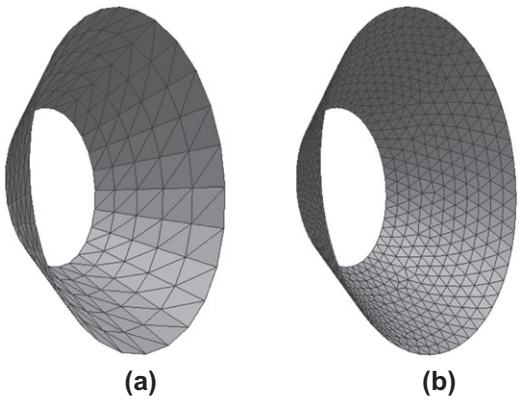



**FIGURE 17.27**
Surface meshes of a face from the machine part of Fig. 17.1: (a) course mesh and (b) fine mesh.

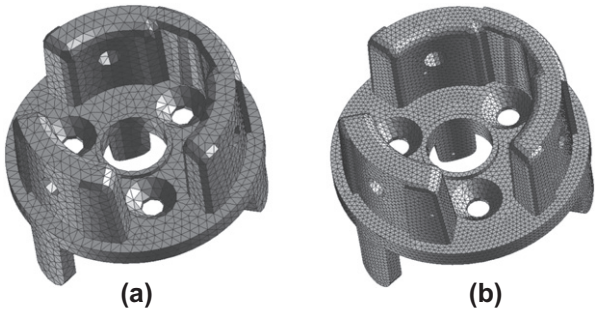



**FIGURE 17.28**
Complete surface meshes of the machine part of Fig. 17.1: (a) course mesh and (b) fine mesh.

**Example 17.6.** Boundary faces of gearbox

The boundary edges of the surface representation of a gearbox part are shown in Fig. 17.29a. The geometry of the part is realistic and hence somewhat complex. A complete surface mesh of the part generated automatically using the algorithms described in this section is illustrated in Fig. 17.29b.

### 17.4.4 Higher order surface elements

One of the advantages of generating a surface mesh in the parametric plane is that it can produce higher order elements without additional difficulties. When a mesh of higher order elements is generated in the parametric plane, it will be mapped onto the surface to form a boundary fit surface mesh with all the nodes on the surface. To preserve the boundary curves, it is important to generate the intermediate boundary

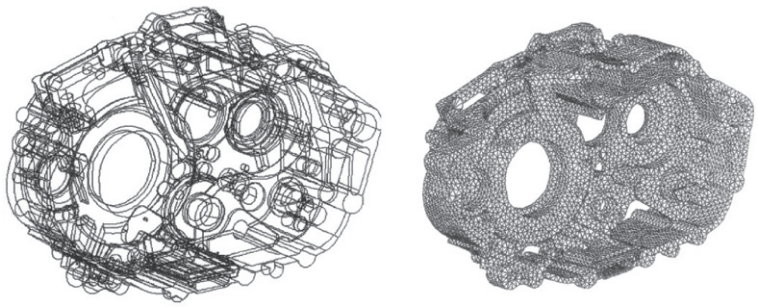



**FIGURE 17.29**
Mesh generation for a gearbox part: (a) boundary faces and edges and (b) complete surface mesh.

edge nodes by following the node generation procedure for curves. Figure 17.30 demonstrates a surface mesh of quadratic elements for a mechanical part.

### 17.4.5 Remarks

Several remarks on issues related to surface mesh generation are given below.

**Remark 4.** Once the discretized boundary of the region in the parametric plane is available, any two-dimensional element generation algorithm could be used to generate a valid mesh on the parametric plane. For convenience, we only mentioned the two-dimensional advancing front method described in Section 17.3.

**Remark 5.** Equations (17.51) and (17.54) have revealed that the mesh parameters in the parametric plane are a function of the parameterization of the surface. In order to satisfy the specified geometrical characteristics of the surface, the two-dimensional mesh generation algorithm is required to be capable of generating the mesh strictly following the computed mesh parameters for the parametric plane. Otherwise, serious distortion in the surface mesh may occur when the planar mesh is mapped onto the surface. We refer the reader to Refs. [71–77] for additional discussion on issues of surface mesh generation.

**Remark 6.** Besides the parametric representation, the surface of the three-dimensional geometry can also be represented in a discrete form. Surface mesh generation algorithms for surfaces represented in such form are widely used in practice. Several surface mesh generation algorithms based on discrete representation of surfaces can be found in Refs. [78–84]. We also refer the reader to Refs. [85–87] for surface mesh generation algorithms based on the data provided by magnetic resonance imaging (MRI) and computed tomography (CT).

**Remark 7.** The surface mesh generation algorithms discussed in this section assume that the geometrical representation and topological representation of the surfaces are

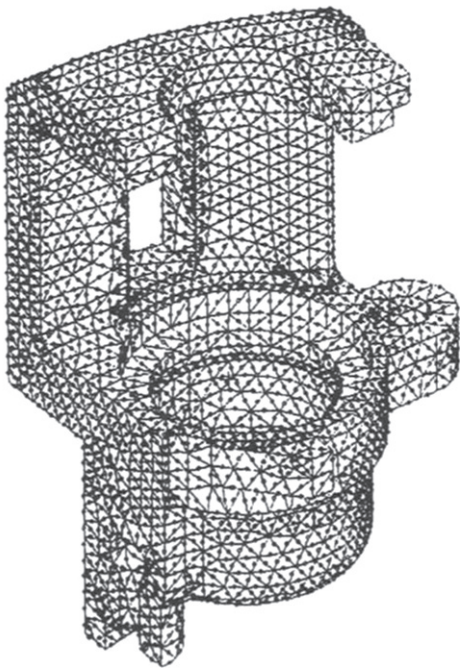


**FIGURE 17.30**
Surface mesh of quadratic elements.

correct, i.e., there are no defects in the surface representation of the three-dimensional geometry. In practical computations, this is often not the case. The boundary representation of the geometry provided by the CAD systems sometimes contains errors or undesirable features that will either cause the mesh generation algorithm to fail or the quality of the surface mesh to become unacceptable for three-dimensional mesh generation algorithms. Although methodologies that automatically remove defects and detrimental features from the boundary representations of the surface are not in the scope of surface mesh generation, they critically affect the success of automatic surface mesh generation and therefore deserve further research.

**Remark 8.** Finally, surface mesh generation is often used as the boundary discretization of a three-dimensional geometry. The quality of the surface mesh not only affects the quality of the three-dimensional mesh, it also affects the robustness of any three-dimensional mesh generation algorithm. This is particularly true for the three-dimensional advancing front mesh generation method. Although the robustness of the three-dimensional Delaunay triangulation method is less dependent on the quality of the surface mesh, the quality of the final three-dimensional mesh is certainly affected. To have a successful three-dimensional mesh generation algorithm, the quality of the surface mesh must be ensured before the interior mesh generation process starts.

## 17.5 Three-dimensional mesh generation: Delaunay triangulation

Many practical finite element computations are carried out on complex three-dimensional domains. The level of difficulty to automatically generate valid meshes for arbitrary three-dimensional domains is much greater than in two dimensions. In principle, the Delaunay triangulation, advancing front, and finite octree method are all applicable to three-dimensional mesh generation. However the Delaunay triangulation method has attracted most of the attention in theoretical research and software development, due to its conceptual simplicity, mathematical rigor, and algorithmic robustness. In this section, we shall be concerned with the Delaunay triangulation method and its application to three-dimensional mesh generation. We shall also introduce mesh quality enhancement methods which are crucial to ensure the final mesh can be used in the finite element computations.

### 17.5.1 Voronoi diagram and Delaunay triangulation

Delaunay triangulation [88] is the dual of the Voronoi diagram [89]. The properties of the Voronoi diagram and Delaunay triangulation provide the theoretical foundation for all the mesh generation methods based on the Delaunay method. In order to facilitate the description of the Delaunay triangulation method for mesh generation, a brief review of the basic properties of the Voronoi diagram and Delaunay triangulation is presented in a two-dimensional setting for visualization convenience, but these properties are equally valid in three dimensions.

Let $P = \{p_i, i = 1, 2, \ldots, N\}$ be a set of distinct points in the two-dimensional Euclidean plane $R^2$. They are referred to as the *forming points* in the mesh generation literature. The *Voronoi region* $V(p_i)$ is defined as the set of points $x \in R^2$ that are at least as close to $p_i$ as to any other forming point, i.e.,

$$V(p_i) = \{x \in R^2 : \|x - p_i\| \leq \|x - p_j\|, \ \forall j \neq i\} \tag{17.75}$$

Figure 17.31a depicts 10 Voronoi regions, with two interior regions bounded by eight others, the total being defined by an equal number of forming points. It follows that the Voronoi region $V(p_i)$ represents a convex polygonal region, possibly unbounded; and any point $x$ inside of $V(p_i)$ is nearer to $p_i$ than any other forming point in $P$. The points that belong to more than one region form the edges of the Voronoi regions and the edges of the Voronoi region $V(p_i)$ are the portion of the perpendicular bisectors separating the segment joining forming points $p_i$ and $p_j$ when $V(p_i)$ and $V(p_j)$ are contiguous. The union of the Voronoi regions is called the Voronoi diagram of the forming point set $P$.

The dual graph of the Voronoi diagram is produced by connecting the forming points of the neighboring Voronoi regions sharing a common edge with straight lines. It forms the Delaunay triangulation $D(P)$ of the Voronoi forming points $P$.

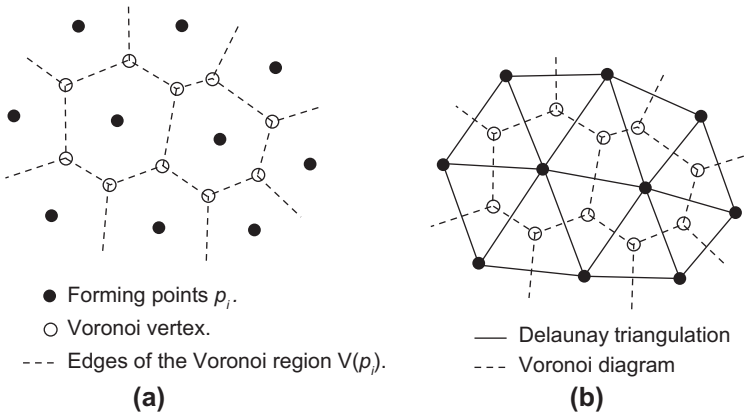



**FIGURE 17.31**

Voronoi and Delaunay diagrams for 10 forming points: (a) Voronoi diagram and (b) Delaunay triangulation.

Figure 17.31b illustrates the Delaunay triangulation and its corresponding Voronoi diagram.

In addition to those already mentioned, several properties of the Delaunay triangulation and Voronoi diagrams that are most relevant to the mesh generation algorithms of Delaunay triangulation are listed below [90,91]:

- **(i)** Delaunay triangulation is formed by triangles if no four points of the forming points $P$ are co-circular. These triangles are called *Delaunay triangles*.
- **(ii)** Each Delaunay triangle corresponds to a Voronoi vertex, which is the center of the circumcircle of the triangle, as depicted in Fig. 17.32.
- **(iii)** The interior of the circumcircle contains no forming points of $P$.
- **(iv)** The boundary of the Delaunay triangulation is the convex hull of the forming points.

In the Delaunay triangulation-based mesh generation algorithm, property (i) is used to avoid degeneracy; property (ii) is often used to construct data structures; property (iii) forms the well-known Delaunay criterion, the *empty circle criterion*, or the *in-circle criterion* when verifying whether it is violated by the new point introduced to the Delaunay triangulation; and property (iv) is the theoretical origin of using a convex hull, which contains all the mesh points, in mesh generation.

**Example 17.7.** Adding points into Delaunay triangulation

Each Voronoi diagram corresponds to a set of forming points which forms Delaunay triangulation. Adding a new forming point will inevitably result in a modification of the Voronoi diagram and the Delaunay triangulation. The process of constructing a new Voronoi diagram and Delaunay triangulation after the insertion of a new node

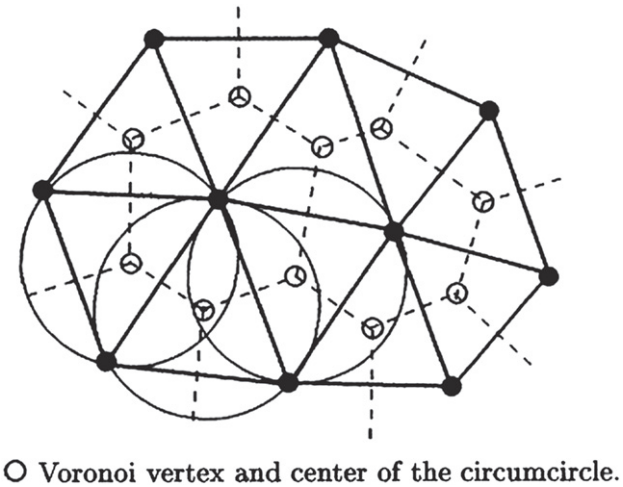



**FIGURE 17.32**
Circumcircles of the Delaunay triangles. Only three are shown.

is frequently used in automatic mesh generation and is illustrated here in the same two-dimensional setting shown in Fig. 17.31b.

Let the new forming point $n$ be inserted in the Delaunay triangulation shown in Fig. 17.33a. It falls into the circumcircles of Delaunay triangles $afg$, $abf$, and $bef$, therefore violating property (iii). This causes the removal of the three Voronoi vertices which are the centers of the circumcircles and their corresponding Delaunay triangles, as illustrated in Fig. 17.33b. The new Delaunay triangulation is constructed by linking the new forming point $n$ and its contiguous forming points that form a face of the neighboring triangle followed by the construction of the new Voronoi diagram as shown in Fig. 17.33c.

As we have indicated previously, the process used in the last example is applicable to three dimensions.

### 17.5.2 Mesh generation by Delaunay triangulation

Although by definition Delaunay triangulation decomposes the convex hull of the forming points into triangles in two dimensions and tetrahedra in three dimensions, it does not address the issues of how Delaunay triangulation can be formed effectively; how to generate those points that will be inserted in the triangulation; and how to preserve the boundary of a region when the forming points are from the boundary of a concave region. These issues are the three most important components of the automatic mesh generation algorithms of the Delaunay type. A large body of literature exists on research of these three subjects. The most representative ones are the work of Bowyer [11] and Watson [12] on the efficient Delaunay triangulation algorithms, which were introduced to mesh generation by Cavendish et al. [13], Weatherill [14],

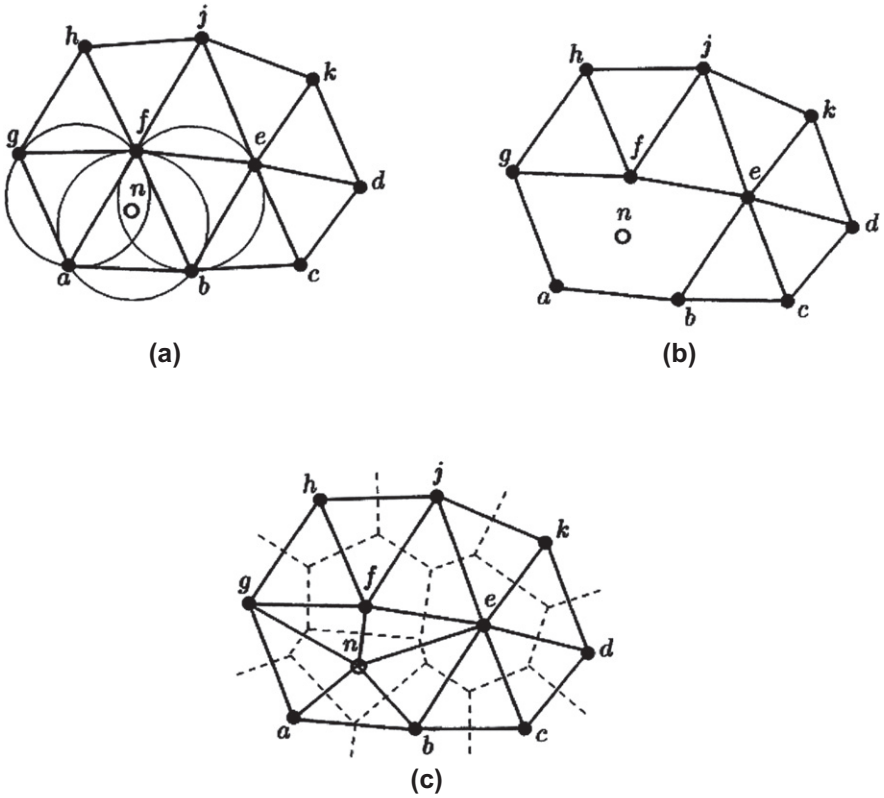



**FIGURE 17.33**

(a) Insertion of new forming point *n* into Delaunay triangulation. (b) Removal of Delaunay triangles; deleted Voronoi vertices are not shown. (c) New Delaunay triangulation and Voronoi diagram.

Schroeder and Shephard [15], Baker [16], and George et al. [17]; the early work of Rebay [18], Weatherill and Hassan [19], Shewchuk [92,93] on automatic point generation algorithms; and the work of Weatherill and Hassan [19], George et al. [17,94], Du and Wang [95] on preserving the integrity of the domain boundary.

In the following, we shall introduce the three-dimensional mesh generation procedure of Weatherill and Hassan, which is one of the first Delaunay mesh generation procedures that contains all three necessary components for a robust three-dimensional Delaunay mesh generation algorithm. It includes a Delaunay triangulation algorithm; a node generation algorithm based on specified mesh size distribution; and a surface mesh recovery procedure that ensure the integrity of the boundary surface.

The global procedure of the three-dimensional mesh generation algorithm is as follows:

1. Input the triangular surface mesh and derive the topological data of the surface mesh, such as edges of surface elements and node element connections. (Figure 17.34 shows the surface mesh of a simple three-dimensional geometry.)
2. Build a convex hull that contains all the mesh points. (An eight-node convex hull is shown in Fig. 17.35.)
3. Perform Delaunay triangulation using nodes of the surface mesh to form tetrahedra. (Figure 17.36a illustrates the Delaunay triangulation of the surface nodes. A cross-section of the triangulation is shown in Fig. 17.36b.)
4. Create interior points, following the specified element size distribution function, and perform Delaunay triangulation to form tetrahedra. (The results are shown in Figs. 17.37a and b.)
5. Recover any missing edges and triangular faces of the surface mesh to ensure the input surface triangulation being contained in the volume triangulation.
6. Identify and remove all the tetrahedra outside the domain of interest to give the final three-dimensional mesh. (Figure 17.38a shows the recovered surface mesh, which is identical to the input mesh, and Fig. 17.38b demonstrates the interior of the tetrahedral mesh at a cross-section of the geometry.)

The mesh generation procedure has been shown to be computationally efficient and, probably more importantly, very robust. Indeed, the robustness of the algorithm

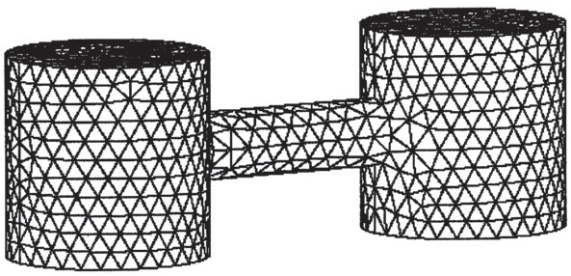


**FIGURE 17.34**
Surface mesh of a simple three-dimensional geometry.

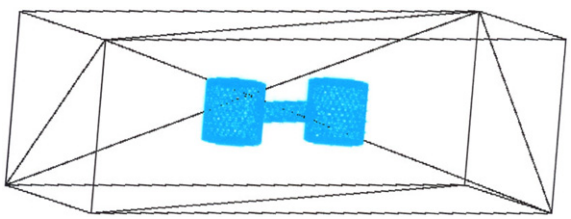


**FIGURE 17.35**
Convex hull and surface mesh.

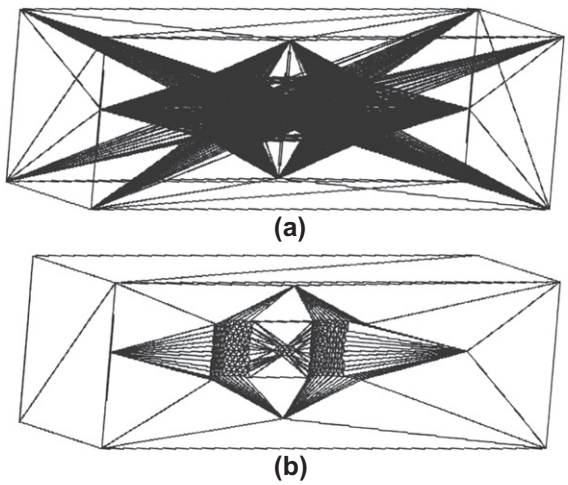



**FIGURE 17.36**
(a) Delaunay triangulation of the surface nodes. (b) A cross-section of the surface nodes Delaunay triangulation. Six additional points are inserted before the surface nodes for efficiency.

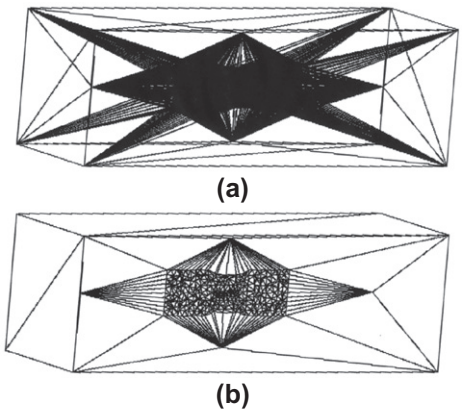



**FIGURE 17.37**
(a) Delaunay triangulation of the interior nodes. (b) A cross-section of the interior nodes Delaunay triangulation.

is independent of the complexity of the three-dimensional geometry. We shall, in the following, describe in more detail the three main components of the procedure, i.e., the Delaunay triangulation algorithm, the node creation algorithm, and the surface mesh recovery methods.

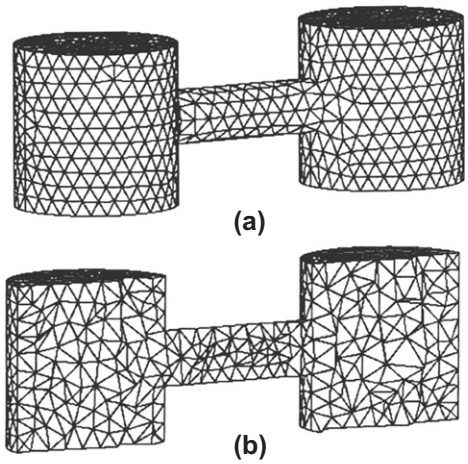



**FIGURE 17.38**
(a) Recovered surface. (b) Tetrahedral mesh shown at a cross-section.

#### 17.5.2.1 *Delaunay triangulation algorithm*

The Delaunay triangulation algorithm discussed below is based on the algorithm proposed by Bowyer [11], but it can be readily replaced by the similar algorithm of Watson [12] which differs only in its data structure.

The process of generating Delaunay triangulation is sequential. Each point is introduced into an existing structure of the Voronoi diagram and the Delaunay triangulation, which will be reformulated based on the in-circle criterion to form a new Delaunay triangulation. The process is similar to that described in Example 17.7, but of course now is in three dimensions. The main steps of the procedure are as follows:

1. Define a set of points which form a convex hull that encloses all the points to be used in the tetrahedral mesh.
2. Introduce a new point into the convex hull.
3. Determine all vertices of the Voronoi diagram to be deleted. A vertex will be deleted if the circumsphere, centered at the vertex, of four forming points contains the new point. This follows from the in-circle criterion.
4. Find the forming points of all the deleted Voronoi vertices. These form the contiguous points to the new point.
5. Determine the neighboring Voronoi vertices to the deleted vertices which have not themselves been deleted. This data provides the necessary information to enable valid combinations of the contiguous points to be constructed.
6. Determine the forming points of the new Voronoi vertices. The forming points of new vertices include the new point together with three points which are contiguous to the new point and form a face of a neighboring tetrahedra. This forms the new Delaunay triangulation.

7. Determine the neighboring Voronoi vertices to the new vertices.
   In Step 6, the forming points of all new vertices have been computed. For each new vertex, perform a search through the forming points of the neighboring vertices, as found in Step 5, to identify common triples of forming points. When a common combination occurs, the neighbor of the new Voronoi vertex has been found. This forms the new Voronoi diagram.
8. Reorder the Voronoi diagram data structure and replace (overwrite) the entries of the deleted vertices.
9. Repeat Steps 2–8 until all the point have been inserted.

