

REPORT

Guide to the use of finite element and finite difference software

T J Eward and L Wright

December 2003

Guide to the use of finite element and finite difference software

T J Esward and L Wright
Centre for Mathematics and Scientific Computing

December 2003

ABSTRACT

Finite element and finite difference methods are becoming increasingly popular techniques within metrology for the numerical solution of continuous modelling problems. A wide range of software packages is currently in use and there is a need for guidance for metrologists on the choice of software and modelling methods and on testing and validating models. This guide to finite element and finite difference modelling introduces the kinds of problems that may be solved using finite element and finite difference techniques, reviews the assumptions that are made when choosing a solution method, and provides generic advice and guidance for the user of such software.

The first part of the guide takes the reader through the main processes of model development: choosing the method, identifying appropriate elements, constructing a mesh, choosing approximations, defining boundary conditions, defining model input parameters including properties of materials, solution techniques, post-processing and results analysis, and how to get support or help with the modelling problem itself or with use of the software. This is followed by a comparison of the most popular packages, provided by experienced users of the software in question, to give an overview of the strengths and weaknesses of each.

© Crown Copyright 2003
Reproduced by Permission of the Controller of HMSO

ISSN 1471-0005

National Physical Laboratory
Queens Road, Teddington, Middlesex, TW11 0LW

Extracts from this report may be reproduced provided the source is acknowledged and the extract is not taken out of context.

Approved on behalf of the Managing Director, NPL
by Dave Rayner, Head of the Centre for Mathematics and Scientific Computing

Contents

| | | |
|-------|--|----|
| 1 | Introduction..... | 1 |
| 1.1 | The background | 1 |
| 1.2 | Structure of the guide..... | 1 |
| 2 | Choosing a method | 3 |
| 2.1 | Introduction..... | 3 |
| 2.2 | The main numerical methods..... | 3 |
| 2.2.1 | The finite difference method..... | 3 |
| 2.2.2 | Finite element method | 5 |
| 2.2.3 | The boundary element method | 6 |
| 2.2.4 | The finite integration technique and finite volume methods | 7 |
| 2.3 | Criteria for selection | 7 |
| 2.3.1 | Which generic method? | 8 |
| 2.3.2 | Which software? | 8 |
| 2.3.3 | How long to develop the model? | 9 |
| 2.3.4 | How much will the model be used?..... | 10 |
| 2.3.5 | Can the modelling task be simplified?..... | 10 |
| 3 | Constructing a mesh..... | 12 |
| 3.1 | Introduction..... | 12 |
| 3.2 | Finite difference meshes | 13 |
| 3.2.1 | Staggered grids | 13 |
| 3.2.2 | Multigrid methods..... | 13 |
| 3.2.3 | Grid refinement..... | 14 |
| 3.2.4 | Sparse grids..... | 14 |
| 3.3 | Finite element meshes..... | 14 |
| 3.3.1 | Specifying the mesh..... | 14 |
| 3.3.2 | Mesh density | 16 |
| 3.4 | Choosing a type of element | 17 |
| 3.4.1 | Beams and bars | 17 |
| 3.4.2 | Membranes and plates | 18 |
| 3.4.3 | Shell elements | 19 |
| 3.4.4 | Three-dimensional solid elements | 19 |
| 3.4.5 | Thermal elements..... | 20 |
| 3.4.6 | Some other specialist elements..... | 20 |
| 3.5 | Automatic finite element mesh generation | 21 |
| 4 | Choosing approximations | 26 |
| 4.1 | Introduction..... | 26 |
| 4.2 | Symmetry..... | 26 |
| 4.3 | Reducing the dimensionality of a problem | 27 |
| 4.4 | Simplifying materials properties..... | 28 |
| 4.5 | Boundary conditions | 28 |
| 4.6 | Avoiding infinity..... | 28 |
| 4.7 | Lumped elements..... | 29 |
| 5 | Material properties | 30 |
| 5.1 | Describing linear materials | 30 |
| 5.2 | The limitations of linear materials | 30 |
| 5.3 | Non-linear material behaviour | 31 |
| 5.4 | Other sources of variability in materials properties..... | 32 |