In the mesh generation process, the new points to be inserted into the Delaunay triangulation and Voronoi diagram are the surface mesh nodes and the mesh points generated automatically following the adopted node generation algorithm.

#### 17.5.2.2 *Automatic node generation*

The detailed step by step node generation algorithm described below creates points based on the element size distribution of the surface mesh:

1. The node spacing function for each surface mesh node $a$ of position $\mathbf{x}_a$ is taken as the average of the surface element edge length,

$$h_a = \frac{1}{M}\sum_{b=1}^{M} \|\mathbf{x}_b - \mathbf{x}_a\| \tag{17.76}$$

   where $\mathbf{x}_b$ $(b = 1, 2, \ldots, M)$ are positions of the surface nodes connected to node $a$.
2. Perform Delaunay triangulation using surface mesh nodes.
3. Initialize the number of new interior points to be created, set $N = 0$.
4. For each tetrahedron within the domain,

   **(a)** Locate a prospective point $c$ at the centroid of the tetrahedron.
   **(b)** Derive the node spacing function $h_c$, for point $c$, by interpolating the node spacing function $h_m$ $(m = 1, 2, 3, 4)$ from the nodes of the tetrahedron.
   **(c)** Compute the distances $d_m$ $(m = 1, 2, 3, 4)$ from the prospective point $c$ to each of the four nodes of the tetrahedron.

   If $\{d_m < \alpha h_c\}$ for any $m = 1, 2, 3, 4$ then
   reject the point and return to the beginning of Step 4 for the next tetrahedron.
   Else
   compute the distance $d_j$ from the prospective point $c$, to other already created nodes $p_j (j = 1, 2, \ldots, N)$.

   If $\{d_j < \beta h_c\}$ then
   reject the point and return to the beginning of Step 4 for the next tetrahedron.

Else
accept the point $c$ and add it to the interior node list $p_j$ ($j = 1, 2, \ldots, N$) and update N.

**(d)** Assign point distribution function $h_c$ to new node $c$.
**(e)** Go to the next tetrahedron.

**5.** If $N = 0$, i.e., no new point is created, exit the node generation process.
**6.** Perform the Delaunay triangulation of the derived points, $p_j$ ($j = 1, 2, \ldots, N$). Then, go to Step 3.

The parameter $\alpha$ controls the element density by changing the allowable shape of the formed tetrahedra, while $\beta$ has an influence on the regularity of the triangulation by not allowing points within a specified distance of each other. Both parameters can be adjusted to control the mesh density. In practical computations, $\alpha$ can be chosen in the range of 0.85–1.1, and $\beta$ in the range of 0.6–1.0 for an isotropic mesh. An effort to find the optimal value of these parameters has been made in Ref. [96].

It is noted that, with minor modification, the node generation algorithm can also create node-based element size distribution defined by a three-dimensional background mesh.

#### 17.5.2.3 Surface mesh recovery

Property (iv) of the Voronoi diagram and Delaunay triangulation presented in Section 17.5.1 implies that the surface mesh and the surface boundary of a general three-dimensional geometry, which is seldom convex, will not be respected during the mesh generation process. Very often, some of the surface triangles and their edges are not present in the resulting Delaunay triangulation due to penetrations by other tetrahedra. The loss of completeness of the original surface mesh causes the loss of integrity of the surface boundary of the geometry. In order to derive a valid three-dimensional mesh for the given geometry, the integrity of the surface boundary of the geometry must be respected, which can be realized by recovering the original surface mesh.

In the surface mesh recovery procedure, the surface triangles and surface edges that are missing from the Delaunay triangulation are first identified and then restored by the following procedure:

*Edge swapping.* Edge swapping is illustrated in Fig. 17.39. If faces $abd$ and $bcd$ appear in the Delaunay triangulation, but faces $abc$ and $acd$ exist in the surface mesh, replacing edge $bd$ by edge $ac$ recovers two surface triangles. This process is attempted for each surface edge because it is the most efficient method to recover the missing edges and triangular faces.

*Boundary edge recovery.* Consider the case when the surface edge joining points $a$ and $b$ is missing from the Delaunay triangulation, a line $ab$ is formed, and its intersection with the faces, edges, and points of the Delaunay triangulation are identified, as shown in Fig. 17.40, with all the possible types of intersections depicted in Fig. 17.41. Local transformations with the newly added nodes at the intersection, as shown in Fig. 17.42,

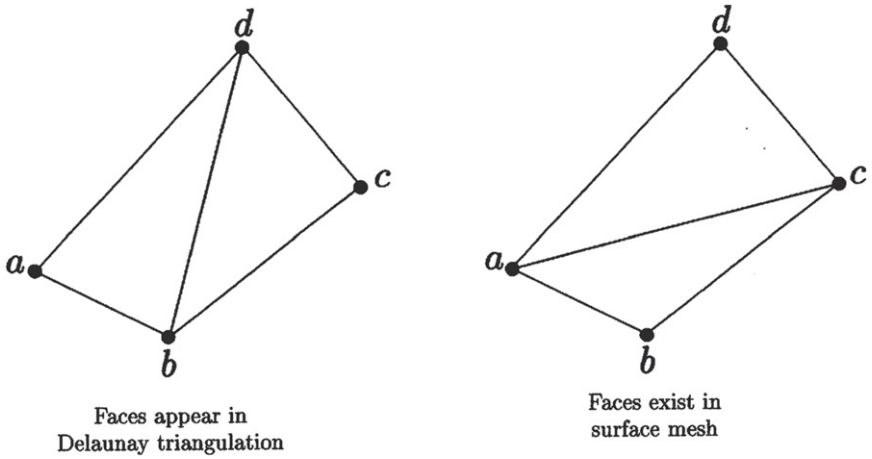



**FIGURE 17.39**
Edge swapping to recover the surface edge and faces.

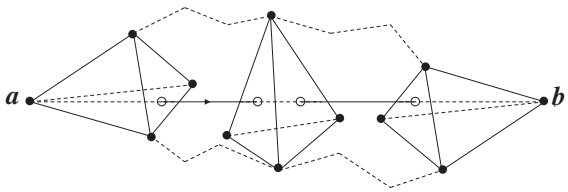



**FIGURE 17.40**
Edge *ab* of a surface triangle missing in the Delaunay triangulation.

are performed on all the involved tetrahedra to recover the edge, segment by segment. This process is executed for every missing edge.

When the combined intersection involves a node-to-face type and a face-to-node type, edge *ab* can be recovered by directly linking nodes *a* and *b*, with the two involved tetrahedra transformed to three tetrahedra as shown in Fig. 17.43.

*Boundary face recovery.* The recovery of the surface triangles is conducted after the completion of the edge recovery. A surface triangle may still be missing from the tetrahedral mesh even though all its edges are present, because the interior of the triangle face is penetrated by other tetrahedra. There are a total of four possibilities where a face can be intersected by a tetrahedra as illustrated in Fig. 17.44. Every missing face can be recovered after all the intersecting tetrahedra are determined and transformed, with newly added points, according to their intersection type. The intersecting tetrahedron shown in Fig. 17.44 is transformed to a combined shape of a tetrahedron, a pyramid, or a prism which can be further divided into tetrahedra as illustrated in Figs. 17.45 and 17.46.

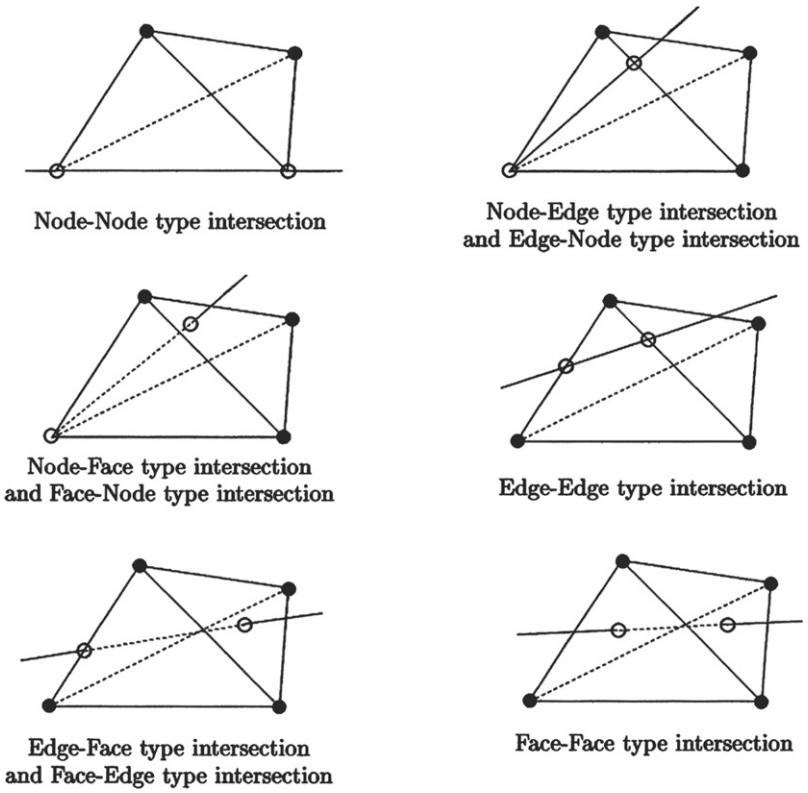



**FIGURE 17.41**
Edge-tetrahedron intersections for missing surface edges.

When a face is intersected by only one edge and the edge is common to three tetrahedra, the face can be recovered directly by deleting the edge, with three tetrahedra transformed to two as shown in Fig. 17.47.

*Removal of added points.* The points that are added in the process of recovering the boundary edges and faces will be removed one by one together with the connected tetrahedra. The empty polyhedron left after the deletion of each added point and its connected tetrahedra will be triangulated directly, often with additional interior points.

It is noted, with reference to the global mesh generation procedure, that once Step 5 of the global procedure for surface recovery starts, the tetrahedral mesh, in general, does not continue to be a Delaunay triangulation, which may seriously damage the quality of the mesh near the boundary. Therefore, mesh quality enhancement becomes indispensable after the completion of the mesh generation procedure.

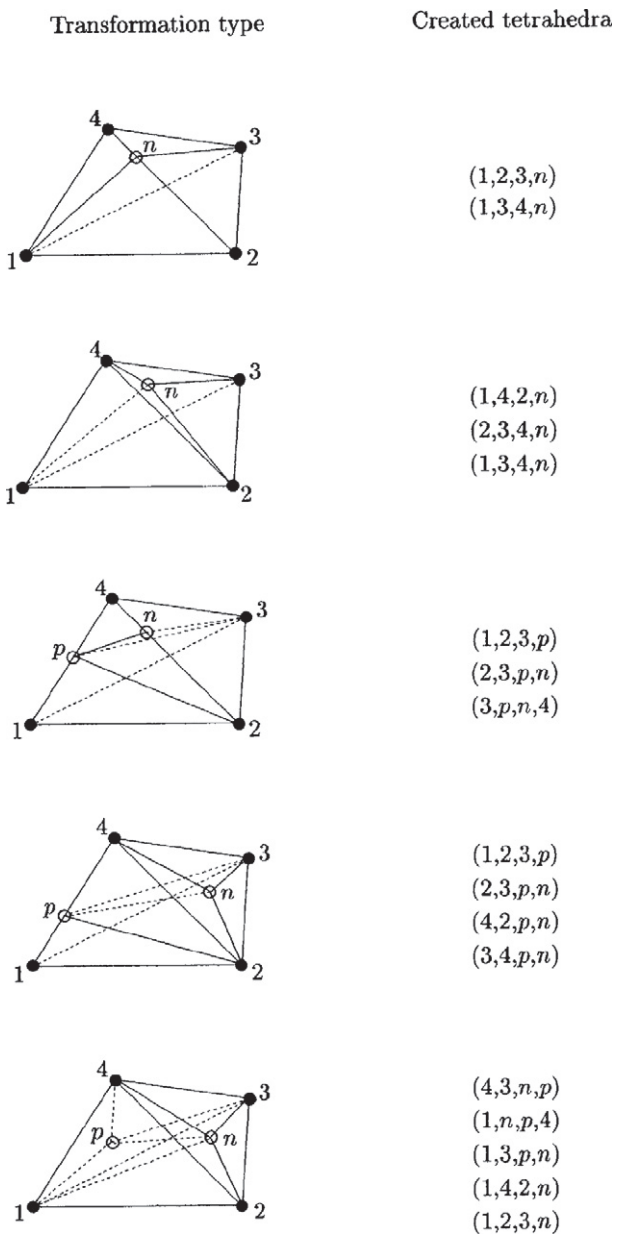



**FIGURE 17.42**
Tetrahedral transformation to recover a segment of the missing edges.

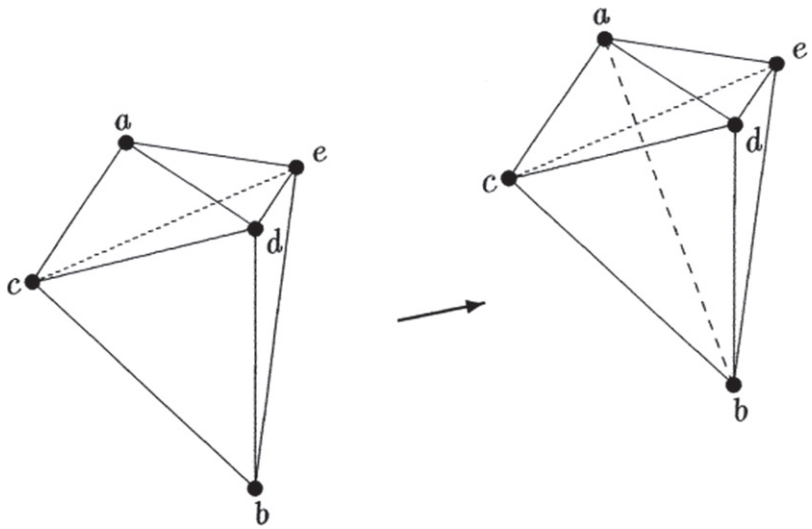



**FIGURE 17.43**
Recovery of boundary edge *ab* by the deletion of face *cde*.

### 17.5.3 Mesh quality enhancement

In three-dimensional mesh generation, mesh quality enhancement is almost as important as element generation itself. This is because some poorly shaped tetrahedral elements are created either in the Delaunay triangulation process, due to the position of the inserting points, or in the surface mesh recovery process. Without applying certain mesh quality enhancement procedures to improve the element quality, these poorly shaped elements may render the three-dimensional mesh unusable in finite element computation. Unlike in two-dimensional mesh generation, the process of improving the quality of a three-dimensional mesh is much more complicated and tedious. The quality of a tetrahedral element may be evaluated by different measures. A wide range of measures for the quality of tetrahedral elements are presented in Refs. [97–101], and any one of these measures can be employed as criterion in the mesh quality enhancement procedure. Here we shall not get into the details of a particular quality measure, but are mainly concerned with the methodologies that can be used to improve the quality of the tetrahedral elements, under any specified quality measure. Several effective element quality enhancement methods are described below.

#### *17.5.3.1 Element transformation*

Modifying the topological structure of the mesh is probably the most effective way to improve the quality of the mesh in three dimensions and is realized by performing element transformations of the following form:

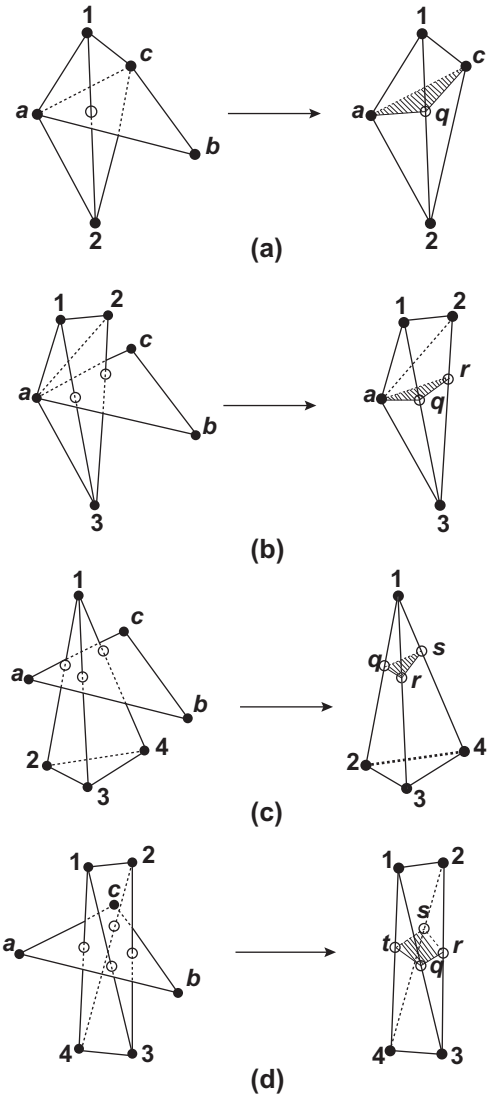



**FIGURE 17.44**

Transformations for the recovery of surface faces. Shaded area shows the recovered portion of the boundary face. (a) One edge intersects face *abc*. (b) Two edges intersect face *abc*. (c) Three edges intersect face *abc*. (d) Four edges intersect face *abc*.

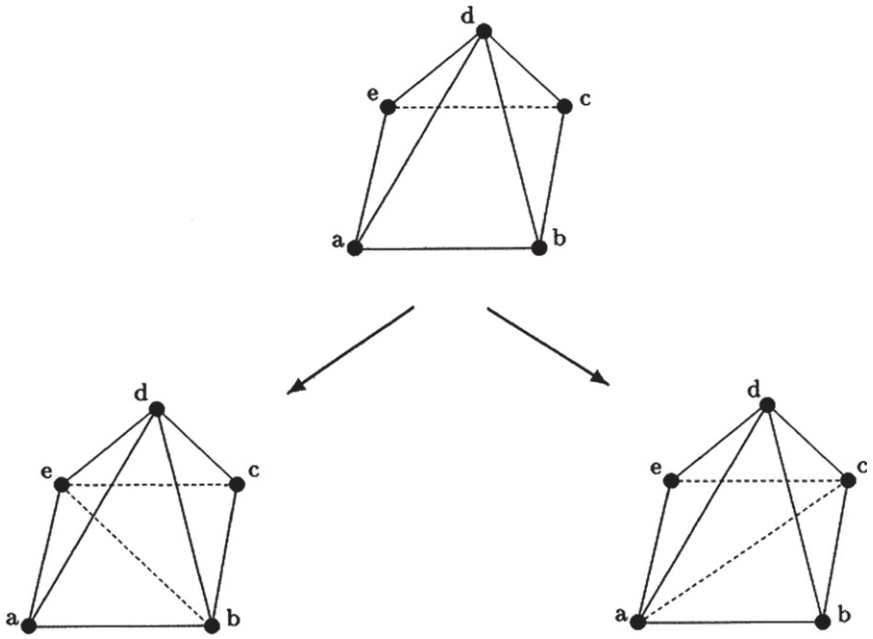



**FIGURE 17.45**
Different patterns of dividing a pyramid to tetrahedra.

*Two-element transformation.* Two elements common to a face can be transformed to three elements as shown in Fig. 17.48, if one of the elements does not satisfy the quality criterion. To ensure that the new elements are valid, the new edge $ab$ must intersect the removed face $cde$.

*Three-element transformation.* As an inverse of the two-element transformation, three elements common to an edge are transformed to two elements as illustrated in Fig. 17.48, if one of the elements does not meet the quality criterion.

*Four-element transformation.* Four elements common to an edge can be transformed to two topologically different patterns of four elements common to an edge as depicted in Fig. 17.49, when one of the elements fails to satisfy the quality criterion.

*Five or more element transformation.* A split-collapse procedure is used for element transformation when an edge is common to five or more elements. For an element that fails the quality criterion, e.g., element $abcd$ as shown in Fig. 17.50a, find its edge $ab$ and all the elements common to the edge (five elements are shown in Fig. 17.50a). A node $n$ is added to the middle of the edge $ab$ and splitting the elements as illustrated in Fig. 17.50b. The new node $n$ is then collapsed to one of its connecting nodes, except nodes $a$ and $b$, to form new elements as demonstrated in Fig. 17.50c (six new elements are formed). This procedure can be used for an edge common to any number of elements and is attempted for each edge of the element.

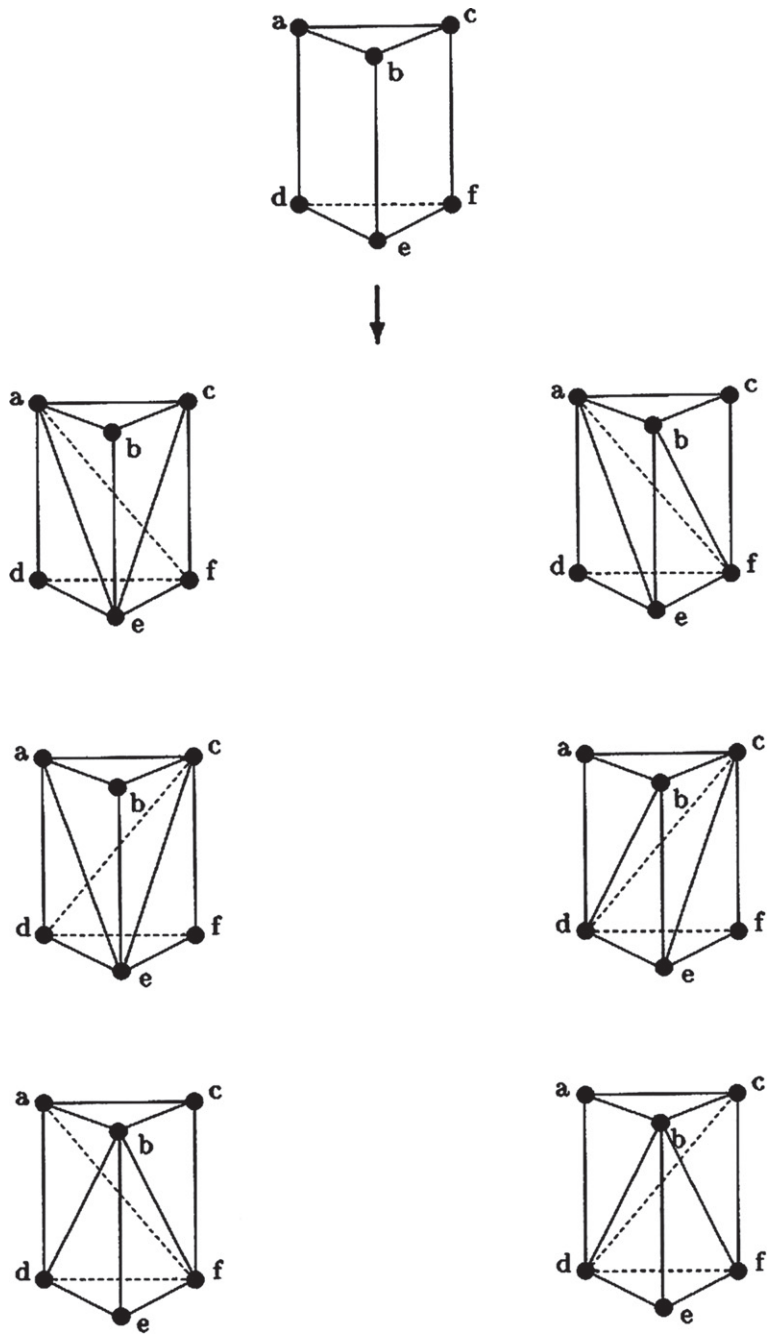



**FIGURE 17.46**
Different patterns of dividing a prism to tetrahedra.

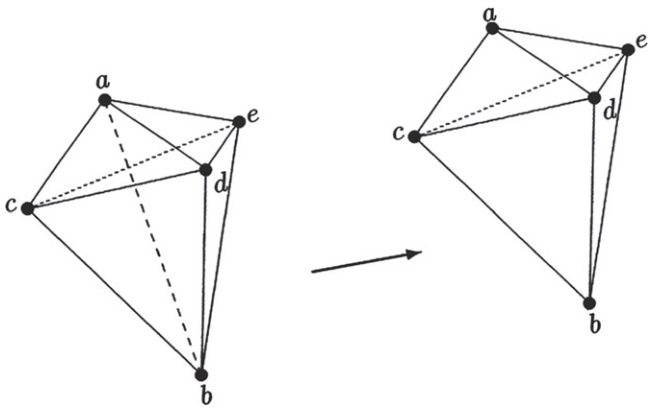



**FIGURE 17.47**
Recovery of boundary face *cde* by the deletion of edge *ab*.

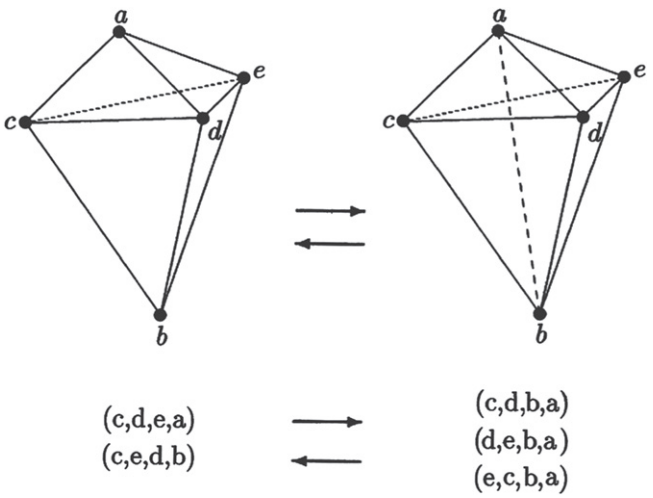



**FIGURE 17.48**
Two elements transformed to three elements and three elements transformed to two elements.

All the transformations are carried out under the condition that the worst quality measure improves after the transformation.

### 17.5.3.2 *Node addition and node elimination*

Unlike element transformations which will only change the topological structure of the mesh, node addition and node elimination will locally change the node density of the mesh.

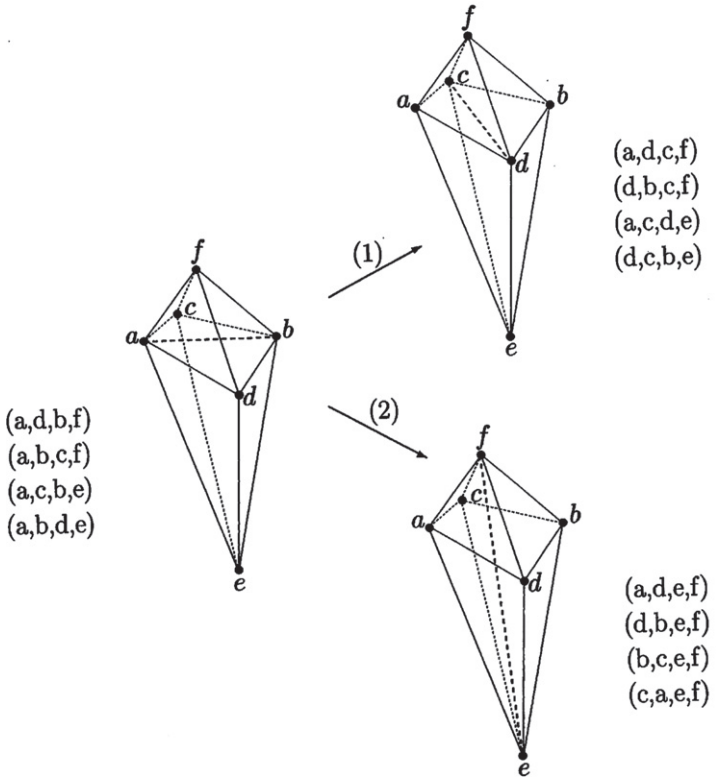



**FIGURE 17.49**
Two patterns of four elements transformed to four elements. (1) Edge *ab* changed to edge *cd*. (2) Edge *ab* changed to edge *ef*.

*Node addition.* A node is added to an edge of the element if the edge is deemed too long. This requires all the elements common to the edge to be split; the process is shown in Fig. 17.50a and b.
*Node elimination.* A node of the element that fails the quality criterion and all the elements connected to it are shown in Fig. 17.51a. The node is collapsed to one of its connecting nodes, as illustrated in Fig. 17.51b, so that the quality of the resulting elements improves. The procedure is attempted for each node of the element.

The quality of the elements can be further improved if the positions of the interior nodes are repositioned, which leads to so-called mesh smoothing.

### 17.5.3.3 *Mesh smoothing*

The standard Laplacian smoothing cannot be applied directly to a tetrahedral mesh. It in fact reduces the quality of the mesh. The procedure is modified to move a node

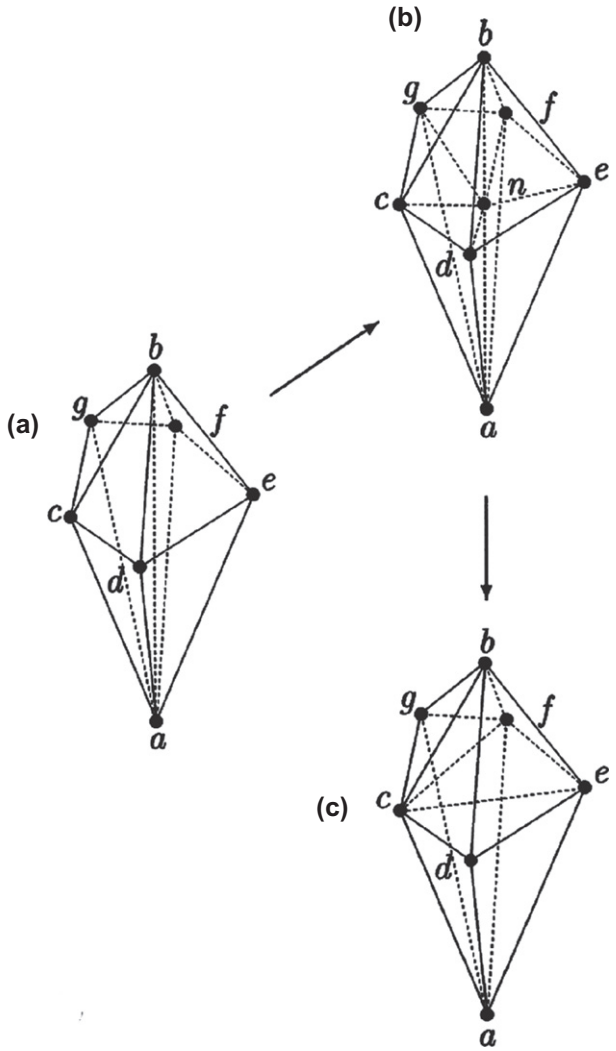



**FIGURE 17.50**

Five or more element transformation. (a) Edge *ab* and all the elements sharing it. (b) Node *n* is added to edge *ab*; all the common elements are split. (c) New node *n* is collapsed to node *c*.

incrementally and iteratively toward each of its connecting nodes and is placed at the position that will increase the quality of the worst element. The procedure stops when the quality of the connected elements does not improve. Several combined applications of the quality enhancement method usually results in a mesh with much improved quality.

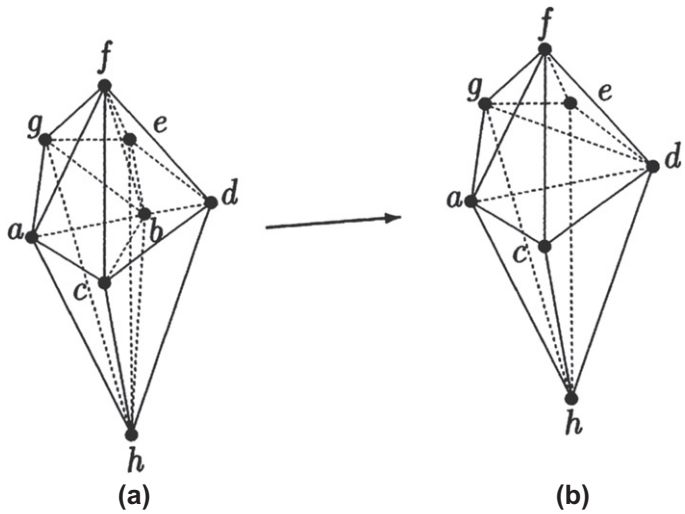



**FIGURE 17.51**
Node collapsing. (a) Node *b* and its connecting elements. (b) Collapse node *b* to node *d*.

It is noted that the condition attached to all the quality enhancement methods, which requires the worst element quality to improve according to an element quality criterion, corresponds to an optimization problem. Its application guarantees improvement of the element quality, but could also be computationally expensive. For additional information on mesh enhancement methods using a specific quality measure as the objective function in the optimization process we refer to Refs. [102–107].

### 17.5.4 Higher order elements

With higher order surface mesh available, higher order tetrahedral elements can be readily obtained by finding the positions of the intermediate nodes using linear interpolation.

### 17.5.5 Numerical examples

Tetrahedral meshes are generated using the mesh generation procedure of Delaunay triangulation described in this section. Figure 17.52a shows the mesh for a flask body casting. A cross-section of the tetrahedral mesh is illustrated in Fig. 17.52b to demonstrate the regularity of the mesh. Figure 17.53 presents a tetrahedral mesh for a complete V8 engine block.

### 17.5.6 Remarks

**Remark 9.** The automatic node generation procedure described in this section was introduced by Weatherill and Hassan; many other node generation methods also

exist. Indeed, the procedure by which new nodes are generated is the main difference between various Delaunay mesh generation algorithms reported in the literature. From our discussion, it is clear that once the points are available, a mesh can always be generated following a Delaunay triangulation algorithm. Therefore, it is important to adopt a suitable node generation method that can meet the specific requirements for a particular application, so that an optimal mesh can be obtained for finite element computation. We refer to Refs. [92, 108–114] for additional information.

**Remark 10.** As we mentioned in the beginning of the chapter, it is still a demanding and challenging task to generate structured or unstructured hexahedral meshes automatically. In Chapter 10, we have demonstrated that some hexahedral elements have advantages in the finite element computation for incompressible materials; in

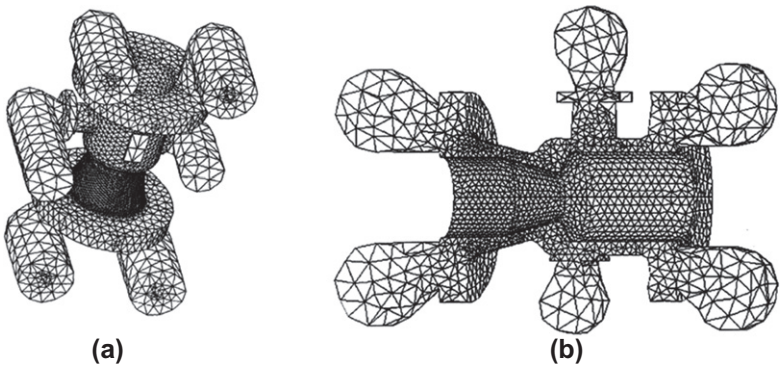



**FIGURE 17.52**
(a) Tetrahedral mesh of casting part of flask body. (b) A cross-section of the tetrahedral mesh.

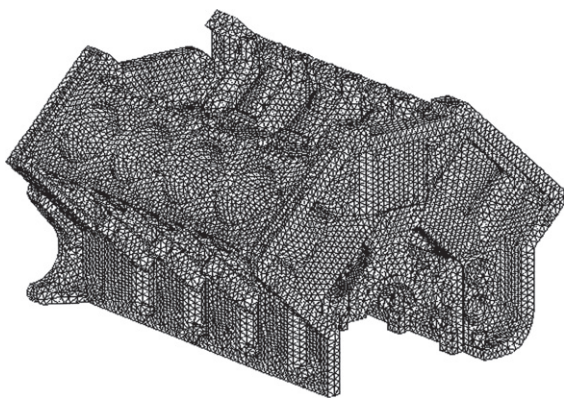


**FIGURE 17.53**
Tetrahedral mesh of a V8 engine block.

addition they generally have better accuracy compared to tetrahedral elements of the same order. Significant progress has been made recently by Zhang et al. [41–43] and Ito et al. [40] in developing automatic mesh generation algorithms for hexahedral elements using Octree-based methods.

## 17.6 Concluding remarks

We have shown in this chapter how to generate mesh on curves, in arbitrary two-dimensional domains, on curved surfaces, and for realistic three-dimensional geometries. We presented detailed discussion as well as algorithmic procedures for curve and surface mesh generation. We also described the advancing front method in two-dimensional mesh generation and the Delaunay triangulation method in three-dimensional mesh generation. The algorithms and methodologies presented in this chapter are not only robust and have been implemented for science and engineering applications, but also provide a basis for further research in the development of various aspects of automatic mesh generation methods. Additional applications of the automatic mesh generation methods discussed in this chapter have been presented in Chapter 16 for adaptive finite element analysis and appear in Ref. [51] for fluid dynamics applications.

## References

[1] O.C. Zienkiewicz, D.V. Phillips, An automatic mesh generation scheme for plane and curved surfaces by isoparametric coordinates, Int. J. Numer. Methods Eng. 3 (1971) 519–528.
[2] W.J. Gordon, C.A. Hall, Construction of curvilinear co-ordinate systems and application to mesh generation, Int. J. Numer. Methods Eng. 3 (1973) 461–477.
[3] J.F. Thompson, F.C. Thames, C.W. Mastin, Automatic numerical generation of body-fitted curvilinear coordinates for a field containing any number of arbitrary two-dimensional bodies, J. Comput. Phys. 15 (1974) 299–319.
[4] J.F. Thompson, Z.U.A. Warsi, Boundary-fitted coordinate systems for numerical solution of partial differential equations, J. Comput. Phys. 47 (1982) 1–108.
[5] J.F. Thompson, Z.U.A. Warsi, C.W. Martin, Numerical Grid Generation: Foundations and Applications, North-Holland, Dordrecht, 1987.
[6] T.K.H. Tam, C.G. Armstrong, 2D finite element mesh generation by medial axis subdivision, Adv. Eng. Softw. 13 (1991) 313–324.
[7] M.A. Price, C.G. Armstrong, M.A. Sabin, Hexahedral mesh generation by medial axis subdivision. I: Solids with convex edges, Int. J. Numer. Methods Eng. 38 (1995) 3335–3359.
[8] M.A. Price, C.G. Armstrong, M.A. Sabin, Hexahedral mesh generation by medial axis subdivision. II: Solids with flat and concave edges, Int. J. Numer. Methods Eng. 40 (1997) 111–136.

[9] N. Chiba, I. Nishigaki, Y. Yamashita, C. Takizawa, K. Fujishiro, A flexible automatic hexahedral mesh generation by boundary-fit method, Comput. Methods Appl. Mech. Eng. 161 (1998) 145–154.
[10] A. Sheffer, M. Bercovier, Hexahedral meshing of non-linear volumes using Voronoi faces and edges, Int. J. Numer. Methods Eng. 49 (2000) 329–351.
[11] A. Bowyer, Computing Dirichlet tessellations, Comput. J. 24 (2) (1981) 162–166.
[12] D.F. Watson, Computing the $n$-dimensional Delaunay tessellation with application to Voronoi polytopes, Comput. J. 24 (1981) 167–172.
[13] J.C. Cavendisha, D.A. Field, W.H. Frey, An approach to automatic three dimensional finite element mesh generation, Int. J. Numer. Methods Eng. 21 (1985) 329–347.
[14] N.P. Weatherill, A method for generating irregular computation grids in multiply connected planar domains, Int. J. Numer. Methods Eng. 8 (1988) 181–197.
[15] W.J. Schroeder, M.S. Shephard, Geometry-based fully automatic mesh generation and the Delaunay triangulation, Int. J. Numer. Methods Eng. 26 (1988) 2503–2524.
[16] T.J. Baker, Automatic mesh generation for complex three-dimensional regions using a constrained Delaunay triangulation, Eng. Comput. 5 (1989) 161–175.
[17] P.L. George, F. Hecht, E. Saltel, Automatic mesh generator with specified boundary, Comput. Methods Appl. Mech. Eng. 92 (1991) 269–288.
[18] S. Rebay, Efficient unstructured mesh generation by means of Delaunay triangulation and Bowyer–Watson algorithm, J. Comput. Phys. 106 (1993) 125–138.
[19] N.P. Weatherill, O. Hassan, Efficient 3-dimensional Delaunay triangulation with automatic point generation and imposed boundary constraints, Int. J. Numer. Methods Eng. 37 (1994) 2005–2039.
[20] D.L. Marcum, N.P. Weatherill, Unstructured grid generation using iterative point insertion and local reconnection, AIAA J. 33 (9) (1995) 1619–1625.
[21] S.H. Lo, A new mesh generation scheme for arbitrary planar domains, Int. J. Numer. Methods Eng. 21 (1985) 1403–1426.
[22] J. Peraire, M. Vahdati, K. Morgan, O.C. Zienkiewicz, Adaptive remeshing for compressible flow computations, J. Comput. Phys. 72 (1987) 449–466.
[23] J. Peraire, J. Peiró, L. Formaggio, K. Morgan, O.C. Zienkiewicz, Finite element Euler computations in three dimensions, Int. J. Numer. Methods Eng. 26 (1988) 2135–2159.
[24] R. Löhner, P. Parikh, Three-dimensional grid generation by the advancing-front method, Int. J. Numer. Methods Eng. 8 (1988) 1135–1149.
[25] S.H. Lo, Volume discretization into tetrahedral—II: 3D triangulation by advancing front approach, Comput. Struct. 39 (5) (1991) 501–511.
[26] H. Jin, R.I. Tanner, Generation of unstructured tetrahedral meshes by advancing front technique, Int. J. Numer. Methods Eng. 36 (1993) 1805–1823.
[27] P.L. George, E. Seveno, The advancing-front mesh generation method revisited, Int. J. Numer. Methods Eng. 37 (1994) 3605–3619.

[28] P. Möller, P. Hansbo, On advancing front mesh generation in three dimensions, Int. J. Numer. Methods Eng. 38 (1995) 3551–3569.
[29] M.A. Yerry, M.S. Shephard, Automatic three-dimensional mesh generation by the modified octree technique, Int. J. Numer. Methods Eng. 20 (1984) 1965–1990.
[30] P.L. Baehmann, S.L. Wittchen, M.S. Shephard, K.R. Grice, M.A. Yerry, Robust geometrically-based, automatic two-dimensional mesh generation, Int. J. Numer. Methods Eng. 24 (1987) 1043–1078.
[31] M.S. Shephard, M.K. Georges, Automatic three-dimensional mesh generation by the finite octree technique, Int. J. Numer. Methods Eng. 32 (1991) 709–749.
[32] T.D. Blacker, R.J. Meyers, Seams and wedges in plastering: a 3D hexahedral mesh generation algorithm, Eng. Comput. 2 (9) (1993) 83–93.
[33] T.J. Tautges, T.D. Blacker, S.A. Mitchell, The Whisker-Weaving algorithm: a connectivity-based method for constructing all-hexahedral finite element meshes, Int. J. Numer. Methods Eng. 39 (1996) 3327–3349.
[34] N.A. Calvo, S.R. Idelsohn, All-hexahedral element meshing: generation of the dual mesh by recurrent subdivision, Comput. Methods Appl. Mech. Eng. 182 (2000) 371–378.
[35] G. Dhondt, A new automatic hexahedral mesher based on cutting, Int. J. Numer. Methods Eng. 50 (2001) 2109–2126.
[36] M.L. Staten, R.A. Kerr, S.J. Owen, T.D. Blacker, M. Stupazzini, K. Shimada, Unconstrained plastering–hexahedral mesh generation via advancing–front geometry decomposition, Int. J. Numer. Methods Eng. 81 (2010) 135–171.
[37] R. Schneiders, A grid-based algorithm for the generation of hexahedral element meshes, Eng. Comput. 12 (1996) 168–177.
[38] R. Schneiders, R. Schindler, F. Weiler, Octree-based generation of hexahedral element meshes, in: Proceedings of the 5th International Meshing Roundtable, Sandia National Laboratories, 1996, pp. 205–216.
[39] Y. Zhang, C. Bajaj, Adaptive and quality quadrilateral/hexahedral meshing from volumetric data. Comput. Methods Appl. Mech. Eng. 195 (2006) 942–960.
[40] Y. Ito, A.M. Shih, B.K. Soni, Octree-based reasonable-quality hexahedral mesh generation using a new set of refinement templates, Int. J. Numer. Methods Eng. 77 (2009) 1809–1833.
[41] Y. Zhang, T.J.R. Hughes, C. Bajaj, An automatic 3d mesh generation method for domains with multiple materials, Comput. Methods Appl. Mech. Eng. 199 (2010) 405–415.
[42] J. Qian, Y. Zhang, Automatic unstructured all-hexahedral mesh generation from b-reps for non-manifold cad assemblies, Eng. Comput. 28 (2012) 345–359.
[43] Y. Zhang, X. Liang, G. Xu, A robust 2-refinement algorithm in octree or rhombic dodecahedral tree based all-hexahedral mesh generation, Comput. Methods Appl. Mech. Eng. 256 (2013) 88–100.
[44] J.F. Thompson, B.K. Soni, N.P. Weatherill (Eds.), Handbook of Grid Generation, CRC Press, January, 1999.

[45] P.J. Frey, P-L. George, Mesh Generation: Application to Finite Element, second ed., Wiley, London, 2000.
[46] L. Piegl, W. Tiller, The NURBS Book (Monographs in Visual Communication), second ed., Springer-Verlag, New York, 1997.
[47] D.F. Rogers, An Introduction to NURBS: With Historical Perspective, Academic Press, San Diego, 2000.
[48] G. Farin, Curves and Surfaces for CAGD: A Practical Guide, fifth ed., Morgan Kaufmann Publishers, San Francisco, CA, 2002.
[49] M.G. Cox, The Numerical Evaluation of B-Splines, Technical Report DNAC 4, National Physics Laboratory, 1971.
[50] C. de Boor, On calculation with B-splines, J. Approximation Theory 6 (1972) 50–62.
[51] O.C. Zienkiewicz, R.L. Taylor, P. Nithiarasu, The Finite Element Method for Fluid Dynamics, seventh ed., Elsevier, Oxford, 2013.
[52] J.C. Cavendish, Automatic triangulation of arbitrary planar domains for the finite element method, Int. J. Numer. Methods Eng. 8 (1974) 679–696.
[53] J. Wu, J.Z. Zhu, J. Szmelter, O.C. Zienkiewicz, Error estimation and adaptivity in Navier-Stokes incompressible flows, Comput. Mech. 6 (1990) 259–270.
[54] R. Löhner, Some useful data structures for the generation of unstructured grids, Commun. Appl. Numer. Methods 4 (1988) 123–135.
[55] J. Bonet, J. Peraire, An alternating digit tree (ADT) algorithm for 3D geometric search and intersection problems, Int. J. Numer. Methods Eng. 31 (1991) 1–17.
[56] W. Kwok, An efficient data structure for the advancing front triangular mesh generation technique, Commun. Numer. Methods Eng. 11 (1995) 465–473.
[57] J.Z. Zhu, O.C. Zienkiewicz, E. Hinton, J. Wu, A new approach to the development of automatic quadrilateral mesh generation, Int. J. Numer. Methods Eng. 32 (1991) 849–866.
[58] J.A. Talbert, A.R. Parkinson, Development of an automatic, two-dimensional finite element mesh generator using quadrilateral elements and Bezier curve boundary definitions, Int. J. Numer. Methods Eng. 29 (1990) 1551–1567.
[59] T.D. Blacker, M.B. Stephenson, Paving: a new approach to automated quadrilateral mesh generation, Int. J. Numer. Methods Eng. 32 (1991) 811–847.
[60] B.P. Johnston, J.M. Sullivan, A. Kwasnik, Automatic conversion of triangular finite element meshes to quadrilateral elements, Int. J. Numer. Methods Eng. 31 (1991) 67–84.
[61] E. Rank, M. Schweingruber, M. Sommer, Adaptive mesh generation, Commun. Numer. Methods Eng. 9 (1993) 121–129.
[62] C.K. Lee, S.H. Lo, A new scheme for the generation of a graded quadrilateral mesh, Comput. Struct. 52 (1994) 847–857.
[63] B. Joe, Quadrilateral mesh generation in polygonal regions, Comput. Aided Des. 27 (1995) 209–222.
[64] S.J. Owen, M.L. Staten, S.A. Canann, S. Saigal, Q-morph: an indirect approach to advancing front quad meshing, Int. J. Numer. Methods Eng. 44 (1999) 1317–1340.