| | | |
|--------|---|----|
| 5.5 | Piezoelectric materials | 32 |
| 5.6 | Testing non-linear models..... | 33 |
| 6 | Boundary conditions | 34 |
| 6.1 | Introduction..... | 34 |
| 6.2 | Representing loads in finite element models | 34 |
| 6.3 | Boundary conditions for symmetry | 35 |
| 6.4 | Non-linear boundary conditions | 35 |
| 6.5 | Singularities in finite difference models..... | 36 |
| 7 | Solution method..... | 37 |
| 7.1 | Introduction..... | 37 |
| 7.2 | Background to FE solution methods..... | 37 |
| 7.3 | Direct methods..... | 38 |
| 7.4 | Indirect methods..... | 38 |
| 8 | Results and post-processing..... | 40 |
| 8.1 | Introduction..... | 40 |
| 8.2 | Warning messages | 40 |
| 8.3 | Studying the results..... | 41 |
| 8.4 | Model validation | 42 |
| 9 | Getting support and help..... | 43 |
| 9.1 | Introduction..... | 43 |
| 9.2 | Other services provided by vendors..... | 43 |
| 9.3 | Help from colleagues | 43 |
| 9.4 | Consultants and academics | 44 |
| 9.5 | Other sources of help | 44 |
| 9.6 | Other software..... | 45 |
| 10 | Future trends | 46 |
| 11 | Survey of popular finite element and finite difference software | 47 |
| 11.1 | General-purpose commercial finite element packages | 47 |
| 11.1.1 | ABAQUS | 47 |
| 11.1.2 | ANSYS | 48 |
| 11.1.3 | FEMLAB | 48 |
| 11.1.4 | LS-DYNA | 49 |
| 11.1.5 | LUSAS..... | 49 |
| 11.1.6 | NASTRAN..... | 50 |
| 11.1.7 | PAFEC..... | 50 |
| 11.2 | Other commercial packages..... | 51 |
| 11.2.1 | General packages | 51 |
| 11.2.2 | Packages with a particular strength..... | 52 |
| 11.3 | Freeware and shareware..... | 53 |
| 11.4 | Finite difference packages | 53 |
| 11.4.1 | Particular finite difference packages..... | 53 |
| 11.5 | Other resources | 54 |
| 11.6 | Useful links | 54 |
| 12 | Users' testimonies..... | 55 |
| 12.1 | ABAQUS | 55 |
| 12.2 | ANSYS | 57 |
| 12.3 | FEAT..... | 61 |
| 12.4 | FEMLAB | 64 |
| 12.5 | LUSAS..... | 67 |
| 12.6 | PAFEC..... | 68 |

| | |
|---------------------------|----|
| Acknowledgements..... | 74 |
| Annex A: Glossary..... | 75 |
| Annex B: References | 80 |

1 Introduction

This report is a guide to the selection and use of finite element and finite difference software packages. It was produced as part of the “Continuous Modelling for Metrology” activity within the “Modelling Techniques” theme of the second Software Support for Metrology programme (SSfM, 2001-2004). It constitutes a deliverable of the fourth Work Package of the “Continuous Modelling for Metrology” activity.

1.1 The background

Continuous modelling is chiefly concerned with cases where the equations that are developed during the modelling process are either ordinary or partial differential equations. However, integral and integro-differential equations also occur. In determining a numerical solution to a model in cases where no analytical solution exists, it is important to choose a suitable numerical approximation scheme (discretisation) and subsequently an appropriate solution method for the discretised problem. The choice of approximation scheme and solution method can be crucial to obtaining an accurate solution in a reasonable time. Often finite element analysis has become the preferred choice, but it is not always the most suitable technique.

Continuous modelling tasks, especially those that can be formulated as finite element or finite difference problems, are frequently solved by using "off-the-shelf" software packages. Much of the commercially-available software was originally designed to solve a particular problem, and was later extended to solve other problems using the same technique. In such a case there is the risk that there is a particular range of applications for which the software is suitable and others where it is less suitable. Although these packages are useful tools, the user can easily lose sight of the errors inherent in the underlying numerical approximations and in determining the solution of the discretised system. Consequently, the user can have an overly optimistic view of the reliability of the solution.

1.2 Structure of the guide

This SSfM continuous modelling theme addresses issues in model validation and uncertainty evaluation for continuous models, with the aim of ensuring that good practice in this branch of modelling is adopted by metrologists. This guide to finite element and finite difference modelling is part of that process. The first part of the document, covered by chapters 2 through to 10, is a general guide to the kinds of problems that can be solved by each technique, the assumptions that are made when choosing a solution method, and the kind of information that will be needed regardless of which software is chosen (e.g. grid spacing, material properties, boundary conditions, geometry, etc.). It will describe the stages in setting up a continuous model. The guide also emphasises the importance of not confusing the usability of software with its correctness. There is not necessarily a correlation between user-friendliness and accuracy.

The structure of the first part of the guide is straightforward. It takes the reader through the main processes of model development: choosing the method, constructing a mesh, choosing approximations, identifying model input parameters, solution techniques, post-processing and results analysis, and how to get support or help with the modelling problem or with use of the software. A chapter is devoted to each of these topics.

The second part of the guide, chapters 11 and 12, contains a comparison of the most popular packages to give an overview of the strengths and weaknesses of each. This part includes a chapter summarising information about popular packages, as a result of information collected through a questionnaire. The questionnaire asked SSfM Club Members for details about their continuous modelling, including details of their software packages and reasons for the use of these packages. The packages that were most popular in the questionnaire responses are described in chapter 11. This part also includes a chapter that has been compiled by regular users of specific software packages and documents their experiences with the software in question.

Finally, the guide contains a glossary of technical terms. This should help the reader who is unfamiliar with the technical jargon of finite element and finite difference modelling, and aid communication between scientists and mathematicians.

2 Choosing a method

2.1 Introduction

It is not the aim of this guide to provide an introduction to the use of differential and partial differential equations in science and engineering. It is assumed that most readers will have received at least some exposure to the equations that are most commonly used to model physical processes – the wave equation, the diffusion or heat equation, and Poisson’s or Laplace’s equations. It is also likely that they will have studied analytical methods of solving at least some of these equations and have an understanding of the limits of analytical methods and the circumstances in which it is necessary to resort to numerical methods. Readers without this mathematical background may find it helpful to consult one of a range of standard textbooks, such as Ames (1992), Morton and Mayers (1994) and Smith (1985). It should be understood that although this guide concentrates on finite element and finite difference methods much of its advice is also relevant to other numerical methods, including boundary integral and boundary element methods, and finite volume and finite integration methods.

The number of problems involving differential equations that can be solved exactly is small. Most realistic problems require recourse to numerical methods and given the development of fast computers and availability of commercial software, numerical methods are often chosen as a matter of routine. As a result, there is often over-reliance on numerical answers combined with a false sense of security about the reliability of results (Gershenfeld, 1999). Other reports produced as part of the Software Support for Metrology programme specifically address the questions of model validation and errors in continuous modelling software and the reader of this guide is also likely to find those reports useful (Esward, Lord and Wright, 2003, Lord and Wright, 2003, Esward, Johnson and Wright, 2004). However, conscientious adoption of the advice set out in this guide should minimise the likelihood of gross errors in modelling.