[65] J. Peraire, J. Peiró, K. Morgan, Adaptive remeshing for 3-dimensional compressible flow computations, J. Comput. Phys. 103 (1992) 269–285.
[66] J. Peiró, Surface grid generation, in: Handbook of Grid Generation, CRC Press, 1999, pp. 19-1–19-20 (Chapter 19).
[67] I.D. Faux, M.J. Pratt, Computational Geometry for Design and Manufacture, Ellis Horwood, 1985.
[68] W.H. Press et al. (Eds.), Numerical Recipes in Fortran: The Art of Scientific Computing, second ed., Cambridge University Press, Cambridge, 1992.
[69] O.C. Zienkiewicz, R.L. Taylor, D. Fox, The Finite Element Method for Solid and Structural Mechanics, seventh ed., Elsevier, Oxford, 2013.
[70] P. Hansbo, Generalized Laplacian smoothing of unstructured grids, Commun. Numer. Methods Eng. 11 (1995) 455–464.
[71] Y. Zheng, R.W. Lewis, D.T. Gethin, Three-dimensional unstructured mesh generation. Part 2: Surface meshes, Comput. Methods Appl. Mech. Eng. 134 (1996) 269–284.
[72] M. Suzuki, Surface grid generation with linkage to geometric generation, Int. J. Numer. Methods Eng. 34 (1993) 163–176.
[73] C.K. Lee, Automatic metric advancing front triangulation over curved surfaces, Eng. Comput. 17 (1) (2000) 48–74.
[74] H. Borouchaki, P. Laug, P.L. George, Parametric surface meshing using combined advancing-front generalized Delaunay approach, Int. J. Numer. Methods Eng. 49 (2000) 233–259.
[75] S.J. Sherwin, J. Peiró, Mesh generation in curvilinear domain using high-order elements, Int. J. Numer. Methods Eng. 53 (2002) 207–223.
[76] Y.K. Lee, C.K. Lee, Automatic generation of anisotropic quadrilateral meshes on three-dimensional surfaces using metric specifications, Int. J. Numer. Methods Eng. 53 (2002) 2673–2700.
[77] Z. Guan, J. Shan, Y. Zheng, Y. Gu, An extended advancing front technique for closed surfaces mesh generation, Int. J. Numer. Methods Eng. 74 (2008) 642–667.
[78] R. Löhner, Regridding surface triangulations, J. Comput. Phys. 126 (1996) 1–10.
[79] A. Rassineux, P. Villon, J.-M. Savignat, O. Stab, Surface remeshing by local Hermite diffuse interpolation, Int. J. Numer. Methods Eng. 49 (2000) 31–49.
[80] Y. Ito, K. Nakahashi, Surface triangulation for polygonal models based on CAD data, Int. J. Numer. Methods Fluids 39 (2002) 75–96.
[81] C.K. Lee, Automatic metric 3D surface mesh generation using subdivision surface geometrical model. Part 1: Construction of underlying geometrical model, Int. J. Numer. Methods Eng. 56 (2003) 1593–1614.
[82] C.K. Lee, Automatics metric 3D surface mesh generation using subdivision surface geometrical model. Part 2: Mesh generation algorithm and examples, Int. J. Numer. Methods Eng. 56 (2003) 1615–1646.

[83] D. Wang, O. Hassan, K. Morgan, N. Weatherill, Enhanced remeshing from STL files with applications to surface grid generation, Commun. Numer. Methods Eng. 23 (2006) 227–239.
[84] R. Aubry, G. Houzeaux, M. Vázquez, A surface remeshing approach, Int. J. Numer. Methods Eng. 85 (2011) 1475–1498.
[85] J.R. Cebral, R. Löhner, From medical images to anatomically accurate finite element grids, Int. J. Numer. Methods Eng. 51 (2001) 985–1008.
[86] P.J. Frey, Generation and adaptation of computational surface meshes from discrete anatomical data, Int. J. Numer. Methods Eng. 60 (2004) 1049–1074.
[87] Y. Ito, P.C. Shum, A.M. Shih, B.K. Soni, K. Nakahashi, Robust generation of high-quality unstructured meshes on realistic biomedical geometry, Int. J. Numer. Methods Eng. 65 (2006) 943–973.
[88] B. Delaunay, Sur la sphère vide, Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk 7 (1934) 793–800.
[89] G. Voronoi, Nouvelles applications des paramètres continus ă la théorie des formes quadratiques, J. Reine Angew. Math. 133 (1907) 97–178.
[90] F.P. Preparata, M.I. Shamos, Computational Geometry, Springer-Verlag, New York, 1988.
[91] J. O'Rourke, Computational Geometry in C, second ed., Cambridge University Press, 2001.
[92] J.R. Shewchuk, Delaunay Refinement Mesh Generation. Ph.d thesis, Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, 1997.
[93] J.R. Shewchuk, Tetrahedral mesh generation by delaunay refinement, in: Proceedings of the 14th Annual Symposium on Computational Geometry, Minneapolis, MN, 1998, pp. 86–95.
[94] P.L. George, H. Borouchaki, E. Saltel, Ultimate robustness in meshing an arbitrary polyhedron, Int. J. Numer. Methods Eng. 58 (2003) 1061–1089.
[95] Q. Du, D. Wang, Constrained boundary recovery for three dimentional delaunay triangulations, Int. J. Numer. Methods Eng. 61 (2004) 1471–1500.
[96] Y. Zheng, R.W. Lewis, D.T. Gethin, Three-dimensional unstructured mesh generation. Part 1: Fundamental aspects of triangulation and point creation, Comput. Methods Appl. Mech. Eng. 134 (1996) 249–268.
[97] V.N. Parthasarathy, C.M. Graichen, A.F. Hathaway, A comparison of tetrahedron quality measures, Finite Elem. Anal. Des. 15 (1993) 255–261.
[98] A. Liu, B. Joe, Relationship between tetrahedron shape measures, BIT 34 (1994) 268–287.
[99] R.W. Lewis, Y. Zheng, D.T. Gethin, Three-dimensional unstructured mesh generation. Part 3: Volume meshes, Comput. Methods Appl. Mech. Eng. 134 (1996) 285–310.
[100] D.A. Field, Qualitative measures for initial meshes, Int. J. Numer. Methods Eng. 47 (2000) 887–906.
[101] J.R. Shewchuk, What is a good linear element? interpolation, conditioning, anisotropy, and quality measures, in: Proceedings of the Eleventh International Meshing Roundtable, Ithaca, NY, 2002, pp. 115–126.

[102] L.A. Freitag, C. Olliver-Gooch, Tetrahedral mesh improvement using swapping and smoothing, Int. J. Numer. Methods Eng. 40 (1997) 3979–4002.
[103] L.A. Freitag, P.M. Knupp, Tetrahedral mesh improvement via optimization of element condition number, Int. J. Numer. Methods Eng. 53 (2002) 1377–1391.
[104] J.M. Escobar, E. Rodríguez, R. Montenegro, G. Montero, J.M. González-Yuste, Simultaneous untangling and smoothing of tetrahedral meshes, Comput. Methods Appl. Mech. Eng. 192 (2003) 2772–2787.
[105] C.L. Bottasso, Anisotropic mesh adaption by metric-driven optimization, Int. J. Numer. Methods Eng. 60 (2004) 597–639.
[106] B.M. Klingner, J.R. Shewchuk, Aggressive tetrahedral mesh improvement, in: Proceedings of the 16th International Meshing Roundtable, Seattle, Washington, 2007, pp. 3–23.
[107] L. Chen, M. Holst, Efficient mesh optimization schemes based on optimal delaunay triangulations, Comput. Methods Appl. Mech. Eng. 200 (2011) 967–984.
[108] D.L. Marcum, Adaptive unstructured grid generation for viscous flow applications, AIAA J. 34 (1996) 2440.
[109] D.J. Mavriplis, An advancing front Delaunay triangulation algorithm designed for robustness, J. Comput. Phys. 117 (1995) 90–101.
[110] H. Borouchaki, P.L. George, F. Hecht, P. Laug, E. Saltel, Delaunay mesh generation governed by metric specifications. Part I: Algorithms, Finite Elem. Anal. Des. 25 (1997) 61–83.
[111] H. Borouchaki, P.L. George, B. Mohammadi, Delaunay mesh generation governed by metric specifications. Part II: Applications, Finite Elem. Anal. Des. 25 (1997) 85–109.
[112] Q. Du, D. Wang, Tetrahedral mesh generation and optimization based on centroidal voronoi tessellations, Int. J. Numer. Methods Eng. 56 (2003) 1355–1373.
[113] H. Si, Adaptive tetrahedral mesh generation by constrained delaunay refinement, Int. J. Numer. Methods Eng. 75 (2008) 856–880.
[114] S. Gosselin, C.Ollivier-Gooch, Tetrahedral mesh generation using delaunay refinement with non-standard quality measures, Int. J. Numer. Methods Eng. 87 (2011) 795–820.

CHAPTER

# Computer Procedures for Finite Element Analysis

# 18

## 18.1 Introduction

A companion program to this book is available which can carry out analyses for most of the theory presented in previous chapters. In particular the computer program discussed here may be used to solve any one-, two-, or three-dimensional linear steady-state or transient problem. The program also has capabilities to perform nonlinear analysis for the type of problems discussed in Ref. [1].

Source listings and a user manual may be obtained at no charge from the website (http://www.ce.berkeley.edu/feap/feappv). The program is written mostly in Fortran with some routines in C (see the website http://www.ce.berkeley.edu/feap for more information on using C for user modules). Any errors reported by readers will be corrected so that up-to-date versions are available.

The version available for download is called *FEAPpv* which is an acronym for *Finite Element Analysis Program—personal version*. It is intended mainly for use in learning finite element programming methodologies and in solving small to moderate size problems on single processor computers. A simple management scheme is employed to permit efficient use of main memory with limited need to read and write information to disk.

Finite element programs can be separated into three basic parts:

1. Data input module and pre-processor
2. Solution module
3. Results module and post-processor

## 18.2 Pre-processing module: Mesh creation

*FEAPpv* is mainly a solution module but provides simple data input and pre-processor capabilities which permit generation of meshes using the multiblock schemes of Zienkiewicz and Phillips [2] and Gordon and Hall [3]. Alternatively the data may be input from neutral files written by other pre-processing systems (e.g., GiD [4] or DistMesh [5]).

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00018-6

Data input for the program consists of specification (or generation) of (1) the coordinates for each node; (2) the element form and the nodal connection list for each element; (3) boundary conditions and loads to be applied; and (4) material property data. The user manual describes the format for specifying the data to be used by *FEAPpv*.

### 18.2.1 Element library

As part of the input data it is necessary to describe the element formulation to be used in forming the "stiffness" matrix and "load" vector of each problem. This may be provided either by user written modules (see below) or using the element library provided with the program.

Currently, the element library in *FEAPpv* includes

1. *Solid elements for 2-D linear elasticity:* Forms are provided for the irreducible formulation described in Chapter 7; the three-field mixed form described in Section 10.4; and the enhanced strain form described in Section 9.5.3. The elements permit consideration of elastic models which are isotropic or orthotropic as described in Chapter 2.
   - **(a)** For the irreducible form the element shape may range from a three-node triangle to a nine-node Lagrangian quadrilateral.
   - **(b)** For the three-field mixed form the element shape may be a four-node, eight-node, or nine-node quadrilateral form.
   - **(c)** For the enhanced strain model the element is restricted to a four-node quadrilateral form.
2. *Solid elements for 3-D linear elasticity:* Only the irreducible form for a four-node tetrahedron or an eight-node brick may be used. The eight-node brick may be degenerated into other forms by giving the same node number to nodes used to perform the degenerate shape (see Section 6.9). The elastic material model may be isotropic or orthotropic as described in Chapter 2.
3. *Frame (rod) elements for 2- and 3-D elasticity:* Conventional structural elements are provided to perform analysis of elastic two- and three-dimensional frame structures. While these forms have not been discussed in this text, except as suggested problems for solution, they are useful for general analysis. The theory is contained in standard references for structural analysis.
4. *Truss elements for 2- and 3-D elasticity:* Similar to frame elements, the *FEAPpv* system includes conventional truss elements which may be used to analyze plane and space truss structures.
5. *Plate element for linear elasticity:* A plate bending element for use in the analysis of thin and thick plates which include the primary effects of transverse shear is provided. The element form may be either a three-node triangle or a four-node quadrilateral. The element uses linked interpolation as described in Chapter 13 and in Refs. [6,7].
6. *Shell element for 3-D with linear elasticity:* A four-node quadrilateral element form for use in modeling general shell forms is provided. The element includes

membrane and bending effects only and, thus, may be used only for analysis of "thin" shells. The theory for the element is given in Ref. [8]. The element form should be a four-node quadrilateral.

7. *Membrane element for linear elasticity.* A general elastic membrane form is provided which is the same as the shell element but without the bending terms. The element form should be a four-node quadrilateral.
8. *Thermal elements for 2- and 3-D Fourier heat conduction.* The theory described in Chapters 2 and 5 for transient heat conduction is provided in elements which solve 2- and 3-D problems. The Fourier model may be isotropic or orthotropic.
9. *User developed elements.* Users may develop and add element modules for any problem which can be formed by the finite element approach described in this book. Details for writing modules will be found in the *Programmers Manual* available at the website.

## 18.3 Solution module

The main part of *FEAPpv* is a solution module which permits users to analyze a large range of problems formulated by the finite element method. Specific solution methods are prepared by the user using a *command language*, which is a sequence of statements which describe each algorithm. The current version of *FEAPpv* permits both "batch" and "interactive" problem solution. The commands provided permit specification of problems with either symmetric or unsymmetric "stiffness" matrices, selection of direct or iterative solution of the linear algebraic equation system, selection of different transient solution algorithms, and output of solution results in either a text or graphical format. Commands which permit solution of a symmetric generalized linear eigenproblem (see Chapter 12) using a "sub-space" method [9,10] are also available as well as a feature to compute the eigenvalues and vectors for a symmetric element stiffness matrix.

While the main thrust of this book is the solution of linear problems, *FEAPpv* is capable of solving both linear and nonlinear problems. The use of special "loop" commands permits the construction of algorithms which require iteration or time-stepping. In addition, features to solve problems in which load-following is needed are provided in the form of "arc-length" type methods [11–13]. The solution of problems for which it is not possible to deduce an accurate "stiffness" matrix may be attempted using a quasi-Newton method based on the BFGS method [14,15]. The user manual available at the website provides examples for several algorithms as well as a list of available commands.

## 18.4 Post-processor module

As noted above the *FEAPpv* system contains capabilities to report results as text data written to an output file or in a graphical form which may displayed on the screen or written to files for processing by other systems. Written files are written in

*PostScript* format (in an encapsulated form which may be used by many document systems—e.g., TeX or LaTeX).

The general features of graphical post-processing can display mesh and contours on two-dimensional objects, as well as perspective views on three-dimensional objects. More complex forms require an interface to a separate pre/post-processing system (e.g., GiD [4]). The graphical capabilities included in *FEAPpv* include display of the mesh including node and element numbers, boundary conditions, and nodal loads. Contour plots for each degree of freedom of the nodal parameters may be displayed as well as contours of element values such as stress or flux components. The user manual provides a list of the commands for constructing graphical outputs.

The available version for graphics is limited to X-window applications and compilers compatible with the current Intel Fortran 95 compiler for Windows-based systems (use a search engine to find details on the compiler).

## 18.5 User modules

A key ingredient of the *FEAPpv* system is the ability of a user to add their own modules to extend the capabilities of the program to other classes of problems, material models, or solution strategies. Some user-developed modules are available at the *FEAPpv* website given above and include element modules for other problem forms, an interface to other linear equation solvers, etc. Experienced programmers should be able to easily adapt these routines to include additional features.

Programming additions to the system may be performed following the descriptions in the *Programmer Manual* available at the website.

## References

[1] O.C. Zienkiewicz, R.L. Taylor, D. Fox, The Finite Element Method for Solid and Structural Mechanics, seventh ed., Elsevier, Oxford, 2013.

[2] O.C. Zienkiewicz, D.V. Phillips, An automatic mesh generation scheme for plane and curved surfaces by isoparametric coordinates, Int. J. Numer. Methods Eng. 3 (1971) 519–528.

[3] W.J. Gordon, C.A. Hall, Transfinite element methods—blending-function interpolation over arbitrary curved element domains, Numer. Math. 21 (1973) 109–129.

[4] GiD—The Personal Pre/Postprocesor, 2004. <www.gidhome.com>.

[5] Per Olaf Persson, DistMesh—a simple mesh generator in MATLAB, Current year. <persson.berkeley.edu/distmesh/>.

[6] F. Auricchio, R.L. Taylor, A shear deformable plate element with an exact thin limit, Comput. Methods Appl. Mech. Eng. 118 (1994) 393–412.

[7] F. Auricchio, R.L. Taylor, A triangular thick plate finite element with an exact thin limit, Finite Elem. Anal. Des. 19 (1995) 57–68.

[8] R.L. Taylor, Finite element analysis of linear shell problems, in: J.R. Whiteman (Ed.), The Mathematics of Finite Elements and Applications VI, Academic Press, London, 1988, pp. 191–203.
[9] J.H. Wilkinson, C. Reinsch, Linear Algebra, Handbook for Automatic Computation, vol. II, Springer-Verlag, Berlin, 1971.
[10] K.-J. Bathe, Finite Element Procedures, Prentice Hall, Englewood Cliffs, NJ, 1996.
[11] E. Riks, An incremental approach to the solution of snapping and buckling problems, Int. J. Solids Struct. 15 (1979) 529–551.
[12] K. Schweizerhof, Nitchlineare Berechnung von Tragwerken unter verformungsabhangiger belastung mit finiten Elementen, Doctoral dissertation, U. Stuttgart, Stuttgart, Germany, 1982.
[13] J.C. Simo, P. Wriggers, K.H. Schweizerhof, R.L. Taylor, Finite deformation post-buckling analysis involving inelasticity and contact constraints, Int. J. Numer. Methods Eng. 23 (1986) 779–800.
[14] J.E. Dennis, J. More, Quasi-Newton methods—motivation and theory, SIAM Rev. 19 (1977) 46–89.
[15] H. Matthies, G. Strang, The solution of nonlinear finite element equations, Int. J. Numer. Methods Eng. 14 (1979) 1613–1626.

This page is intentionally left blank

APPENDIX

# Matrix Algebra

It will be found that in order to follow the present text and carry out the necessary computation only a limited knowledge for a few basic definitions of matrix algebra is required.

## Definition of a matrix

The linear relationship between a set of variables $x$ and $b$

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + a_{14}x_4 &= b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + a_{24}x_4 &= b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + a_{34}x_4 &= b_3 \end{aligned} \tag{A.1}$$

can be written, in a shorthand way, as

$$[A]\{x\} = \{b\} \tag{A.2}$$

or

$$\mathbf{A}\,\mathbf{x} = \mathbf{b} \tag{A.3}$$

where

$$\mathbf{A} \equiv [A] = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \end{bmatrix}$$

$$\mathbf{x} \equiv \{x\} = \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{Bmatrix} \quad \text{and} \quad \mathbf{b} \equiv \{b\} = \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \end{Bmatrix} \tag{A.4}$$

The above notation contains within it both the definition of a matrix and of the process of multiplication of two matrices. Matrices are *defined* as "arrays of numbers" of the type shown in Eq. (A.4). The particular form listing a single column of numbers is often referred to as a vector or column matrix, whereas a matrix with multiple columns and rows is called a rectangular matrix. The multiplication of a matrix by a column vector is *defined* by the equivalence of the left and right sides of Eqs. (A.1) and (A.2).

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00027-7

The use of bold characters to define both vectors and matrices will be followed throughout the text—generally lowercase letters denoting vectors and capital letters matrices.

If another relationship, using the same $a$ constants, but a different set of $x$ and $b$, exists and is written as

$$\begin{aligned} a_{11}x'_1 + a_{12}x'_2 + a_{13}x'_3 + a_{14}x'_4 &= b'_1 \\ a_{21}x'_1 + a_{22}x'_2 + a_{23}x'_3 + a_{24}x'_4 &= b'_2 \\ a_{31}x'_1 + a_{32}x'_2 + a_{33}x'_3 + a_{34}x'_4 &= b'_3 \end{aligned} \tag{A.5}$$

then we could write

$$[A]\,[X] = [B] \qquad \text{or} \qquad \mathbf{A}\,\mathbf{X} = \mathbf{B} \tag{A.6}$$

in which

$$\mathbf{X} \equiv [X] = [\mathbf{x},\ \mathbf{x}'] = \begin{bmatrix} x_1, & x'_1 \\ x_2, & x'_2 \\ x_3, & x'_3 \\ x_4, & x'_4 \end{bmatrix} \qquad \mathbf{B} \equiv [B] = [\mathbf{b},\ \mathbf{b}'] = \begin{bmatrix} b_1, & b'_1 \\ b_2, & b'_2 \\ b_3, & b'_3 \end{bmatrix} \tag{A.7}$$

implying both the statements (A.1) and (A.5) arranged simultaneously as

$$\begin{bmatrix} a_{11}x_1 + \cdots, & a_{11}x'_1 + \cdots \\ a_{21}x_1 + \cdots, & a_{21}x'_1 + \cdots \\ a_{31}x_1 + \cdots, & a_{31}x'_1 + \cdots \end{bmatrix} = \mathbf{B} \equiv [B] = \begin{bmatrix} b_1, & b'_1 \\ b_2, & b'_2 \\ b_3, & b'_3 \end{bmatrix} \tag{A.8}$$

It is seen, incidentally, that matrices can be equal only if each of the individual terms is equal.

The multiplication of full matrices is defined above, and it is obvious that it has a meaning only if the number of columns in **A** is equal to the number of rows in **X** for a relation of the type (A.6). One property that distinguishes matrix multiplication is that, in general,

$$\mathbf{A}\,\mathbf{X} \neq \mathbf{X}\,\mathbf{A}$$

i.e., multiplication of matrices does not commute.

## Matrix addition or subtraction

If relations of the form from (A.1) and (A.5) are added then we have

$$\begin{aligned} a_{11}(x_1 + x'_1) + a_{12}(x_2 + x'_2) + a_{13}(x_3 + x'_3) + a_{14}(x_4 + x'_4) &= b_1 + b'_1 \\ a_{21}(x_1 + x'_1) + a_{22}(x_2 + x'_2) + a_{23}(x_3 + x'_3) + a_{24}(x_4 + x'_4) &= b_2 + b'_2 \\ a_{31}(x_1 + x'_1) + a_{32}(x_2 + x'_2) + a_{33}(x_3 + x'_3) + a_{34}(x_4 + x'_4) &= b_3 + b'_3 \end{aligned} \tag{A.9}$$

which will also follow from

$$\mathbf{A}\,\mathbf{x} + \mathbf{A}\,\mathbf{x}' = \mathbf{b} + \mathbf{b}'$$

if we define the addition of matrices by a simple addition of the individual terms of the array. Clearly this can be done only if the size of the matrices is identical, i.e., for example,

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{bmatrix} = \begin{bmatrix} a_{11}+b_{11} & a_{12}+b_{12} \\ a_{21}+b_{21} & a_{22}+b_{22} \\ a_{31}+b_{31} & a_{32}+b_{32} \end{bmatrix}$$

or

$$\mathbf{A} + \mathbf{B} = \mathbf{C} \tag{A.10}$$

implies that every term of **C** is equal to the sum of the appropriate terms of **A** and **B**. Subtraction obviously follows similar rules.

## Transpose of a matrix

This is simply a definition for reordering the terms in an array in the manner

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}^T = \begin{bmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \\ a_{13} & a_{23} \end{bmatrix} \tag{A.11}$$

and will be indicated by the symbol $T$ as shown.

Its use is not immediately obvious but will be indicated later and can be treated here as a simple prescribed operation.

## Inverse of a matrix

If in the relationship (A.3) the matrix **A** is "square." i.e., it represents the coefficients of simultaneous equations of type (A.1) equal in number to the number of unknowns **x**, then in general it is possible to solve for the unknowns in terms of the known coefficients **b**. This solution can be written as

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b} \tag{A.12}$$

in which the matrix $\mathbf{A}^{-1}$ is known as the "inverse" of the square matrix **A**. Clearly $\mathbf{A}^{-1}$ is also square and of the same size as **A**.

We could obtain (A.12) by multiplying both sides of (A.3) by $\mathbf{A}^{-1}$ and hence

$$\mathbf{A}^{-1}\mathbf{A} = \mathbf{I} = \mathbf{A}\,\mathbf{A}^{-1} \tag{A.13}$$

where **I** is an "identity" matrix having zero on all off-diagonal positions and unity on each of the diagonal positions.

If the equations are "singular" and have no solution then clearly an inverse does not exist.