2.2 The main numerical methods

2.2.1 The finite difference method

To understand how the finite difference method works it is helpful to consider a simple partial differential equation. Laplace’s equation can be used to represent a number of physical situations from the velocity potential for the steady flow of an incompressible non-viscous fluid to a steady-state temperature distribution problem. With the addition of a constant it becomes Poisson’s equation which can be employed to calculate the electrical potential associated with a charge distribution of known charge density (Smith, 1985). Laplace’s equation in two dimensions is:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$$

The analytical solution of this equation is a function of the spatial co-ordinates x and y that satisfies the partial differential equation at every point within the area of a plane closed curve and satisfies certain conditions at every point on this boundary curve (Smith, 1985, p2). Such a problem is a continuous modelling problem simply because the solution provides a value for every point in the domain of interest. Note also that the terms in Laplace’s equation are derivatives that apply at a point in space. The finite difference method approximates these derivatives by difference quotients over a small

interval, so that $\partial u/\partial x$ becomes $\delta u/\delta x$, for small values of δx and constant y . This approximation of the problem to be solved by difference equations also indicates the approach that must be taken to discretising the domain of interest in finite difference modelling. Typically the domain is discretised into a grid or mesh in which the mesh points are separated by intervals equivalent to δx and δy . Clearly, the easiest problems to tackle are those with rectangular domains and constant intervals between grid points in the direction of the problem co-ordinates (i.e. δx and δy need not be equal but they should each be constant).

To make finite difference approximations to derivatives Taylor's theorem is employed:

$$u(x+h) = u(x) + hu'(x) + \frac{1}{2}h^2u''(x) + \frac{1}{6}h^3u'''(x) + \dots,$$

$$u(x-h) = u(x) - hu'(x) + \frac{1}{2}h^2u''(x) - \frac{1}{6}h^3u'''(x) + \dots$$

Adding these two Taylor series expansions gives

$$u(x+h) + u(x-h) = 2u(x) + h^2u''(x) + \dots,$$

where terms with fourth or higher powers are ignored. Thus

$$u''(x) = \left(\frac{d^2u}{dx^2} \right)_{x=x_0} \cong \frac{1}{h^2} \{u(x+h) - 2u(x) + u(x-h)\}.$$

Subtracting the two Taylor series expansions gives

$$u'(x) = \left(\frac{du}{dx} \right)_{x=x_0} \cong \frac{1}{2h} \{u(x+h) - u(x-h)\}.$$

The above equation is a central difference approximation to the derivative. The forward-difference and backward-difference approximations are respectively:

$$u'(x) \cong \frac{1}{h} \{u(x+h) - u(x)\},$$

$$u'(x) \cong \frac{1}{h} \{u(x) - u(x-h)\}.$$

Employing the above equations in software requires generalisation of the equations and a method for identifying positions within the grid. Suppose that in the case of the Laplace equation example, the domain of interest is divided into a mesh of equal rectangles with sides $\delta x=h$, $\delta y=k$, with grid points in the x direction given by $x_i = ih$, $i = 0, \pm 1, \pm 2, \dots$ and in the y direction by $y_j = jk$, $j = 0, \pm 1, \pm 2, \dots$

If U_{ij} is the value of u at the point given by the co-ordinates (ih, jk) , then the equations above give, for example:

$$\left(\frac{\partial^2 u}{\partial x^2} \right)_{i,j} \cong \frac{u\{(i+1)h, jk\} - 2u\{ih, jk\} + u\{(i-1)h, jk\}}{h^2}$$

$$\left(\frac{\partial^2 u}{\partial x^2} \right)_{i,j} \cong \frac{U_{i+1,j} - 2U_{i,j} + U_{i-1,j}}{h^2}$$

Other derivatives can be obtained in a similar manner. The discretised forms of the equation at every grid point can then be combined into a large matrix of linear equations that can be solved to calculate the $U_{i,j}$.

The equations above have been set out at some length to give the reader a clear picture of how differential equations are approximated numerically and how difference equations can be turned into a form which can be utilised in software.

2.2.2 Finite element method

The finite element method, in common with the finite difference method, requires the modeller to set up a mesh or grid over which the problem of interest is solved. However, the motivation and the basis of the finite element approach is different, as the method does not involve the approximation of differential equations by difference equations. The method is usually traced back to the work of the German mathematician Richard Courant who is credited with introducing the concept of trial functions to simulate the behaviour of physical systems over small regions (Woodford et al, 1992). The trial function approximates the physical behaviour of a system of interest, a beam under load, say, or a region through which heat flows. Cook (1995) gives two interpretations of the process of assembling the trial functions and the small physical systems to which they apply into finite element models. The “unsophisticated description”, in Cook’s words, is that the finite element method involves cutting up a structure into several elements or pieces of the structure, describing the behaviour of each element in a simple way, and then reconnecting the elements at “nodes”. This process produces a set of simultaneous algebraic equations. In stress analysis, for example, these would be the equilibrium equations for the nodes. Cook’s “sophisticated description” is that the finite element method is piecewise polynomial interpolation. Within each element a field quantity such as displacement, temperature, or pressure is interpolated from values of the field quantity at the nodes. As the elements are connected together the field quantity is interpolated over the whole structure in a piecewise manner, with as many polynomial expressions as there are elements. Thus eventually a set of simultaneous equations for values of the field quantity of interest at the nodes is obtained. Values at positions not defined by nodes can be calculated using the interpolating polynomial for the element in question. Readers who wish to understand this process of setting up the basic equations of a finite element analysis are referred to the NAFEMS (National Agency for Finite Element Methods and Standards) Finite Element Primer (NAFEMS, 1992). Cook (1995) also contains a good description of this process.

It should be clear from the above description that the finite element method does not require a regular mesh or grid to define the problem domain. Provided the appropriate polynomials can be written, elements can take a range of sizes and shapes from one to three dimensions, and it is possible to assemble them into complex structures relatively easily. Essentially, the finite difference method employs a point-wise approximation to the equations of interest, whereas the finite element method works with a piecewise or regional approximation. Much commercial finite element software includes tools for generating meshes of complex structures rapidly, or for taking computer-aided design drawings and turning them into finite element meshes. This is one advantage which finite element methods have over the finite difference approach: the ability to cope easily with irregular grids and meshes.

To the users of most commercial finite element software the mathematics of the solution process is hidden, although it is usually necessary to define the problem type

(static or dynamic, linear or non-linear, steady state or transient, for example) and perhaps a solution method at the initial stages of setting up an analysis. Typically, the user works with the pre-processing and post-processing aspects of the software. In the pre-processing stage the finite element mesh is generated and the loading, the boundary conditions and the material properties are described. The post-processing stage is concerned with defining and using results output, either through text files listing numerical results or graphically. Often the calculations of interest, such as deriving a stress distribution from displacement data and materials properties, are carried out at the post-processing stage. These topics are discussed at greater length in the following chapters of this guide.

2.2.3 The boundary element method

For certain physics and engineering problems the boundary element technique may provide some advantages over finite element methods. The problems for which it is most suitable are those involving Laplace's equation (electrostatics, steady state heat flow, potentials) and the Helmholtz equation (electromagnetism, acoustics), and it has also been applied successfully to problems involving cracked materials. The main advantage of the boundary element method is that only the boundary of the domain of interest is discretised, with the result that the number of dimensions in a problem is reduced (a three dimensional problem becomes a problem over a surface, for example) with an associated reduction in required computing time. The partial differential equation that governs the system of interest is reformulated as an integral equation of the Fredholm type, that is, an integral equation with fixed limits on the integration domain. For computational purposes the integral is replaced by a numerical integration or quadrature scheme so that the integral equation is reduced to a linear system of equations and the solution of the system gives an estimate of the values of the unknown function at the quadrature points. Another advantage is that the integral formulation can make it possible to deal with problems involving singularity and discontinuity more easily since the integration often reduces the severity of such features.

Boundary element methods can, however, present substantial mathematical challenges to the user. An example of this is provided by the use of the boundary element method to solve the Helmholtz equation. In such a case the boundary element method is used to predict a field quantity (pressure in an acoustic problem, say) either for a bounded interior domain (e.g. the inside of a shell structure) or an unbounded exterior domain (the field surrounding a radiating surface such as a transducer). The problem is that the Helmholtz integral equation for the exterior problem does not have a unique solution at certain eigenfrequencies associated with the corresponding interior problem. Some methods of overcoming this difficulty have been proposed (Schenk, 1968, Burton and Miller, 1971) and Wu (2000) contains a discussion of this problem, and advice on the application of boundary elements to acoustic problems in particular.