## A sum of products

In problems of mechanics we often encounter a number of quantities such as force that can be listed as a matrix "vector":

$$\mathbf{f} = \begin{Bmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{Bmatrix} \tag{A.14}$$

These in turn, are often associated with the same number of displacements given by another vector, say,

$$\mathbf{u} = \begin{Bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{Bmatrix} \tag{A.15}$$

It is known that the work is represented as a sum of products of force and displacement

$$W = \sum_{k=1}^{n} f_k \, u_k$$

Clearly the transpose becomes useful here as we can write, by the rule of matrix multiplication,

$$W = \begin{bmatrix} f_1 & f_2 \dots f_n \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{Bmatrix} = \mathbf{f}^T \mathbf{u} = \mathbf{u}^T \mathbf{f} \tag{A.16}$$

This fact is frequently used in this book.

## Transpose of a product

An operation that sometimes occurs is that of taking the transpose of a matrix product. It can be left to the reader to prove from previous definitions that

$$(\mathbf{A}\,\mathbf{B})^T = \mathbf{B}^T \mathbf{A}^T \tag{A.17}$$

## Symmetric matrices

In finite element problems symmetric matrices are often encountered. If a term of a matrix $\mathbf{A}$ is defined as $a_{ij}$, then for a symmetric matrix

$$a_{ij} = a_{ji} \qquad \text{or} \qquad \mathbf{A} = \mathbf{A}^T$$

A symmetric matrix must be square. It can be shown that the inverse of a symmetric matrix is also symmetric:

$$\mathbf{A}^{-1} = (\mathbf{A}^{-1})^T \equiv \mathbf{A}^{-T}$$

## Partitioning

It is easy to verify that a matrix product $\mathbf{A}\,\mathbf{B}$ in which for example

$$\mathbf{A} = \left[\begin{array}{ccc|cc} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ \hline a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \end{array}\right]$$

$$\mathbf{B} = \left[\begin{array}{cc} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \\ \hline b_{41} & b_{42} \\ b_{51} & b_{52} \end{array}\right]$$

could be obtained by dividing each matrix into submatrices, indicated by the lines, and applying the rules of matrix multiplication first to each of such submatrices as if it were a scalar number and then carrying out further multiplication in the usual way. Thus, if we write

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \end{bmatrix}$$

then

$$\mathbf{A}\,\mathbf{B} = \begin{bmatrix} \mathbf{A}_{11}\mathbf{B}_1 + \mathbf{A}_{12}\mathbf{B}_2 \\ \mathbf{A}_{21}\mathbf{B}_1 + \mathbf{A}_{22}\mathbf{B}_2 \end{bmatrix}$$

can be verified as representing the complete product by further multiplication.

The essential feature of partitioning is that the size of subdivisions has to be such as to make the products of the type $\mathbf{A}_{11}\mathbf{B}_1 + \mathbf{A}_{12}\mathbf{B}_2$ meaningful. Operations can be conducted on partitioned matrices, treating each partition as if it were a scalar.

It should be noted that any matrix can be multiplied by a scalar (number):

$$c\,\mathbf{A} = \begin{bmatrix} cA_{11} & cA_{12} & \dots \\ cA_{21} & cA_{22} & \dots \\ \vdots & \vdots & \text{etc.} \end{bmatrix}$$

Here, obviously, the requirements of equality of appropriate rows and columns no longer apply.

If a symmetric matrix is divided into an equal number of submatrices $\mathbf{A}_{ij}$ in rows and columns then

$$\mathbf{A}_{ij} = \mathbf{A}_{ji}^T$$

## The standard eigenvalue problem

An *eigenvalue* of a symmetric matrix $\mathbf{A}$ of size $n \times n$ is a scalar $\lambda_i$ which allows the solution of

$$(\mathbf{A} - \lambda_i \mathbf{I})\,\boldsymbol{\phi}_i = \mathbf{0} \qquad \text{and} \qquad \det|(\mathbf{A} - \lambda_i \mathbf{I})| = 0 \tag{A.18}$$

where $\boldsymbol{\phi}_i$ is called the *eigenvector*.

There are, of course, $n$ such eigenvalues $\lambda_i$ to each of which corresponds an eigenvector $\boldsymbol{\phi}_i$. Such vectors can be shown to be orthonormal and we write

$$\boldsymbol{\phi}_i^T \boldsymbol{\phi}_j = \delta_{ij} = \begin{cases} 1 & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases}$$

The full set of eigenvalues and eigenvectors can be written as

$$\boldsymbol{\Lambda} = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix} \qquad \boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\phi}_1 & \dots & \boldsymbol{\phi}_n \end{bmatrix} \tag{A.19}$$

Using these the matrix $\mathbf{A}$ may be written in its *spectral form* by noting from the orthonormality conditions on the eigenvectors that

$$\boldsymbol{\Phi}^{-1} = \boldsymbol{\Phi}^T$$

Then from

$$\mathbf{A}\,\boldsymbol{\Phi} = \boldsymbol{\Phi}\,\boldsymbol{\Lambda}$$

it follows immediately that

$$\mathbf{A} = \boldsymbol{\Phi}\,\boldsymbol{\Lambda}\,\boldsymbol{\Phi}^T \tag{A.20}$$

The condition number $\kappa$ (which is related to equation solution roundoff) is defined as

$$\kappa = \frac{|\lambda_{\max}|}{|\lambda_{\min}|} \tag{A.21}$$

## The generalized eigenvalue problem

A *generalized eigenvalue problem* for two symmetric matrices $\mathbf{A}$ and $\mathbf{B}$ of size $n \times n$ is given by

$$\left(\mathbf{A} - \lambda_i \mathbf{B}\right)\,\boldsymbol{\phi}_i = \mathbf{0} \qquad \text{and} \qquad \det|(\mathbf{A} - \lambda_i \mathbf{B})| = 0 \quad \text{for } i = 1, \dots, n \tag{A.22}$$

where $\lambda_i$ and $\boldsymbol{\phi}_i$ are the set of $n$ *eigenvalues* and *eigenvectors*.

In the above we generally will assume that $\mathbf{B}$ is positive definite, i.e., it has all positive eigenvalues in a standard eigenvalue problem. In this case the eigenvectors

can be made to be orthonormal with respect to $\mathbf{B}$ and we write

$$\boldsymbol{\phi}_i^T \mathbf{B} \boldsymbol{\phi}_j = \delta_{ij} = \begin{cases} 1 & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases}$$

The full set of eigenvalues and eigenvectors again may be written as

$$\mathbf{A}\,\boldsymbol{\Phi} = \mathbf{B}\,\boldsymbol{\Phi}\,\boldsymbol{\Lambda}$$

and it follows immediately that

$$\boldsymbol{\Lambda} = \boldsymbol{\Phi}^T \mathbf{A}\,\boldsymbol{\Phi} \tag{A.23}$$

Note that for the generalized eigenproblem

$$\boldsymbol{\Phi}^{-1} \neq \boldsymbol{\Phi}^T$$

and thus the simple spectral form for $\mathbf{A}$ given in (A.20) does not follow. Instead one can perform a Cholesky factorization of $\mathbf{B}$ as

$$\mathbf{B} = \mathbf{L}\,\mathbf{L}^T$$

in which $\mathbf{L}$ is a lower triangular matrix (i.e., $L_{ij} = 0$ for $j > i$). Using the orthonormality of $\mathbf{B}$ as

$$\boldsymbol{\Phi}^T \mathbf{L}\,\mathbf{L}^T \boldsymbol{\Phi} = \boldsymbol{\Psi}^T \boldsymbol{\Psi} = \mathbf{I}$$

yields

$$\boldsymbol{\Phi} = \mathbf{L}^{-T} \boldsymbol{\Psi}$$

This permits the generalized problem to be transformed to a standard problem as

$$\mathbf{A}^\star \boldsymbol{\Psi} = \boldsymbol{\Psi}\,\boldsymbol{\Lambda}$$

where

$$\mathbf{A}^\star = \mathbf{L}^{-1} \mathbf{A}\,\mathbf{L}^{-T}$$

Now all the properties of the standard eigenproblem may be used.

This page is intentionally left blank

APPENDIX

# Some Vector Algebra

# B

Some knowledge and understanding of basic vector algebra is needed in dealing with complexities of elements oriented in space as occur in shells, etc. Some of the operations are summarized here.

Vectors (in the geometric sense) can be described by their components along the directions of the $x$, $y$, $z$ axes.

Thus, the vector $\mathbf{v}_{01}$ shown in Fig. B.1 can be written as

$$\mathbf{v}_{01} = x_1\,\mathbf{i} + y_1\,\mathbf{j} + z_1\,\mathbf{k} \tag{B.1}$$

in which $\mathbf{i}$, $\mathbf{j}$, $\mathbf{k}$ are unit vectors in the direction of the $x$, $y$, $z$ axes.

Alternatively, the same vector could be written as

$$\mathbf{v}_{01} = \begin{Bmatrix} x_1 \\ y_1 \\ z_1 \end{Bmatrix} \tag{B.2}$$

(now a "vector" in the column matrix sense) in which the components are distinguished by positions in the column.

## Addition and subtraction

Addition and subtraction is defined by addition and subtraction of components. Thus, for example,

$$\mathbf{v}_{02} - \mathbf{v}_{01} = (x_2 - x_1)\,\mathbf{i} + (y_2 - y_1)\,\mathbf{j} + (z_2 - z_1)\,\mathbf{k} \tag{B.3}$$

The same result is achieved by the definitions of matrix algebra; thus

$$\mathbf{v}_{02} - \mathbf{v}_{01} = \mathbf{v}_{21} = \begin{Bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{Bmatrix} \tag{B.4}$$

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00028-9

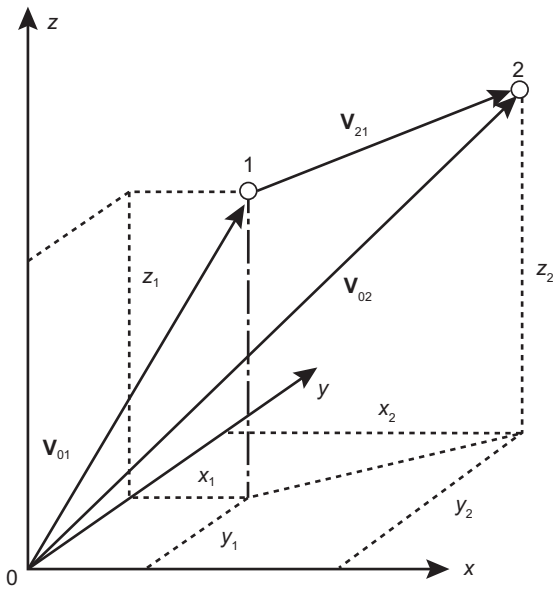



**FIGURE B.1**
Vector addition.

## "Scalar" products

A scalar product of two vectors is *defined* as

$$\mathbf{a} \cdot \mathbf{b} = \mathbf{b} \cdot \mathbf{a} = \sum_{k=1}^{3} a_k \, b_k \tag{B.5}$$

If

$$\begin{aligned} \mathbf{a} &= a_x \, \mathbf{i} + a_y \, \mathbf{j} + a_z \, \mathbf{k} \\ \mathbf{b} &= b_x \, \mathbf{i} + b_y \, \mathbf{j} + b_z \, \mathbf{k} \end{aligned} \tag{B.6}$$

then

$$\mathbf{a} \cdot \mathbf{b} = a_x \, b_x + a_y \, b_y + a_z \, b_z \tag{B.7}$$

Using matrix notation

$$\mathbf{a} = \begin{Bmatrix} a_x \\ a_y \\ a_z \end{Bmatrix} \quad \text{and} \quad \mathbf{b} = \begin{Bmatrix} b_x \\ b_y \\ b_z \end{Bmatrix} \tag{B.8}$$

the scalar product becomes

$$\mathbf{a} \cdot \mathbf{b} = \mathbf{a}^T \mathbf{b} = \mathbf{b}^T \mathbf{a} \tag{B.9}$$

## Length of vector

The length of the vector $\mathbf{v}_{21}$ is given, purely geometrically, as

$$l_{21} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \tag{B.10}$$

or in terms of matrix algebra as

$$l_{21} = \sqrt{\mathbf{v}_{21} \cdot \mathbf{v}_{21}} = \sqrt{\mathbf{v}_{21}^T \mathbf{v}_{21}} \tag{B.11}$$

## Direction cosines

Direction cosines of a vector are simply, from the definition of the projected component of lengths, given as (Fig. B.1)

$$\cos \alpha_x = \frac{x_2 - x_1}{l_{21}} = \frac{\mathbf{v}_{21} \cdot \mathbf{i}}{l_{21}} \tag{B.12}$$

The scalar product may also be written as (Fig. B.2)

$$\mathbf{a} \cdot \mathbf{b} = \mathbf{b} \cdot \mathbf{a} = l_a \, l_b \, \cos \gamma \tag{B.13}$$

where $\gamma$ is the angle between the two vectors $\mathbf{a}$ and $\mathbf{b}$ and $l_a$ and $l_b$ are their lengths, respectively.

## "Vector" or cross-product

Another product of vectors is defined as a vector oriented normal to the plane given by two vectors and equal in magnitude to the product of the length of the two vectors multiplied by the sine of the angle between them. Further, the direction of the normal vector follows the right-hand rule as shown in Fig. B.2 in which

$$\mathbf{a} \times \mathbf{b} = \mathbf{c} \tag{B.14}$$

is shown. Thus, from the right-hand rule, we have

$$\mathbf{a} \times \mathbf{b} = -\mathbf{b} \times \mathbf{a} \tag{B.15}$$

It is worth noting that the magnitude (or length) of $\mathbf{c}$ is equal to the area of the parallelogram shown in Fig. B.2.

Using the definition of Eq. (B.6) and noting that

$$\mathbf{i} \times \mathbf{i} = \mathbf{j} \times \mathbf{j} = \mathbf{k} \times \mathbf{k} = 0$$
$$\mathbf{i} \times \mathbf{j} = \mathbf{k}, \qquad \mathbf{j} \times \mathbf{k} = \mathbf{i}, \qquad \mathbf{k} \times \mathbf{i} = \mathbf{j}$$

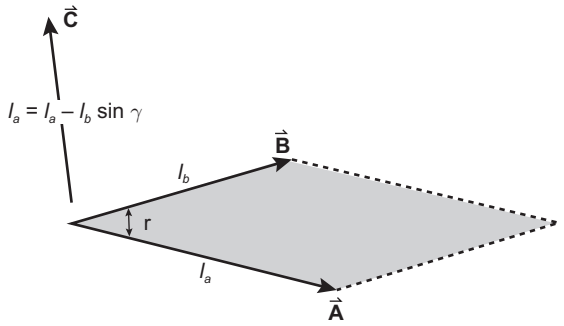



**FIGURE B.2**
Vector multiplication (cross-product).

we have

$$\mathbf{a}\times\mathbf{b} = \det\begin{bmatrix}\mathbf{i} & \mathbf{j} & \mathbf{k}\\ a_x & a_y & a_z\\ b_x & b_y & b_z\end{bmatrix}$$
$$= (a_y b_z - a_z b_y)\,\mathbf{i} + (a_z b_x - a_x b_z)\,\mathbf{j} + (a_x b_y - a_y b_x)\,\mathbf{k}$$

In matrix algebra this does not find a simple counterpart but we can use the above to define the vector $\mathbf{c}$[1]

$$\mathbf{c} = \mathbf{a}\times\mathbf{b} = \begin{Bmatrix} a_y b_z - a_z b_y\\ a_z b_x - a_x b_z\\ a_x b_y - a_y b_x\end{Bmatrix} \tag{B.16}$$

The vector product will be found particularly useful when the problem of erecting a normal direction to a surface is considered.

## Elements of area and volume

If $\xi$ and $\eta$ are some curvilinear coordinates, then the vectors in a two-dimensional plane

$$\mathrm{d}\boldsymbol{\xi} = \begin{Bmatrix}\frac{\partial x}{\partial \xi}\\ \frac{\partial y}{\partial \xi}\end{Bmatrix}\mathrm{d}\xi \qquad \mathrm{d}\boldsymbol{\eta} = \begin{Bmatrix}\frac{\partial x}{\partial \eta}\\ \frac{\partial y}{\partial \eta}\end{Bmatrix}\mathrm{d}\eta \tag{B.17}$$

defined from the relationship between the Cartesian and curvilinear coordinates, are vectors directed tangentially to the $\xi$ and $\eta$ equal constant contours, respectively. As

[1] If we rewrite $\mathbf{a}$ as a skew symmetric matrix

$$\widehat{\mathbf{a}} = \begin{bmatrix} 0 & -a_z & a_y\\ a_z & 0 & -a_x\\ -a_y & a_x & 0\end{bmatrix}$$

then an alternative representation of the vector product in matrix form is $\mathbf{c} = \widehat{\mathbf{a}}\,\mathbf{b}$.

the *length* or the vector resulting from a cross-product of $d\boldsymbol{\xi} \times d\boldsymbol{\eta}$ is equal to the area of the elementary parallelogram we can write

$$\mathrm{d}(area) = \det \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{bmatrix} \mathrm{d}\xi \, \mathrm{d}\eta \tag{B.18}$$

by Eq. (B.16).

Similarly, if we have three curvilinear coordinates $\xi, \eta, \zeta$ in the Cartesian space, the "triple" or box product defines a differential volume

$$\mathrm{d}(vol) = \left(\mathrm{d}\boldsymbol{\xi} \times \mathrm{d}\boldsymbol{\eta}\right) \cdot \mathrm{d}\boldsymbol{\zeta} = \det \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \zeta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial y}{\partial \zeta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} & \frac{\partial z}{\partial \zeta} \end{bmatrix} \mathrm{d}\xi \, \mathrm{d}\eta \, \mathrm{d}\zeta \tag{B.19}$$

This follows simply from the geometry. The bracketed product, by definition, forms a vector whose length is equal to the parallelogram area with sides tangent to two of the coordinate derivatives, similar to that shown in Figure B.2. The second scalar multiplication by a length and the cosine of the angle between that length and the normal to the parallelogram establishes a differential volume element.

The above equations serve in changing the variables in surface and volume integrals.

This page is intentionally left blank

APPENDIX

# Tensor-Indicial Notation in the Approximation of Elasticity Problems

C

## Introduction

The matrix type of notation used in this volume for description of tensor quantities such as stresses and strains is compact and we believe easy to understand. However, in a computer program each quantity often will still have to be identified by appropriate indices and the conciseness of matrix notation does not always carry over to the programming steps. Further, many readers are accustomed to the use of indicial-tensor notation which is a standard tool in the study of solid mechanics. For this reason we summarize here the formulation of the finite element arrays in an indicial form.

Some advantages of such reformulation from the matrix setting become apparent when evaluation of stiffness arrays for isotropic materials is considered. Here some multiplication operations previously necessary become redundant and the element module programs can be written more economically.

When finite deformation problems in solid mechanics have to be considered the use of indicial notation is almost essential to form many of the arrays needed for the residual and tangent terms.

This appendix adds little new to the discretization ideas—it merely repeats in a different language the results already presented.

## Indicial notation: Summation convention

A point $P$ in three-dimensional space may be represented in terms of its Cartesian coordinates $x_i$, $i = 1, 2, 3$. The limits that $i$ can take define its *range*. To define these components we must first establish an oriented orthogonal set of coordinate directions as shown in Fig. C.1. The distance from the origin of the coordinate axes to the point define a position vector $\mathbf{x}$. Along each of the coordinate axes we define the set of unit orthonormal base vectors, $\mathbf{e}_i$, $i = 1, 2, 3$ which have the property

$$\mathbf{e}_i \cdot \mathbf{e}_j = \delta_{ij} = \begin{cases} 1 & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases} \tag{C.1}$$

where $() \cdot ()$ denotes the vector dot product. The components of the position vector are constructed from the vector dot product

$$x_i = \mathbf{e}_i \cdot \mathbf{x}, \quad i = 1, 2, 3 \tag{C.2}$$

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00029-0

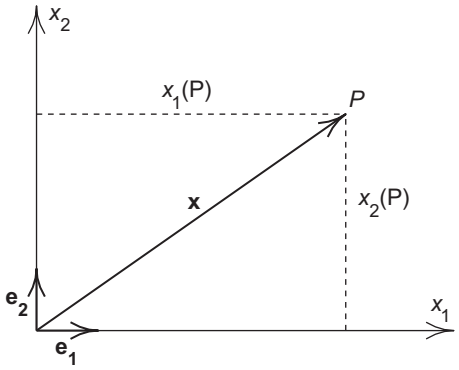



**FIGURE C.1**
Orthogonal axes and a point: Cartesian coordinates.

From this construction it is easy to observe that the vector **x** may be represented as

$$\mathbf{x} = \sum_{i=1}^{3} x_i \, \mathbf{e}_i \tag{C.3}$$

In dealing with vectors, and later tensors, the form **x** is called the *intrinsic* notation of the coordinates, and $x_i \, \mathbf{e}_i$ the *indicial form*.[1] An intrinsic form is a physical entity which is independent of the coordinate system selected, whereas an indicial form depends on a particular coordinate system.

To simplify notation we adopt the common convention that any index which is repeated in any given term implies a summation over the range of the index. Thus, our shorthand notation for Eq. (C.3) is

$$\mathbf{x} = x_i \, \mathbf{e}_i = x_1 \, \mathbf{e}_1 + x_2 \, \mathbf{e}_2 + x_3 \, \mathbf{e}_3 \tag{C.4}$$

For two-dimensional problems, unless otherwise stated, it will be understood that the range of the index is two.

Similarly, we can define the components of the displacement vector **u** as

$$\mathbf{u} = u_i \, \mathbf{e}_i \tag{C.5}$$

Note that the components $(u_1, u_2, u_3)$ replace the components $(u, v, w)$ used throughout most of this volume.

To avoid confusion with nodal quantities to which we previously also attached subscripts we shall simply change their position to a superscript. Thus, $\tilde{u}_2^a$ has the same meaning as $\tilde{v}_a$ used previously, etc.

[1] Often in an indicial form of equations the base vectors are omitted from the final equation.

## Derivatives and tensorial relations

In indicial notation the derivative of any quantity with respect to a coordinate component $x_i$ is written compactly as

$$\frac{\partial}{\partial x_i} \equiv (\,)_{,i} \tag{C.6}$$

Thus we can write the *gradient* of the displacement vector as

$$\frac{\partial u_i}{\partial x_j} \equiv u_{i,j}, \quad i, j = 1, 2, 3 \tag{C.7}$$

In a Cartesian coordinate system the base vectors do not change their magnitude or direction along any coordinate direction. Accordingly their derivatives with respect to any coordinate is zero as indicated in Eq. (C.8):

$$\frac{\partial \mathbf{e}_i}{\partial x_j} = \mathbf{e}_{i,j} = \mathbf{0} \tag{C.8}$$

Thus, in Cartesian coordinates the derivative of the intrinsic displacement $\mathbf{u}$ is given by

$$\mathbf{u}_{,j} = u_{i,j}\mathbf{e}_i + u_i\mathbf{e}_{i,j} = u_{i,j}\mathbf{e}_i \tag{C.9}$$

The collection of all the derivatives defines the *displacement gradient* which we write in intrinsic notation as

$$\nabla\mathbf{u} = u_{i,j}\,\mathbf{e}_i \otimes \mathbf{e}_j \tag{C.10}$$

The symbol $\otimes$ denotes the *tensor product* (or *dyadic product*) between two base vectors and since only two vectors are involved the gradient of the displacement is called *second rank*. The notation used to define a tensor product follows that used in Ref. [1].

Any second rank intrinsic quantity can be split into a symmetric and a skew symmetric (anti-symmetric) parts as

$$\mathbf{A} = \frac{1}{2}\left[\mathbf{A} + \mathbf{A}^{\mathrm{T}}\right] + \frac{1}{2}\left[\mathbf{A} - \mathbf{A}^{\mathrm{T}}\right] = \mathbf{A}^{(s)} + \mathbf{A}^{(a)} \tag{C.11}$$

where $\mathbf{A}$ and its transpose have Cartesian components

$$\mathbf{A} = A_{ij}\,\mathbf{e}_i \otimes \mathbf{e}_j, \quad \mathbf{A}^{\mathrm{T}} = A_{ji}\,\mathbf{e}_i \otimes \mathbf{e}_j \tag{C.12}$$

The symmetric part of the displacement gradient defines the (small) strain[2]

$$\begin{aligned} \boldsymbol{\varepsilon} = \nabla\mathbf{u}^{(s)} &= \frac{1}{2}\left[\nabla\mathbf{u} + (\nabla\mathbf{u})^{\mathrm{T}}\right] \\ &= \frac{1}{2}[u_{i,j} + u_{j,i}]\,\mathbf{e}_i \otimes \mathbf{e}_j \\ &= \varepsilon_{ij}\,\mathbf{e}_i \otimes \mathbf{e}_j = \varepsilon_{ji}\,\mathbf{e}_i \otimes \mathbf{e}_j \end{aligned} \tag{C.13}$$

[2]Note that this definition is slightly different from that occurring in Chapters 2 and 7. Now the shearing strain is given by $\varepsilon_{ij} = 1/2\,\gamma_{ij}$ when $i \neq j$.

and the skew symmetric part gives the (small) rotation

$$\begin{aligned}\boldsymbol{\omega} = \boldsymbol{\nabla}\mathbf{u}^{(a)} &= \frac{1}{2}\left[\boldsymbol{\nabla}\mathbf{u} - (\boldsymbol{\nabla}\mathbf{u})^{\mathrm{T}}\right] \\ &= \frac{1}{2}\left[u_{i,j} - u_{j,i}\right]\mathbf{e}_i \otimes \mathbf{e}_j \\ &= \omega_{ij}\,\mathbf{e}_i \otimes \mathbf{e}_j = -\omega_{ji}\,\mathbf{e}_i \otimes \mathbf{e}_j\end{aligned} \tag{C.14}$$

The strain expression is analogous to Eq. (2.6). The components $\varepsilon_{ij}$ and $\omega_{ij}$ may be represented by a matrix as

$$\varepsilon_{ij} = \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} \\ \varepsilon_{21} & \varepsilon_{22} & \varepsilon_{23} \\ \varepsilon_{31} & \varepsilon_{32} & \varepsilon_{33} \end{bmatrix} = \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} \\ \varepsilon_{12} & \varepsilon_{22} & \varepsilon_{23} \\ \varepsilon_{13} & \varepsilon_{23} & \varepsilon_{33} \end{bmatrix} \tag{C.15}$$

$$\omega_{ij} = \begin{bmatrix} 0 & \omega_{12} & \omega_{13} \\ \omega_{21} & 0 & \omega_{23} \\ \omega_{31} & \omega_{32} & 0 \end{bmatrix} = \begin{bmatrix} 0 & \omega_{12} & \omega_{13} \\ -\omega_{12} & 0 & \omega_{23} \\ -\omega_{13} & -\omega_{23} & 0 \end{bmatrix} \tag{C.16}$$

## Coordinate transformation

Consider now the representation of the intrinsic coordinates in a system which has different orientation than that given in Fig. C.1. We represent the components in the new system by

$$\mathbf{x} = x'_{i'}\,\mathbf{e}'_{i'} \tag{C.17}$$

Using Eq. (C.2) we can relate the components in the prime system to those in the original system as

$$x'_{i'} = \mathbf{e}'_i \cdot \mathbf{x} = \mathbf{e}'_i \cdot \mathbf{e}_j\, x_j = \Lambda_{i'j}\, x_j \tag{C.18}$$

where

$$\Lambda_{i'j} = \mathbf{e}'_{i'} \cdot \mathbf{e}_j = \cos\left(x'_{i'}, x_j\right) \tag{C.19}$$

defines the direction cosines of the coordinate in a manner similar to that of Eq. (2.30).

Equation (C.18) defines how the Cartesian coordinate components transform from one coordinate frame to another. Recall that summation convention implies

$$x'_{i'} = \Lambda_{i'1}\, x_1 + \Lambda_{i'2}\, x_2 + \Lambda_{i'3}\, x_3 \quad i' = 1, 2, 3 \tag{C.20}$$

In Eq. (C.18) $i'$ is called a *free index* whereas $j$ is called a *dummy index* since it may be replaced by any other unique index without changing the meaning of the term (note that the notation used never permits an index to appear more than twice in any term). Summation convention will be employed throughout the remainder of this discussion and the reader should ensure that the concept is fully understood before proceeding. Some examples will be given occasionally to illustrate its use.

Using the notion of the direction cosines, Eq. (C.18) may be used to transform any vector with three components. Thus, transformation of the components of the

displacement vector is given by

$$u'_{i'} = \Lambda_{i'j}\, u_j \quad i', j = 1, 2, 3 \tag{C.21}$$

Indeed we can also use the above to express the transformation for the base vectors since

$$\mathbf{e}'_{i'} = \left(\mathbf{e}'_{i'} \cdot \mathbf{e}_j\right) \mathbf{e}_j = \Lambda_{i'j}\, \mathbf{e}_j \tag{C.22}$$

Similarly, by interchanging the role of the base vectors we obtain

$$\mathbf{e}_j = \left(\mathbf{e}_j \cdot \mathbf{e}'_{i'}\right) \mathbf{e}'_{i'} = \Lambda_{i'j}\, \mathbf{e}'_{i'} \tag{C.23}$$

Which indicates that the *inverse* of the direction cosine coefficient array is the same as its *transpose*.

The strain transformation follows from the intrinsic form written as

$$\boldsymbol{\varepsilon} = \varepsilon'_{i'j'}\, \mathbf{e}_{i'} \otimes \mathbf{e}_{j'} = \varepsilon_{kl}\, \mathbf{e}_k \otimes \mathbf{e}_l \tag{C.24}$$

Substitution of the base vectors from Eq. (C.23) into Eq. (C.24) gives

$$\boldsymbol{\varepsilon} = \Lambda_{i'k}\, \varepsilon_{kl}\, \Lambda_{j'l}\, \mathbf{e}_{i'} \otimes \mathbf{e}_{j'} \tag{C.25}$$

Comparing Eq. (C.25) with Eq. (C.24) the components of the strain transform according to the relation

$$\varepsilon'_{i'j'} = \Lambda_{i'k}\, \varepsilon_{kl}\, \Lambda_{j'l} \tag{C.26}$$

Variables that transform according to Eq. (C.21) are called *first rank Cartesian tensors* whereas quantities that transform according to Eq. (C.26) are called *second rank Cartesian tensors*. The use of indicial notation in the context of Cartesian coordinates will lead naturally to each mechanics variable being defined in terms of a Cartesian tensor of an appropriate rank.

Stress may be written in terms of its components $\sigma_{ij}$ which may be written in a matrix form similar to Eq. (C.15):

$$\sigma_{ij} = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix}, \quad i, j = 1, 2, 3 \tag{C.27}$$

In intrinsic form stress is given by

$$\boldsymbol{\sigma} = \sigma_{ij}\, \mathbf{e}_i \otimes \mathbf{e}_j \tag{C.28}$$

and, using similar logic as used for strain, can be shown to transform as a second rank Cartesian tensor. The symmetry of the components of stress may be established by summing moments (angular momentum balance) about each of the coordinate axes to obtain

$$\sigma_{ij} = \sigma_{ji} \tag{C.29}$$

## Equilibrium and energy

Introducing a body force vector

$$\mathbf{b} = b_i \, \mathbf{e}_i \tag{C.30}$$

we can write the static equilibrium equations (linear momentum balance) for a differential element as

$$\operatorname{div} \boldsymbol{\sigma} + \mathbf{b} \equiv \left(\sigma_{ji,j} + b_i\right) \mathbf{e}_i = \mathbf{0} \tag{C.31}$$

where the repeated index again implies summation over the range of the index, i.e.,

$$\sigma_{ji,j} \equiv \sum_{j=1}^{3} \sigma_{ji,j} = \sigma_{1i,1} + \sigma_{2i,2} + \sigma_{3i,3}$$

Note that the free index $i$ must appear in each term for the equation to be meaningful.

As a further example of the summation convention consider an internal energy term

$$W = \sigma_{ij} \, \varepsilon_{ij} \tag{C.32}$$

This expression implies a double summation; hence summing first on $i$ gives

$$W = \sigma_{1j} \, \varepsilon_{1j} + \sigma_{2j} \, \varepsilon_{2j} + \sigma_{3j} \, \varepsilon_{3j}$$

and then summing on $j$ gives finally

$$\begin{aligned} W = {} & \sigma_{11} \, \varepsilon_{11} + \sigma_{12} \, \varepsilon_{12} + \sigma_{13} \, \varepsilon_{13} + \sigma_{21} \, \varepsilon_{21} + \sigma_{22} \, \varepsilon_{22} \\ & + \sigma_{23} \, \varepsilon_{23} + \sigma_{31} \, \varepsilon_{31} + \sigma_{32} \, \varepsilon_{32} + \sigma_{33} \, \varepsilon_{33} \end{aligned}$$

We may use symmetry conditions on $\sigma_{ij}$ and $\varepsilon_{ij}$ to reduce the nine terms to six terms. Accordingly,

$$\begin{aligned} W &= \sigma_{11} \, \varepsilon_{11} + \sigma_{22} \, \varepsilon_{22} + \sigma_{33} \, \varepsilon_{33} + 2 \left(\sigma_{12} \, \varepsilon_{12} + \sigma_{23} \, \varepsilon_{23} + \sigma_{31} \, \varepsilon_{31}\right) \\ &= \sigma_{11} \, \varepsilon_{11} + \sigma_{22} \, \varepsilon_{22} + \sigma_{33} \, \varepsilon_{33} + \sigma_{12} \, \gamma_{12} + \sigma_{23} \, \gamma_{23} + \sigma_{31} \, \gamma_{31} \end{aligned} \tag{C.33}$$

Indeed, the change to Voigt notation is fully established. Following a similar expansion we can also show the result

$$\sigma_{ij} \, \omega_{ij} \equiv 0 \tag{C.34}$$

## Elastic constitutive equations

For an elastic material the most general linear relationship we can write for components of the stress–strain characterization is

$$\sigma_{ij} = \mathrm{D}_{ijkl} \left(\varepsilon_{kl} - \varepsilon^0_{kl}\right) + \sigma^0_{ij} \tag{C.35}$$

Equation (C.35) is the equivalent of Eq. (2.46a) but now written in index notation. We note that the elastic moduli which appear in Eq. (C.35) are components of the fourth rank tensor

$$\mathbf{D} = \mathrm{D}_{ijkl}\, \mathbf{e}_i \otimes \mathbf{e}_j \otimes \mathbf{e}_k \otimes \mathbf{e}_l \tag{C.36}$$

The elastic moduli possess the symmetry conditions

$$\mathrm{D}_{ijkl} = \mathrm{D}_{jikl} = \mathrm{D}_{ijlk} = \mathrm{D}_{klij} \tag{C.37}$$

with the latter arising from the existence of an internal energy density in the form [2]

$$W(\boldsymbol{\varepsilon}) = \frac{1}{2}\, \varepsilon_{ij}\, \mathrm{D}_{ijkl}\, \varepsilon_{kl} + \varepsilon_{ij} \left[\sigma^0_{ij} - \mathrm{D}_{ijkl} \varepsilon^0_{kl}\right] \tag{C.38}$$

This yields the stress from

$$\sigma_{ij} = \frac{\partial W}{\partial \varepsilon_{ij}} \tag{C.39}$$

By writing the constitutive equation with respect to $x'_{i'}$ and using properties of the base vectors, we can deduce the transformation equation for moduli as

$$\mathrm{D}'_{i'j'k'l'} = \Lambda_{i'm}\, \Lambda_{j'n}\, \Lambda_{k'p}\, \Lambda_{l'q}\, \mathrm{D}_{mnpq} \tag{C.40}$$

A common notation for the intrinsic form of Eq. (C.35) is

$$\boldsymbol{\sigma} = \mathbf{D} : \left(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^0\right) + \boldsymbol{\sigma}^0 \tag{C.41}$$

in which : denotes the double summation (contraction) between the elastic moduli and the strains.

The elastic moduli for an isotropic elastic material may be written in indicial form as

$$\mathrm{D}_{ijkl} = \lambda\, \delta_{ij}\, \delta_{kl} + \mu \left(\delta_{ik}\, \delta_{jl} + \delta_{il}\, \delta_{jk}\right) \tag{C.42}$$

where $\lambda$, $\mu$ are the Lamé constants. An isotropic linear elastic material is always characterized by two independent elastic constants. Instead of the Lamé constants we can use Young's modulus, $E$, and Poisson's ratio, $\nu$ to characterize the material. The Lamé constants may be deduced from (see Table 2.1)

$$\mu = \frac{E}{2(1+\nu)} \quad \text{and} \quad \lambda = \frac{\nu E}{(1+\nu)(1-2\nu)} \tag{C.43}$$

## Finite element approximation

If we now introduce the finite element displacement approximation given by Eq. (7.12), using indicial notation we may write for a single element

$$u_i \approx \hat{u}_i = N_a\, \tilde{u}^a_i \quad i = 1, 2, 3;\ a = 1, 2, \ldots, n \tag{C.44}$$

where $n$ is the total number of nodes on an element. The strain approximation in each element is given by the definition of Eq. (C.13) as

$$\hat{\varepsilon}_{ij} = \frac{1}{2}\left[N_{a,j}\,\tilde{u}_i^a + N_{a,i}\,\tilde{u}_j^a\right] \tag{C.45}$$

The internal virtual work for an element is given as

$$\delta U^I = \int_{\Omega_e} \delta\boldsymbol{\varepsilon} : \boldsymbol{\sigma}\,\mathrm{d}\Omega = \int_{\Omega_e} \delta\varepsilon_{ij}\,\sigma_{ij}\,\mathrm{d}\Omega \tag{C.46}$$

Using Eqs. (C.45) and (C.46) and noting symmetries in $\mathrm{D}_{ijkl}$ we may write the internal virtual work for a linear elastic material as

$$\begin{aligned}\delta U^I = {} & \delta\tilde{u}_i^a \int_{\Omega_e} N_{a,j}\,\mathrm{D}_{ijkl}\,N_{b,l}\,\mathrm{d}\Omega\,\tilde{u}_k^b \\ & + \delta\tilde{u}_i^a \int_{\Omega_e} N_{a,j}\left(\sigma_{ij}^0 - \mathrm{D}_{ijkl}\varepsilon_{kl}^0\right)\mathrm{d}\Omega\end{aligned} \tag{C.47}$$

which replaces in indicial notation the matrix form presented in Chapter 7.

In describing a stiffness coefficient two subscripts have been used previously and the submatrix $\mathbf{K}_{ab}$ implied $2 \times 2$ or $3 \times 3$ entries for the $ab$ nodal pair, depending on whether two- or three-dimensional displacement components were involved. Now the scalar components

$$K_{ij}^{ab} \quad i, j = 1, 2, 3; \quad a, b = 1, 2, \ldots, n \tag{C.48}$$

define completely the appropriate stiffness coefficient with $ij$ indicating the relative submatrix position (in this case for a three-dimensional displacement).

Note that for a symmetric matrix we have previously required that

$$\mathbf{K}_{ab} = \mathbf{K}_{ba}^{\mathrm{T}} \tag{C.49}$$

In indicial notation the same symmetry is implied if

$$K_{ij}^{ab} = K_{ji}^{ba} \tag{C.50}$$

The stiffness tensor is now defined from Eq. (C.47) as

$$K_{ik}^{ab} = \int_{\Omega_e} N_{a,j}\,\mathrm{D}_{ijkl}\,N_{b,l}\,\mathrm{d}\Omega \tag{C.51}$$

When the elastic properties are constant over the element we may separate the integration from the material constants by defining

$$W_{ij}^{ab} = \int_{\Omega_e} N_{a,i}\,N_{b,j}\,\mathrm{d}\Omega \tag{C.52}$$

and then perform the summations with the material moduli as

$$K_{ik}^{ab} = W_{jl}^{ab}\, \mathrm{D}_{ijkl} \tag{C.53}$$

In the case of isotropy a particularly simple result is obtained:

$$K_{ik}^{ab} = \lambda\, W_{ik}^{ab} + \mu \left[ W_{ki}^{ab} + \delta_{ik}\, W_{jj}^{ab} \right] \tag{C.54}$$

This allows the construction of the stiffness to be carried out using fewer arithmetic operations as compared with the use of matrix form [3]. An alternative approach directly using the tensor form is given by Planas et al. [4].

Using indicial notation the final equilibrium equations of the system are written as

$$K_{ik}^{ab}\, \tilde{u}_k^b + f_i^a = 0 \quad i,k = 1,2,3;\ a,b = 1,2,\ldots,n \tag{C.55}$$

and in this scalar form every coefficient is simply identified. The reader can, as a simple exercise, complete the derivation of the force terms due to the initial strain $\varepsilon_{ij}^0$, stress $\sigma_{ij}^0$, body force $b_i$, and external traction $\bar{t}_i$.

Indicial notation is at times useful in clarifying individual terms, and this introduction should be helpful as a key to reading some of the current literature.

## Relation between indicial and matrix notation

The matrix form used throughout most of this volume can be deduced from the indicial form by a simple transformation between the indices. The relationship between the indices of the second rank tensors and their corresponding matrix Voigt form can be performed by an inspection of the ordering in the matrix for stress and its representation shown in Eq. (C.27). In the matrix form the stress was given in Chapter 2 as

$$\boldsymbol{\sigma} = \left[ \sigma_{11}\ \sigma_{22}\ \sigma_{33}\ \sigma_{12}\ \sigma_{23}\ \sigma_{31} \right]^{\mathrm{T}} \tag{C.56}$$

This form includes use of symmetry of stress components. The mapping of the indices follows that shown in Table C.1.

Table C.1 may also be used to perform the map of the material moduli by noting that the components in the energy are associated with the index pairs from the stress

**Table C.1** Mapping between Matrix and Tensor Indices for Second Rank Symmetric Tensors

| Form | Index Number | | | | | |
|---|---|---|---|---|---|---|
| **Matrix** | **1** | **2** | **3** | **4** | **5** | **6** |
| Tensor | 11<br>xx | 22<br>yy | 33<br>zz | 12 and 21<br>xy and yx | 23 and 32<br>yz and zy | 31 and 13<br>zx and xz |

and the strain. Accordingly, the moduli transform as

$$\mathrm{D}_{1111} \rightarrow D_{11}, \mathrm{D}_{2233} \rightarrow D_{23}, \mathrm{D}_{1231} \rightarrow D_{46}, \text{etc.} \tag{C.57}$$

The symmetry of the stress and strain is embedded in Table C.1 and existence of an energy function yields symmetry of the modulus matrix, i.e., $D_{ij} = D_{ji}$.

## References

[1] P. Chadwick, Continuum Mechanics, John Wiley & Sons, New York, 1976.
[2] I.S. Sokolnikoff, The Mathematical Theory of Elasticity, second ed., McGraw-Hill, New York, 1956.
[3] A.K. Gupta, B. Mohraz, A method of computing numerically integrated stiffness matrices, Int. J. Numer. Methods Eng. 5 (1972) 83–89.
[4] J. Planas, I. Romero, J.M. Sancho, B free, Comput. Methods Appl. Mech. Eng. 217–220 (2012) 226–235.

APPENDIX

# Solution of Simultaneous Linear Algebraic Equations

D

A finite element problem leads to a large set of simultaneous semi-discrete linear equations whose solution provides the nodal and element parameters in the formulation. For example, in the analysis of linear steady-state problems the direct assembly of the element coefficient matrices and load vectors leads to a set of linear algebraic equations. In this section methods to solve the simultaneous algebraic equations are summarized. We consider both a *direct* method where an *a priori* calculation of the number of numerical operations can be made, and an *indirect or iterative* method where no such estimate can be made but for very large problems considerable efficiency in solution time can occur.

## Direct solution

Consider first the general problem of direct solution of a set of algebraic equations given by

$$\mathbf{K}\tilde{\mathbf{u}} = \mathbf{f} \tag{D.1}$$

where $\mathbf{K}$ is a square coefficient matrix, $\tilde{\mathbf{u}}$ is a vector of unknown parameters, and $\mathbf{f}$ is a vector of known values. The reader can associate these with the quantities described previously: namely, the stiffness matrix, the nodal unknowns, and the specified forces or residuals.

In the discussion to follow it is assumed that the coefficient matrix has properties such that row and/or column interchanges are unnecessary to achieve an accurate solution. This is true in cases where $\mathbf{K}$ is symmetric positive (or negative) definite.[1] Pivoting may or may not be required with unsymmetric, or indefinite, conditions which can occur when the finite element formulation is based on some weighted residual methods. In these cases some checks or modifications may be necessary to ensure that the equations can be solved accurately [1–3].

For the moment consider that the coefficient matrix can be written as the product of a lower triangular matrix with unit diagonals and an upper triangular matrix.

[1]For mixed methods which lead to forms of the type given in Eq. (9.14) with proper ordering of the equations the solution is given in terms of a positive definite part for $\tilde{\mathbf{q}}$ followed by a negative definite part for $\tilde{\boldsymbol{\phi}}$.

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00030-7

Accordingly,

$$\mathbf{K} = \mathbf{L}\mathbf{U} \tag{D.2}$$

where

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ L_{21} & 1 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ L_{n1} & L_{n2} & \cdots & 1 \end{bmatrix} \tag{D.3}$$

and

$$\mathbf{U} = \begin{bmatrix} U_{11} & U_{12} & \cdots & U_{1n} \\ 0 & U_{22} & \cdots & U_{2n} \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & U_{nn} \end{bmatrix} \tag{D.4}$$

This form is called a *triangular decomposition* of $\mathbf{K}$. The solution to the equations can now be obtained by solving the pair of equations

$$\mathbf{L}\mathbf{y} = \mathbf{f} \tag{D.5}$$

and

$$\mathbf{U}\tilde{\mathbf{u}} = \mathbf{y} \tag{D.6}$$

where $\mathbf{y}$ is introduced to facilitate the separation, e.g., see Refs. [1–5] for additional details.

The reader can easily observe that the solution to these equations is trivial. In terms of the individual equations the solution is given by

$$\begin{aligned} y_1 &= f_1 \\ y_i &= f_i - \sum_{j=1}^{i-1} L_{ij} y_j \quad i = 2, 3, \ldots, n \end{aligned} \tag{D.7}$$

and

$$\begin{aligned} \tilde{u}_n &= \frac{y_n}{U_{nn}} \\ \tilde{u}_i &= \frac{1}{U_{ii}} \left( y_i - \sum_{j=i+1}^{n} U_{ij} \tilde{u}_j \right) \quad i = n-1, n-2, \ldots, 1 \end{aligned} \tag{D.8}$$

Equation (D.7) is commonly called *forward elimination* while Eq. (D.8) is called *back substitution*.

The problem remains to construct the triangular decomposition of the coefficient matrix. This step is accomplished using variations on Gaussian elimination. In practice, the operations necessary for the triangular decomposition are performed directly in the coefficient array; however, to make the steps clear the basic steps are shown in Fig. D.1 using separate arrays. The decomposition is performed in the same way as that used in the subprogram `DATRI` contained in the *FEAPpv* program; thus, the

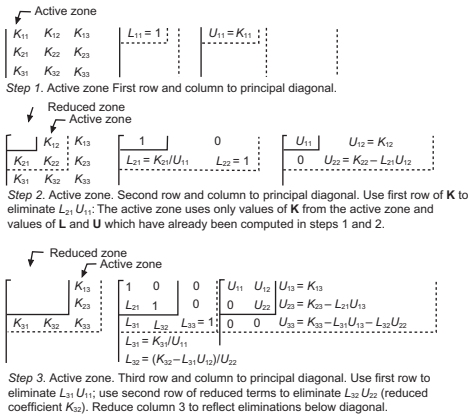



**FIGURE D.1**

Triangular decomposition of **K**.

reader can easily grasp the details of the subprograms included once the steps in Fig. D.1 are mastered. Additional details on this step may be found in Refs. [3–5].

In `DATRI` a Crout form of Gaussian elimination is used to successively reduce the original coefficient array to upper triangular form. The lower portion of the array is used to store $\mathbf{L} - \mathbf{I}$ as shown in Fig. D.1. In this form the unit diagonals for **L** are not stored.

Based on the organization of Fig. D.1 it is convenient to consider the coefficient array to be divided into three parts: part one being the region that is fully reduced; part two the region which is currently being reduced (called the active zone); and part three the region which contains the original unreduced coefficients. These regions are shown in Fig. D.2 where the $j$th column above the diagonal and the $j$th row to the left of the

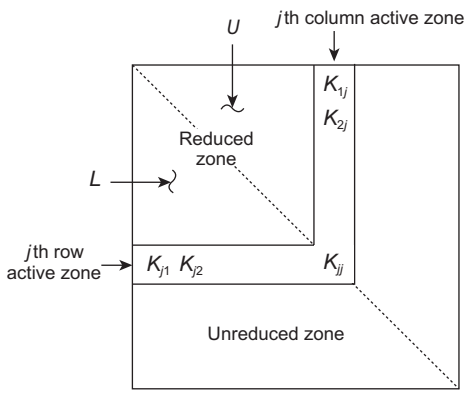



**FIGURE D.2**

Reduced, active, and unreduced parts.

diagonal constitute the active zone. The algorithm for the triangular decomposition of an $n \times n$ square matrix can be deduced from Figs D.1 and D.3 as follows:

$$U_{11} = K_{11}, \quad L_{11} = 1 \tag{D.9}$$

For each active zone $j$ from 2 to $n$,

$$L_{j1} = \frac{K_{j1}}{U_{11}}, \quad U_{1j} = K_{1j} \tag{D.10}$$

$$\begin{aligned} L_{ji} &= \frac{1}{U_{ii}} \left( K_{ji} - \sum_{m=1}^{i-1} L_{jm} U_{mi} \right) \\ U_{ij} &= K_{ij} - \sum_{m=1}^{i-1} L_{im} U_{mj} \quad i = 2, 3, \ldots, j-1 \end{aligned} \tag{D.11}$$

and finally with $L_{jj} = 1$

$$U_{jj} = K_{jj} - \sum_{m=1}^{j-1} L_{jm} U_{mj} \tag{D.12}$$

The ordering of the reduction process and the terms used are shown in Fig. D.3. The results from Fig. D.1 and Eqs. (D.9)–(D.12) can be verified using the matrix given in the example shown in Table D.1.

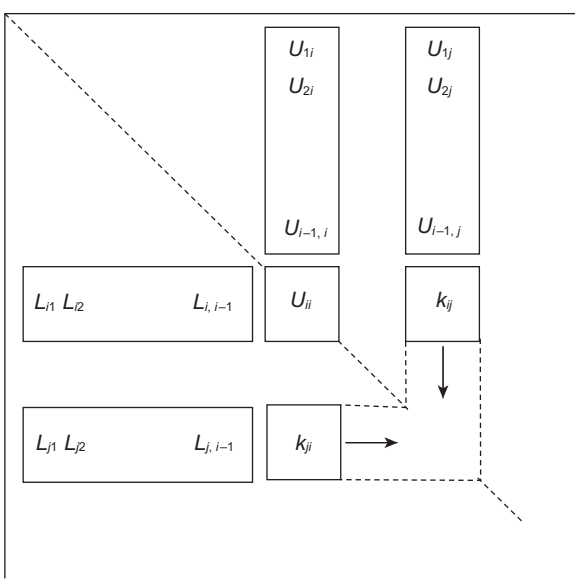



**FIGURE D.3**
Terms used to construct $U_{ij}$ and $L_{ji}$.

**Table D.1** Example: Triangular Decomposition of 3 × 3 Matrix

| K | L | U |
|---|---|---|
| $\begin{bmatrix} 4 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 4 \end{bmatrix}$ | $\begin{bmatrix} 1 & & \\ & & \\ & & \end{bmatrix}$ | $\begin{bmatrix} 4 & & \\ & & \\ & & \end{bmatrix}$ |
| *Step 1.* $L_{11} = 1; U_{11} = 4$ | | |
| $\begin{bmatrix} & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 4 \end{bmatrix}$ | $\begin{bmatrix} 1 & & \\ 0.5 & 1 & \\ & & \end{bmatrix}$ | $\begin{bmatrix} 4 & 2 & \\ & 3 & \\ & & \end{bmatrix}$ |
| *Step 2.* $L_{21} = \frac{2}{4} = 0.5; U_{12} = 2; U_{22} = 1, L_{22} = 4 - 0.5 \times 2 = 3$ | | |
| $\begin{bmatrix} & & 1 \\ & & 2 \\ 1 & 2 & 4 \end{bmatrix}$ | $\begin{bmatrix} 1 & & \\ 0.5 & 1 & \\ 0.25 & 0.5 & 1 \end{bmatrix}$ | $\begin{bmatrix} 4 & 2 & 1 \\ & 3 & 1.5 \\ & & 3 \end{bmatrix}$ |
| *Step 3.* $L_{31} = \frac{1}{4} = 0.25; U_{13} = 1; L_{32} = \frac{2 - 0.25 \times 2}{3} = \frac{1.5}{3} = 0.5$ | | |
| $U_{23} = 2 - 0.5 \times 1 = 1.5;\ L_{33} = 1;\ U_{33} = 4 - 0.25 \times 1 - 0.5 \times 1.5 = 3$ | | |
| $\begin{bmatrix} 1 & & \\ 0.5 & 1 & \\ 0.25 & 0.5 & 1 \end{bmatrix} \begin{bmatrix} 4 & 2 & 1 \\ & 3 & 1.5 \\ & & 3 \end{bmatrix} = \begin{bmatrix} 4 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 4 \end{bmatrix}$ | | |
| *Step 4.* Check | | |

Once the triangular decomposition of the coefficient matrix is computed, several solutions for different right-hand sides **f** can be computed using Eqs. (D.7) and (D.8). This process is often called a *resolution* since it is not necessary to recompute the **L** and **U** arrays. For large size coefficient matrices the triangular decomposition step is very costly while a resolution is relatively cheap; consequently, a resolution capability is necessary in any finite element solution system using a direct method.

The above discussion considered the general case of equation solving (without row or column interchanges). In coefficient matrices resulting from a finite element formulation some special properties are usually present. Often the coefficient matrix is symmetric ($K_{ij} = K_{ji}$) and it is easy to verify in this case that

$$U_{ij} = L_{ji} U_{ii} \tag{D.13}$$

For this problem class it is not necessary to store the entire coefficient matrix. It is sufficient to store only the coefficients above (or below) the principal diagonal and the diagonal coefficients. Equation (D.13) may be used to construct the missing part. This reduces by almost half the required storage for the coefficient array as well as the computational effort to compute the triangular decomposition.

The required storage can be further reduced by storing only those rows and columns which lie within the region of nonzero entries of the coefficient array. Problems formulated by the finite element method and the Galerkin process normally have

a symmetric profile which further simplifies the storage form. Storing the upper and lower parts in separate arrays and the diagonal entries of **U** in a third array is used in DATRI. Figure D.4 shows a typical *profile* matrix and the storage order adopted for the upper array AU, the lower array AL, and the diagonal array AD. An integer array JD is used to locate the start and end of entries in each column. With this scheme it is necessary to store and compute only within the nonzero profile of the equations. This form of storage does not severely penalize the presence of a few large columns/rows and is also an easy form to program a resolution process (e.g., see subprogram DASOL in *FEAPpv* and Ref. [4]).

The routines included in *FEAPpv* are restricted to problems for which the coefficient matrix can fit within the space allocated in the main storage array. In two-dimensional formulations, problems with several thousand degrees of freedom can be solved on today's personal computers. In three-dimensional cases however, problems are restricted to a few thousand equations. To solve larger size problems there are several options. The first is to retain only part of the coefficient matrix in the main array with the rest saved on backing store (e.g., hard disk). This can be quite easily achieved but the size of problem is not greatly increased due to the very large solve times required and the rapid growth in the size of the profile-stored coefficient matrix in three-dimensional problems.

A second option is to use sparse solution schemes. These lead to significant program complexity over the procedure discussed above but can lead to significant savings in storage demands and compute time—especially for problems in three dimensions. Nevertheless, capacity in terms of storage and compute time is again rapidly encountered and alternatives are needed. Examples for sparse solvers can be found in SuperLU [6], LAPACK [7], and UMFPACK [8].

## Iterative solution

One of the main problems in direct solutions is that terms within the coefficient matrix which are zero in a finite element formulation often become nonzero during the triangular decomposition step. While sparse methods are better at limiting this fill than profile methods they still lead to a very large increase in the number of nonzero terms in the factored coefficient matrix. To be more specific consider the case of a three-dimensional linear elastic problem solved using eight-node isoparametric hexahedron elements. In a regular mesh each interior node is associated with 26 other nodes, thus, the equation of such a node has 81 nonzero coefficients—three for each of the 27 associated nodes. On the other hand, for a rectangular block of elements with $n$ nodes on each of the sides the typical column height is approximately proportional to $n^2$ and the number of equations to $n^3$. In Table D.2 we show the size and approximate number of nonzero terms in **K** from a finite element formulation for linear elasticity (i.e., with three degrees of freedom per node). The table also indicates the size growth with column height and storage requirements for a direct solution based on a profile solution method.

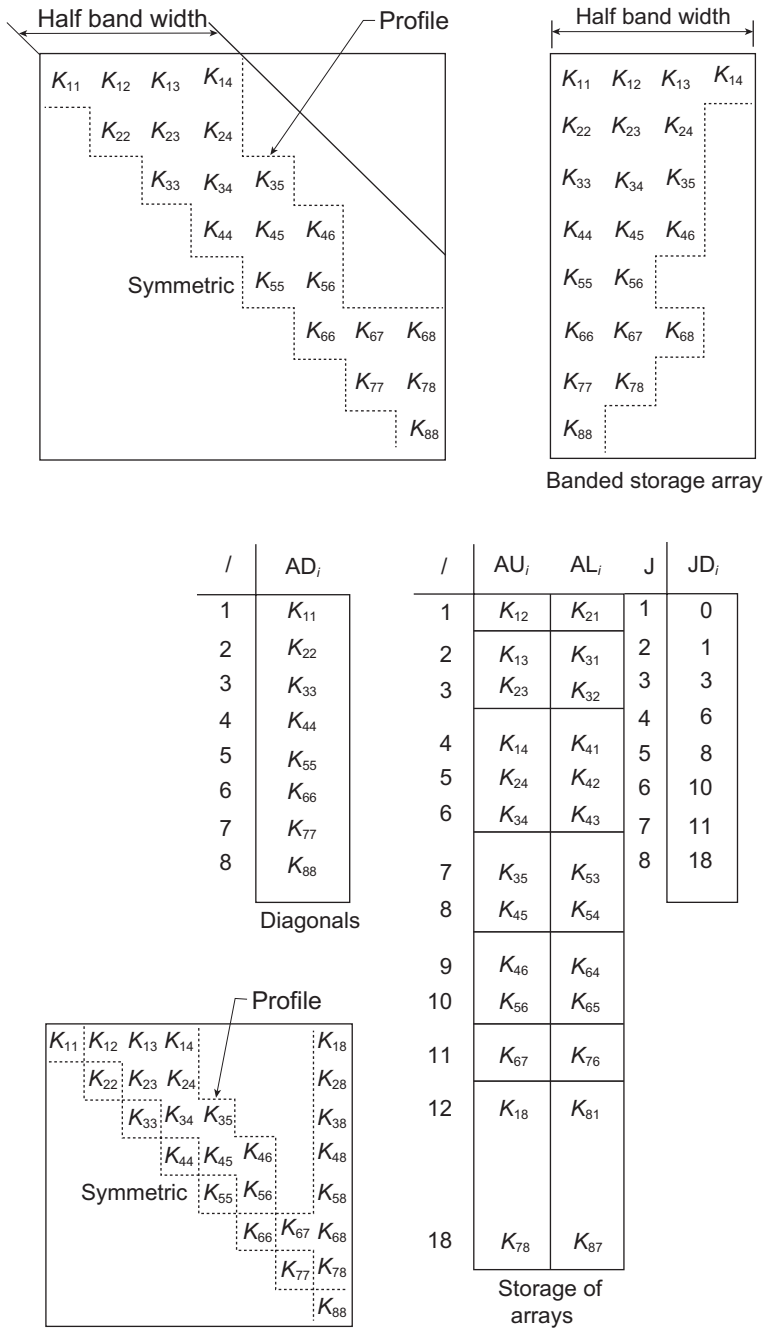



**FIGURE D.4**
Profile storage for coefficient matrix.

From the table it can be observed that the demands for a direct solution are growing very rapidly (storage is approximately proportional to $n^5$) while at the same time the demands for storing the nonzero terms in the stiffness matrix grows proportional to the number of equations (i.e., proportional to $n^3$ for the block).

Iterative solution methods use the terms in the stiffness matrix directly and thus for large problems have the potential to be very efficient for large three-dimensional problems. On the other hand, iterative methods require the resolution of a set of equations until the residual of the linear equations, given by

$$\mathbf{r}^{(i)} = \mathbf{f} - \mathbf{K}\tilde{\mathbf{u}}^{(i)} \tag{D.14}$$

becomes less than a specified tolerance.

In order to be effective the number of iterations $i$ to achieve a solution must be quite small—generally no larger than a few hundred. Otherwise, excessive solution costs will result. At the time of writing this book the subject of iterative solution for general finite element problems remains a topic of intense research. There are some impressive results available for the case where $\mathbf{K}$ is symmetric positive (or negative) definite (e.g., see, for example, Prometheus [9]); however, those for other classes (e.g., unsymmetric or indefinite forms) are generally not as efficient.

For the symmetric positive definite case methods based on a preconditioned conjugate gradient method have been particularly effective [10–12]. The convergence of the method depends on the condition number of the matrix $\mathbf{K}$—the larger the condition number, the slower the convergence (see Ref. [3] for more discussion). The condition number for a finite element problem with a symmetric positive definite stiffness matrix $\mathbf{K}$ is defined as

$$\kappa = \frac{\lambda_n}{\lambda_1} \tag{D.15}$$

where $\lambda_1$ and $\lambda_n$ are the smallest and largest eigenvalue from the solution of the eigenproblem (viz. Appendix A and Chapter 12)

$$\mathbf{K}\boldsymbol{\Phi} = \boldsymbol{\Phi}\boldsymbol{\Lambda} \tag{D.16}$$

in which $\boldsymbol{\Lambda}$ is a diagonal matrix containing the individual eigenvalues $\lambda_i$ and the columns of $\boldsymbol{\Phi}$ are the eigenvectors $\boldsymbol{\phi}_i$ associated with each of the eigenvalues.

**Table D.2** Matrix Size and Storage for Direct Solution

| Side | Number of | Nonzeros in K | Profile Storage Data | |
|---|---|---|---|---|
| Nodes | Equations | Words ($\times 10^{-6}$) | Col. Ht. | Words ($\times 10^{-6}$) |
| 5 | 375 | 0.02 | 90 | 0.03 |
| 10 | 3000 | 0.12 | 330 | 0.99 |
| 20 | 24,000 | 0.96 | 1260 | 30.24 |
| 40 | 192,000 | 7.68 | 4920 | 944.64 |
| 80 | 1,536,000 | 61.44 | 18,440 | 28323.84 |

Usually, the condition number for an elasticity problem modeled by the finite element method is too large to achieve rapid convergence and a *preconditioned conjugate gradient* (PCG) is used [10]. A symmetric form of preconditioned system is written as

$$\mathbf{K}_p\mathbf{z} = \mathbf{P}\mathbf{K}\mathbf{P}^{\mathrm{T}}\mathbf{z} = \mathbf{P}\mathbf{f} \tag{D.17}$$

where

$$\mathbf{P}^{\mathrm{T}}\mathbf{z} = \tilde{\mathbf{u}} \tag{D.18}$$

Now the convergence of the PCG algorithm depends on the condition number of $\mathbf{K}_p$. The problem remains to construct a preconditioner which adequately reduces the condition number. In *FEAPpv* the diagonal of $\mathbf{K}$ is used, however, more efficient schemes also incorporating multigrid methods are discussed in Refs. [11,12].

## References


[1] A. Ralston, A First Course in Numerical Analysis, McGraw-Hill, New York, 1965.

[2] J.H. Wilkinson, C. Reinsch, Linear Algebra, Handbook for Automatic Computation, vol. II, Springer-Verlag, Berlin, 1971.

[3] J. Demmel, Applied Numerical Linear Algebra, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1997.

[4] R.L. Taylor, Solution of linear equations by a profile solver, Eng. Comput. 2 (1985) 344–350.

[5] G. Strang, Linear Algebra and Its Application, Academic Press, New York, 1976.

[6] X.S. Li, J. Demmel, et al., SuperLU, Current. crd-legacy.lbl.gov/xiaoye/SuperLU/.

[7] J. Demmel, J. Dongarra, et al., LAPACK—Linear Algebra PACKage, Current. www.netlib.org/lapack.

[8] T.A. Davis, UMFPACK: unsymmetric multifrontal sparse LU factorization package, Current. <www.cise.ufl.edu/research/sparse/umfpack/>.

[9] M. Adams, PROMETHEUS: parallel multigrid solver library for unstructured finite element problems. <http://www.columbia.edu/~ma2325>.

[10] R.M. Ferencz, Element-by-element preconditioning techniques for large-scale, vectorized finite element analysis in nonlinear solid and structural mechanics, Ph.D. thesis, Department of Mechanical Engineering, Stanford University, Stanford, California, 1989.

[11] M. Adams, A parallel maximal independent set algorithm, in: Proceedings of the Fifth Copper Mountain Conference on Iterative, Methods, 1998.

[12] M. Adams, Parallel multigrid solver algorithms and implementations for 3D unstructured finite element problems, in: Supercomputing'99: High Performance Networking and Computing, vol. <http://www.sc99.org/>, proceedings Portland, Oregon, November 1999.

This page is intentionally left blank

APPENDIX

# Triangle and Tetrahedron Integrals

E

## Triangles

Let a triangle be defined in the $x\ y$ plane by three points $(x_1, y_1)$, $(x_2, y_2)$, $(x_3, y_3)$ with the origin of the coordinates taken at the centroid (or baricenter), i.e.,

$$\frac{x_1 + x_2 + x_3}{3} = \frac{y_1 + y_2 + y_3}{3} = 0$$

Then integrating over the triangle area we obtain

$$\int \mathrm{d}x\,\mathrm{d}y = \frac{1}{2}\begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix} = \Delta = \text{area of triangle}$$

$$\int x\,\mathrm{d}x\,\mathrm{d}y = \int y\,\mathrm{d}x\,\mathrm{d}y = 0$$

$$\int x^2\,\mathrm{d}x\,\mathrm{d}y = \frac{\Delta}{12}\left(x_1^2 + x_2^2 + x_3^2\right)$$

$$\int y^2\,\mathrm{d}x\,\mathrm{d}y = \frac{\Delta}{12}\left(y_1^2 + y_2^2 + y_3^2\right)$$

$$\int x\,y\,\mathrm{d}x\,\mathrm{d}y = \frac{\Delta}{12}\left(x_1\,y_1 + x_2\,y_2 + x_3\,y_3\right)$$

## Tetrahedron

Let a tetrahedron be defined in the $x\ y\ z$ coordinate system by four points $(x_1, y_1, z_1)$, $(x_2, y_2, z_2)$, $(x_3, y_3, z_3)$, $(x_4, y_4, z_4)$ with the origin of the coordinates taken at the centroid, i.e.,

$$\frac{x_1 + x_2 + x_3 + x_4}{4} = \frac{y_1 + y_2 + y_3 + y_4}{4} = \frac{z_1 + z_2 + z_3 + z_4}{4} = 0$$

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00031-9

Then integrating over the tetrahedron volume

$$\int \mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \frac{1}{6}\begin{vmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{vmatrix} = V = \text{tetrahedron volume}$$

Provided the order of numbering the nodes is as indicated on Fig. 6.13a, to give positive volume, then also

$$\int x\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \int y\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \int z\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = 0$$

$$\int x^2\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \frac{V}{20}\left(x_1^2 + x_2^2 + x_3^2 + x_4^2\right)$$

$$\int y^2\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \frac{V}{20}\left(y_1^2 + y_2^2 + y_3^2 + y_4^2\right)$$

$$\int z^2\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \frac{V}{20}\left(z_1^2 + z_2^2 + z_3^2 + z_4^2\right)$$

$$\int xy\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \frac{V}{20}\left(x_1\,y_1 + x_2\,y_2 + x_3\,y_3 + x_4\,y_4\right)$$

$$\int yz\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \frac{V}{20}\left(y_1\,z_1 + y_2\,z_2 + y_3\,z_3 + y_4\,z_4\right)$$

$$\int zx\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \frac{V}{20}\left(z_1\,x_1 + z_2\,x_2 + z_3\,x_3 + z_4\,x_4\right)$$

# Integration by Parts in Two or Three Dimensions (Green's Theorem)

F

Consider the integration by parts of the following two-dimensional expression:

$$\iint_{\Omega} \phi \frac{\partial \psi}{\partial x} \, \mathrm{d}x \, \mathrm{d}y \tag{F.1}$$

Integrating first with respect to $x$ and using the well-known relation for integration by parts in one-dimension

$$\int_{x_L}^{x_R} u \, \mathrm{d}v = -\int_{x_L}^{x_R} v \, \mathrm{d}u + (u\, v)_{x=x_R} - (u\, v)_{x=x_L} \tag{F.2}$$

we have, using the symbols of Fig. F.1

$$\begin{aligned} \iint_{\Omega} \phi \frac{\partial \psi}{\partial x} \, \mathrm{d}x \, \mathrm{d}y = &- \iint_{\Omega} \frac{\partial \phi}{\partial x} \psi \, \mathrm{d}x \, \mathrm{d}y \\ &+ \int_{y_B}^{y_T} \left[ (\phi\, \psi)_{x=x_R} - (\phi\, \psi)_{x=x_L} \right] \mathrm{d}y \end{aligned} \tag{F.3}$$

If now we consider a direct segment of the boundary $\mathrm{d}\Gamma$ on the right-hand boundary, we note that

$$\mathrm{d}y = n_x \, \mathrm{d}\Gamma \tag{F.4}$$

where $n_x$ is the direction cosine between the outward normal and the $x$ direction. Similarly on the left-hand section we have

$$\mathrm{d}y = -n_x \, \mathrm{d}\Gamma \tag{F.5}$$

The final term of Eq. (F.3) can thus be expressed as the integral taken around an anti-clockwise direction of the complete closed boundary:

$$\oint_{\Gamma} \phi \, \psi \, n_x \, \mathrm{d}\Gamma \tag{F.6}$$

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00032-0

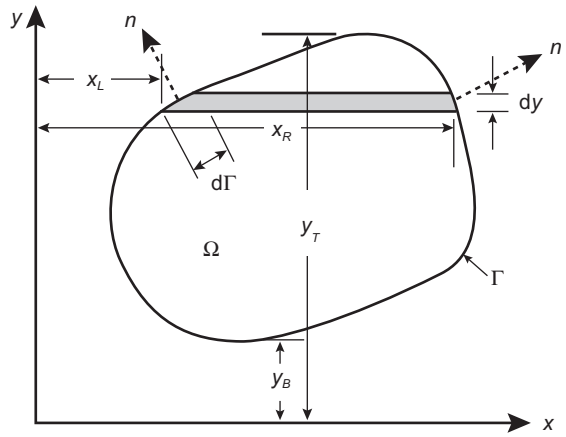



**FIGURE F.1**
Definitions for integrations in two dimensions.

If several closed contours are encountered this integration has to be taken around each such contour. The general expression in all cases is

$$\iint_{\Omega} \phi \frac{\partial \psi}{\partial x} \, dx \, dy = - \iint_{\Omega} \frac{\partial \phi}{\partial x} \psi \, dx \, dy + \oint_{\Gamma} \phi \, \psi \, n_x \, d\Gamma \tag{F.7}$$

Similarly, if differentiation in the $y$ direction arises we can write

$$\iint_{\Omega} \phi \frac{\partial \psi}{\partial y} \, dx \, dy = - \iint_{\Omega} \frac{\partial \phi}{\partial y} \psi \, dx \, dy + \oint_{\Gamma} \phi \, \psi \, n_y \, d\Gamma \tag{F.8}$$

where $n_y$ is the direction cosine between the outward normal and the $y$ axis.

In three dimensions by identical procedure we can write

$$\iiint_{\Omega} \phi \frac{\partial \psi}{\partial y} \, dx \, dy \, dz = - \iiint_{\Omega} \frac{\partial \phi}{\partial y} \psi \, dx \, dy \, dz + \oint_{\Gamma} \phi \, \psi \, n_y \, d\Gamma \tag{F.9}$$

where $d\Gamma$ becomes the element of the surface area and the last integral is taken over the whole surface.

APPENDIX

# Solutions Exact at Nodes

G

The finite element solution of ordinary differential equations may be made exact at the interelement nodes by a proper choice of the *weighting* function in the weak (Galerkin) form. To be more specific, let us consider the set of ordinary differential equations with dependent variable $\mathbf{u}(x)$ given by

$$\mathbf{A}(\mathbf{u}) + \mathbf{f}(x) = \mathbf{0} \tag{G.1}$$

where $\mathbf{f}$ is a vector of specified load functions. The weak form of this set of differential equations is given by

$$\int_{x_L}^{x_R} \mathbf{v}^{\mathrm{T}} \left[\mathbf{A}(\mathbf{u}) + \mathbf{f}\right] \mathrm{d}x = 0 \tag{G.2}$$

The weak form may be integrated by parts to remove all the derivatives from $\mathbf{u}$ and place them on $\mathbf{v}$. The result of this step may be expressed as

$$\int_{x_L}^{x_R} \left[\mathbf{u}^{\mathrm{T}}\mathbf{A}^{\star}(\mathbf{v}) + \mathbf{v}^{\mathrm{T}}\mathbf{f}\right] \mathrm{d}x + \left[\mathbf{B}^{\star}(\mathbf{v})\right]^{\mathrm{T}} \mathbf{B}(\mathbf{u})\Big|_{x_L}^{x_R} = 0 \tag{G.3}$$

where $\mathbf{A}^{\star}(\mathbf{v})$ is the *adjoint differential equation* and $\mathbf{B}^{\star}(\mathbf{v})$ and $\mathbf{B}(\mathbf{u})$ are terms on the boundary resulting from integration by parts.

If we can find the general integral to the homogeneous adjoint differential equation

$$\mathbf{A}^{\star}(\mathbf{v}) = \mathbf{0} \tag{G.4}$$

then the weak form of the problem reduces to

$$\int_{x_L}^{x_R} \mathbf{v}^{\mathrm{T}}\mathbf{f}\, \mathrm{d}x + \left[\mathbf{B}^{\star}(\mathbf{v})\right]^{\mathrm{T}} \mathbf{B}(\mathbf{u})\Big|_{x_L}^{x_R} = 0 \tag{G.5}$$

The first term is merely an expression to generate equivalent forces from the solution to the adjoint equation and the last term is used to construct the residual equation for the problem. If the differential equation is linear these lead to a residual which depends linearly on the values of $\mathbf{u}$ at the ends $x_L$ and $x_R$. If we now let these be the location of the end nodes of a typical element we immediately find an expression to generate a stiffness matrix. Since in this process we have never had to construct an *approximation* for the dependent variables $\mathbf{u}$ it is immediately evident that at the

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00033-2

end points the discrete values of the exact solution must coincide with any admissible approximation we choose—including the exact solution. Thus, we always obtain exact solutions at these points.

If we consider that all values of the forcing function are contained in $f$ (i.e., no point loads at nodes), the terms in $\mathbf{B}(\mathbf{u})$ must be continuous between adjacent elements. At the boundaries the terms in $\mathbf{B}(\mathbf{u})$ include a traction or flux term as well as displacements.

As an example problem, consider the single differential equation

$$\frac{\mathrm{d}^2u}{\mathrm{d}x^2} + P\,\frac{\mathrm{d}u}{\mathrm{d}x} + f = 0 \tag{G.6}$$

with the associated weak form

$$\int_{x_L}^{x_R} v\left[\frac{\mathrm{d}^2u}{\mathrm{d}x^2} + P\,\frac{\mathrm{d}u}{\mathrm{d}x} + f\right]\mathrm{d}x = 0 \tag{G.7}$$

After integration by parts the weak form becomes

$$\int_{x_L}^{x_R}\left[u\left(\frac{\mathrm{d}^2v}{\mathrm{d}x^2} - P\,\frac{\mathrm{d}v}{\mathrm{d}x}\right) + vf\right]\mathrm{d}x + \left[v\left(\frac{\mathrm{d}u}{\mathrm{d}x} + Pu\right) - \frac{\mathrm{d}v}{\mathrm{d}x}\,u\right]_{x_L}^{x_R} = 0 \tag{G.8}$$

The adjoint differential equation is given by

$$\mathbf{A}^\star(v) = \frac{\mathrm{d}^2v}{\mathrm{d}x^2} - P\,\frac{\mathrm{d}v}{\mathrm{d}x} = 0 \tag{G.9}$$

and the boundary terms by

$$\mathbf{B}^\star(v) = \begin{Bmatrix} v \\ -\frac{\mathrm{d}v}{\mathrm{d}x} \end{Bmatrix} \tag{G.10}$$

and

$$\mathbf{B}(u) = \begin{Bmatrix} \frac{\mathrm{d}u}{\mathrm{d}x} + Pu \\ u \end{Bmatrix} \tag{G.11}$$

For the above example two cases may be identified:

1. $P$ zero—where the adjoint differential equation is identical to the homogeneous equation in which case the problem is called *self-adjoint*.
2. $P$ nonzero—where we then have the *non-self-adjoint* problem.

The finite element solution for these two cases is often quite different. In the first case an equivalent variational theorem exists, whereas for the second case no such theorem exists.[1]

[1] An integrating factor often may be introduced to make the weak form generate a self-adjoint problem; however, the approximation problem will remain the same.

In the first case the solution to the adjoint equation is given by

$$v = A\,x + B \tag{G.12}$$

which may be written as conventional linear shape functions in each element as

$$N_L = \frac{x_R - x}{x_R - x_L}, \quad N_R = \frac{x - x_L}{x_R - x_L} \tag{G.13}$$

Thus, for linear shape functions in each element used as the weighting function the interelement nodal displacements for $u$ will always be exact (e.g., see Fig. 3.6) irrespective of the interpolation used for $u$ inside the element.

For the second case the exact solution to the adjoint equation is

$$v = A\,e^{Px} + B = A\,z + B \tag{G.14}$$

This yields the shape functions for the weighting function

$$N_L = \frac{z_R - z}{z_R - z_L}, \quad N_R = \frac{z - z_L}{z_R - z_L} \tag{G.15}$$

which when used in the weak form again yield exact answers at the interelement nodes.

After constructing exact nodal solutions for $u$, exact solutions for the traction or flux at the interelement nodes can also be obtained from the weak form for each element. The above process was first given by Tong for self-adjoint differential equations [1]. Here it is generalized to consider any set of linear ordinary differential equations.

## Reference

[1] P. Tong, Exact solution of certain problems by the finite element method, J. AIAA 7 (1969) 179–180.

This page intentionally left blank

APPENDIX

# Matrix Diagonalization or Lumping

H

Some of the algorithms discussed in this volume for solution of transient problems become more efficient if one of the global matrices can be diagonalized (also called "lumped" by many engineers). For example, the solution of some mixed and transient problems are more efficient if a global matrix to be inverted (or equations solved) is diagonal [viz. Chapter 10, Eq. (10.81), and Chapter 12, Section 12.4.2]. Engineers have often used purely physical concepts of lumping; however, there is clearly a need for devising a systematic and mathematically acceptable procedure for such lumping.

We shall define the matrix to be considered as

$$\mathbf{A} = \int_{\Omega} \mathbf{N}^{\mathrm{T}} \mathbf{c}\, \mathbf{N}\, \mathrm{d}\Omega \tag{H.1}$$

where $\mathbf{c}$ is a matrix with small dimension. Often $\mathbf{c}$ is a diagonal matrix (e.g., in mass or simple least square problems $\mathbf{c}$ is an identity matrix times some scalar). When $\mathbf{A}$ is computed exactly it has full rank and is not diagonal—this is called the *consistent* form of $\mathbf{A}$ since it is computed consistently with the other terms in the finite element model. The diagonalized form is defined with respect to "nodes" or the shape functions, e.g., $\mathbf{N}_a = N_a\mathbf{I}$; hence, the diagonalized "lumped" matrix will be composed of small blocks on the diagonal, each with the maximum dimension of $\mathbf{c}$. Only when $\mathbf{c}$ is diagonal can the matrix $\mathbf{A}$ be completely diagonalized. Four basic lines of argument may be followed in constructing a diagonal form.

The first procedure is to use different shape functions to approximate each term in the finite element discretization. For the $\mathbf{A}$ matrix we use substitute shape functions $\bar{\mathbf{N}}_a$ for the lumping process. No derivatives exist in the definition of $\mathbf{A}$, hence, for this term the shape functions may be piecewise continuous within and between elements and still lead to acceptable approximation. If the shape functions used to define $\mathbf{A}$ are piecewise constants, such that $\bar{\mathbf{N}}_a$ in a certain part of the element surrounding the node $i$ is unity and zero elsewhere, and such parts are not overlapping or disjoint, then clearly the matrix of Eq. (H.1) becomes nodally diagonal as

$$\int_{\Omega} \bar{\mathbf{N}}_a^{\mathrm{T}} \mathbf{c} \bar{\mathbf{N}}_b \, \mathrm{d}\Omega = \begin{cases} \int_{\Omega_a} \mathbf{c}\, \mathrm{d}\Omega & a = b \\ 0 & a \neq b \end{cases} \tag{H.2}$$

**The Finite Element Method: Its Basis and Fundamentals.** http://dx.doi.org/10.1016/B978-1-85617-633-0.00034-4

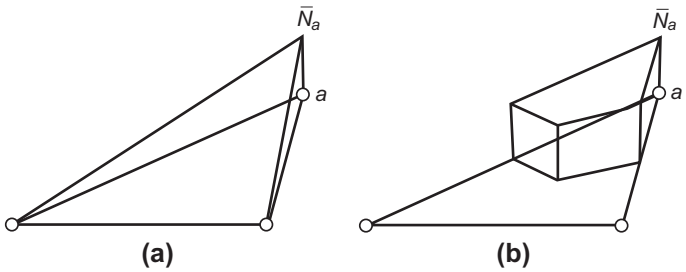



**FIGURE H.1**
(a) Linear and (b) piecewise constant shape functions for a triangle.

Such an approximation with different shape functions is permissible since the usual finite element criteria of integrability and completeness are satisfied. We can verify this using a patch test to show that consistency is still maintained in the approximation. The functions selected need only satisfy the condition

$$\bar{\mathbf{N}}_a = \bar{N}_a \mathbf{I} \quad \text{with} \sum_a \bar{N}_a = 1 \tag{H.3}$$

for all points in the element and this also maintains a partition of unity property in all of $\Omega$. In Fig. H.1 we show the functions $N_a$ and $\bar{N}_a$ for a triangular element.

The second method to diagonalize a matrix is to note that condition (H.1) is simply a requirement that ensures conservation of the quantity $\mathbf{c}$ over the element. For structural dynamics applications this is the conservation of mass at the element level. Accordingly, it has been noted that any lumping that preserves the integral of $\mathbf{c}$ on the element will lead to convergent results, although the rate of convergence may be lower than with use of a consistent $\mathbf{A}$. Many alternatives have been proposed based upon this method. The earliest procedures performed the diagonalization using physical intuition only. Later alternative algorithms were proposed. One suggestion, often called a "row sum" method, is to compute the diagonal matrix from

$$\mathbf{A}_{ab} = \begin{cases} \sum_c \int_{\Omega_a} \mathbf{N}_a^{\mathrm{T}} \mathbf{c}\, \mathbf{N}_c \, \mathrm{d}\Omega & a = b \\ \mathbf{0} & a \neq b \end{cases} \tag{H.4}$$

This simplifies to

$$\mathbf{A}_{ab} = \begin{cases} \int_{\Omega_a} \mathbf{N}_a^{\mathrm{T}} \mathbf{c} \, \mathrm{d}\Omega & a = b \\ \mathbf{0} & a \neq b \end{cases} \tag{H.5}$$

since the sum of the shape functions is unity. This algorithm makes sense only when the degrees of freedom of the problem all have the same physical interpretation. An alternative is to scale the diagonals of the consistent mass to satisfy the conservation requirement. In this case the diagonal matrix is deduced from

$$\mathbf{A}_{ab} = \begin{cases} m \int_{\Omega_a} \mathbf{N}_a^{\mathrm{T}} \mathbf{c}\, \mathbf{N}_b \, \mathrm{d}\Omega & a = b \\ \mathbf{0} & a \neq b \end{cases} \tag{H.6}$$

where $m$ is selected so that

$$\sum_a \mathbf{A}_{aa} = \int_\Omega \mathbf{c}\, d\Omega \tag{H.7}$$

The third procedure uses numerical integration to obtain a diagonal array without apparently introducing additional shape functions. Use of numerical integration to evaluate the $\mathbf{A}$ matrix of Eq. (H.1) yields a typical term in a summation form (following Chapter 6)

$$\mathbf{A}_{ab} = \int_\Omega \mathbf{N}_a^{\mathrm{T}} \mathbf{c}\, \mathbf{N}_b\, d\Omega = \sum_q \left(\mathbf{N}_a^{\mathrm{T}} \mathbf{c}\, \mathbf{N}_b\right)_{\boldsymbol{\xi}_q} j_q\, W_q \tag{H.8}$$

where $\boldsymbol{\xi}_q$ refers to the quadrature point at which the integrand is evaluated, $j_q$ is the Jacobian volume transformation at the same point, and $W_q$ gives the appropriate quadrature weight.

If the quadrature points for the numerical integration are located at nodes then, for standard shape functions (viz. Chapter 6), by Eq. (H.3) the diagonal matrix is given

$$\mathbf{A}_{ab} = \begin{cases} \mathbf{c}_a\, j_a\, W_a & a = b \\ \mathbf{0} & a \neq b \end{cases} \tag{H.9}$$

where $j_a$ is the Jacobian and $W_a$ is the quadrature weight at node $a$.

Appropriate weighting values may be deduced by requiring the quadrature formula to exactly integrate particular polynomials in the natural coordinate system. In general the quadrature should integrate a polynomial of the highest complete order in the shape functions. Thus, for four-node quadrilateral elements, linear functions should be exactly integrated. Integrating additional terms may lead to improved accuracy but is not required. Indeed, only conservation of $\mathbf{c}$ is required.

For low order elements, symmetry arguments may be used to lump the matrix. It is, for instance, obvious that in a simple triangular element little improvement can be obtained by any other lumping than the simple one in which the total $\mathbf{c}$ is distributed in three equal parts. For an eight-node two-dimensional isoparametric element no such obvious procedure is available. In Fig. H.2 we show the case of rectangular elements of four-, eight-, and nine-node type and lumping by Eqs. (H.5), (H.6), and (H.9).

It is noted that for the eight-node element some of the lumped quantities are negative when Eq. (H.5) or Eq. (H.9) is used. These will have some adverse effects in certain algorithms (e.g., time-stepping schemes to integrate transient problems) and preclude their use. In Fig. H.3 we show some lumped matrices for triangular elements computed by quadrature [i.e., Eq. (H.9)]. It is noted here that the cubic element has negative terms while the quadratic element has zero terms. The zero terms are particularly difficult to handle as the resulting diagonal $\mathbf{A}$ matrix no longer has full rank and thus may not be inverted.

Another aspect of lumping is the performance of the element when distorted from its parent element shape. For example, as a rectangular element is distorted and approaches a triangular shape it is desirable to have the limit triangular shape case behave appropriately. In the case of a four-node rectangular element the lumped

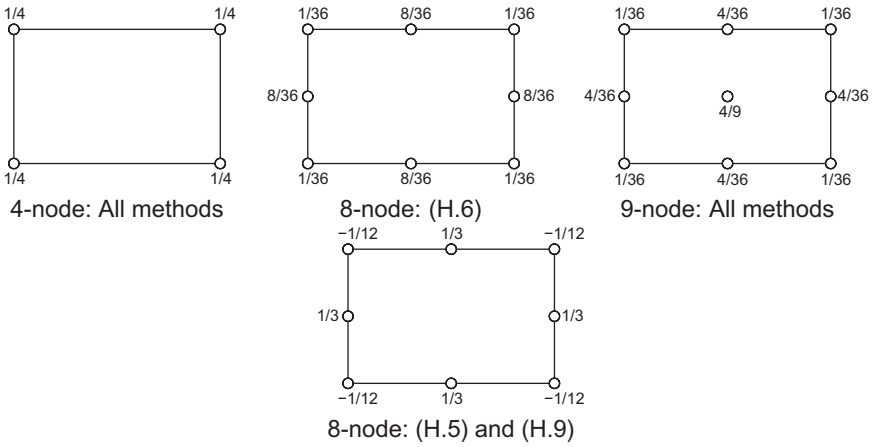



**FIGURE H.2**
Diagonalization of rectangular elements by three methods.

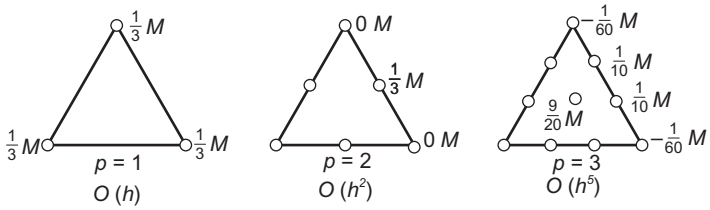



**FIGURE H.3**
Diagonalization of triangular elements by quadrature.

matrix for all three procedures gives the same answer. However if the element is distorted by a transformation defined by one parameter $f$ as shown in Fig. H.4 then the three lumping procedures discussed so far give different answers. The Jacobian transformation is given by

$$j = ab\,(1 - f\,\eta), \quad \eta = \frac{y}{b} \tag{H.10}$$

and $\mathbf{c}$ is here taken as the identity matrix.

The form (H.5) gives

$$\mathbf{A}_{aa} = \begin{cases} ab\,(1 - f/3) & \text{at top nodes} \\ ab\,(1 + f/3) & \text{at bottom nodes} \end{cases} \tag{H.11}$$

the form (H.6) gives

$$\mathbf{A}_{aa} = \begin{cases} ab\,(1 - f/2) & \text{at top nodes} \\ ab\,(1 + f/2) & \text{at bottom nodes} \end{cases} \tag{H.12}$$

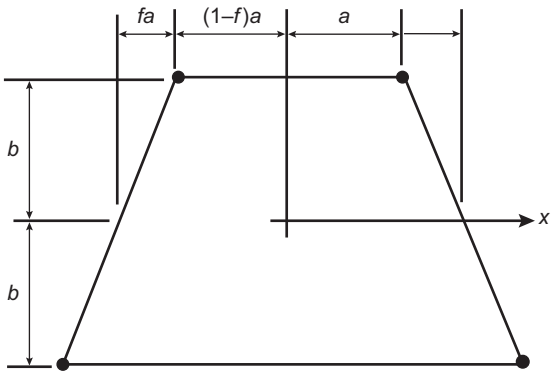



**FIGURE H.4**
Distorted four-node element.

and the quadrature form (H.9) yields

$$\mathbf{A}_{aa} = \begin{cases} ab\,(1-f) & \text{at top nodes} \\ ab\,(1+f) & \text{at bottom nodes} \end{cases} \tag{H.13}$$

The four-node element has the property that a triangle may be defined by coalescing two nodes and assigning them to the same global node in the mesh. Thus, the quadrilateral is identical to a three-node triangle when the parameter $f$ is unity. The limit value for the row sum method will give equal lumped terms at the three nodes while method (H.6) yields a lumped value for the coalesced node which is two-thirds the value at the other nodes and the quadrature method (H.9) yields a zero lumped value at the coalesced node. Thus, methods (H.6) and (H.9) give limit cases which depend on how the nodes are numbered to form each triangular element. This lack of invariance is not desirable in computer programs; hence for the four-node quadrilateral, method (H.5) appears to be superior to the other two. On the other hand, we have observed above that the row sum method (H.5) leads to negative diagonal elements for the eight-node element; hence there is no universal method for diagonalizing a matrix.

A fourth but not widely used method is available which may be explored to deduce a consistent matrix that is diagonal. This consists of making a mixed representation for the term creating the **A** matrix [1,2].

Consider a functional given by

$$\Pi_1 = \frac{1}{2}\int_\Omega \mathbf{u}^{\mathrm{T}}\mathbf{c}\,\mathbf{u}\,\mathrm{d}\Omega \tag{H.14}$$

The first variation of $\Pi_1$ yields

$$\delta\Pi_1 = \int_\Omega \delta\mathbf{u}^{\mathrm{T}}\mathbf{c}\,\mathbf{u}\,\mathrm{d}\Omega \tag{H.15}$$

Approximation using the standard form

$$\mathbf{u} \approx \hat{\mathbf{u}} = \mathbf{N}_a \tilde{\mathbf{u}}_a = \mathbf{N}\,\tilde{\mathbf{u}} \tag{H.16}$$

yields

$$\delta\Pi_1 = \delta\tilde{\mathbf{u}}^{\mathrm{T}} \int_\Omega \mathbf{N}^{\mathrm{T}}\mathbf{c}\,\mathbf{N}\,\mathrm{d}\Omega\,\tilde{\mathbf{u}} \tag{H.17}$$

This yields exactly the form for $\mathbf{A}$ given by Eq. (H.1).

We can construct an alternative mixed form by introducing a momenta type variable given by

$$\mathbf{p} = \mathbf{c}\,\mathbf{u} \tag{H.18}$$

A Hellinger-Reissner type mixed form may then be expressed as

$$\Pi_2 = \int_\Omega \mathbf{u}^{\mathrm{T}}\mathbf{p}\,\mathrm{d}\Omega - \frac{1}{2}\int_\Omega \mathbf{p}^{\mathrm{T}}\mathbf{c}^{-1}\,\mathbf{p}\,\mathrm{d}\Omega \tag{H.19}$$

and has the first variation

$$\delta\Pi_2 = \int_\Omega \delta\mathbf{u}^{\mathrm{T}}\mathbf{p}\,\mathrm{d}\Omega + \int_\Omega \delta\mathbf{p}^{\mathrm{T}}\left(\mathbf{u} - \mathbf{c}^{-1}\,\mathbf{p}\right)\,\mathrm{d}\Omega \tag{H.20}$$

The term with variation on $\mathbf{u}$ will combine with other terms and so is not set to zero; however the other term will not appear elsewhere and so can be solved separately.

If we now introduce an approximation for $\mathbf{p}$ as

$$\mathbf{p} \approx \hat{\mathbf{p}} = \mathbf{n}_b\,\tilde{\mathbf{p}}_b = \mathbf{n}\,\tilde{\mathbf{p}} \tag{H.21}$$

then the variational equation becomes

$$\delta\Pi_2 = \delta\tilde{\mathbf{u}}^{\mathrm{T}} \int_\Omega \mathbf{N}^{\mathrm{T}}\mathbf{n}\,\mathrm{d}\Omega\,\tilde{\mathbf{p}} + \delta\tilde{\mathbf{p}}^{\mathrm{T}}\left(\int_\Omega \mathbf{n}^{\mathrm{T}}\mathbf{N}\,\mathrm{d}\Omega\,\tilde{\mathbf{u}} - \int_\Omega \mathbf{n}^{\mathrm{T}}\mathbf{c}^{-1}\,\mathbf{n}\,\mathrm{d}\Omega\,\tilde{\mathbf{p}}\right) \tag{H.22}$$

If we now define the matrices

$$\begin{aligned}\mathbf{G} &= \int_\Omega \mathbf{n}^{\mathrm{T}}\mathbf{N}\,\mathrm{d}\Omega \\ \mathbf{H} &= \int_\Omega \mathbf{n}^{\mathrm{T}}\mathbf{c}^{-1}\mathbf{n}\,\mathrm{d}\Omega\end{aligned} \tag{H.23}$$

then the weak form is

$$\delta\Pi_2 = \begin{bmatrix}\delta\tilde{\mathbf{u}}^{\mathrm{T}} & \delta\tilde{\mathbf{p}}^{\mathrm{T}}\end{bmatrix}\begin{bmatrix}\mathbf{0} & \mathbf{G}^{\mathrm{T}} \\ \mathbf{G} & -\mathbf{H}\end{bmatrix}\begin{Bmatrix}\tilde{\mathbf{u}} \\ \tilde{\mathbf{p}}\end{Bmatrix} \tag{H.24}$$

Eliminating $\tilde{\mathbf{p}}$ using the second row of Eq. (H.24) gives

$$\mathbf{A} = \mathbf{G}^{\mathrm{T}}\mathbf{H}^{-1}\mathbf{G} \tag{H.25}$$

for which diagonal forms may now be sought. This form has again the same options as discussed above but, in addition, forms for the shape functions $\mathbf{n}$ can be sought which also renders the matrix diagonal.

## References

[1] T.J.R. Hughes, H.M. Hilber, R.L. Taylor, A reduction scheme for problems of structural dynamics, Int. J. Solids Struct. 12 (1976) 749–767.

[2] T.J.R. Hughes, Reduction scheme for some structural eigenvalue problems by a variational theorem, Int. J. Numer. Methods Eng. 10 (1976) 845–852.

This page intentionally left blank

# Author Index

Page numbers in **bold** are for pages at the end of chapters with names of author references.

## N



## O



## P

## Q



## R



## S

## T



## U



## V



## W

## X



## Y



## Z

# Subject Index

## D



## E

## F



## G

## H



## I



## J



## K



## L

## M

## N



## O



## P

## Q



## R

## S

## T

## U



## V



## W



## Z