Some finite element software packages also include the option to model exterior and interior domains using boundary elements rather than finite elements. An example of such a package is PAFEC. These packages allow the user to combine both types of analysis, so that a finite element mesh is produced for the structure of interest and then the interior or exterior region is represented using a boundary element mesh. Clearly, it would be impossible to represent an infinite exterior domain by finite elements but the boundary element method overcomes this problem as only the surface is modelled. The integral equation is then solved for the points of interest in the exterior domain.

2.2.4 The finite integration technique and finite volume methods

The Finite Volume Method of solving continuous models, which is employed in computational fluid dynamics (CFD), falls outside the scope of this guide. However, it is mathematically similar to the Finite Integration Technique (FIT), which is often used to investigate the interaction of waves with internal and external boundaries of solids. The first application of the Finite Integration Technique, due to Weiland (1984), was in electromagnetics, where it is known as EMFIT. The commercial software package MAFIA is based on EMFIT. More information about MAFIA can be found on the website of Computer Simulation Technology (<http://www.cst.de>). The method has since been extended to acoustic, elastodynamic and piezoelectric wave propagation problems.

In contrast to finite difference methods, the FIT procedure begins with the governing equations in their integral rather than differential form. The elastodynamic case, for example, begins with Cauchy's equation of motion and the time derivative of Hooke's Law. The FIT method carries out an integration over control volumes or integration cells and over the surfaces of these cells. It is claimed that the FIT approach produces a stable and efficient numerical code that allows a straightforward and flexible treatment on various boundary conditions on a staggered grid (Weiland, 1996). In addition it is possible to model heterogeneous media by defining material cells with different material properties. A useful short introduction to the technique can be found in Schubert et al (1998). The references cited in this paper are also of use in understanding this method.

2.3 Criteria for selection

The selection of a practical method of solving the continuous modelling problem of interest will depend on a range of factors, most of which are not strictly mathematical. In fact, several of them can be regarded as economic or project management questions. An additional question that is particularly relevant when a number of commercial software packages is available is, having identified a preferred method, which software should be chosen to implement that method. Chapter 12 of this guide, which documents the experiences of users of specific software, may be helpful here.

Before beginning the search for a suitable method and software, it is essential that the problem to be tackled has been properly defined. It is advisable to treat this definition stage as though a new software project is being planned and prepare a user specification and functional specification for the task. In this way it is possible to ensure that no aspect of the problem is overlooked and that all of the requirements of the software have been defined. The specification documents may prove useful in discussions with sales and technical staff employed by the software vendor. The same approach can be adopted if there is a need for software which may be applied to a range of problems in the future. Clear definitions of the likely modelling tasks are necessary to help ensure that an expensive piece of software is not unable to meet all the demands put on it because insufficient care was taken during initial investigations of software suitability.

Once the problem definition process has been completed the questions that need to be considered include:

- Which generic method should be used?
- Which software should be used?
- How long will it take to develop the model?

- How much will the model be used?
- Can the modelling task be simplified?

These questions are considered in the following sections.

2.3.1 Which generic method?

If software is to be developed in-house rather than obtained from commercial or other sources, once it has been shown that the methods under consideration can solve the problem of interest it is likely that the next important point to resolve will be how easy it is to program the problem. For simple geometries, homogeneous materials, and for straightforwardly-defined boundaries and boundary conditions, especially in two-dimensions or where symmetries can be exploited, the finite difference method may have advantages. Section 2.1 demonstrated how to set up the difference equations at a point and how to discretise spatial intervals and time intervals. In addition, there is much information in the applied mathematics literature on the selection of space and time steps to ensure stability in the solution (e.g. Smith, 1985, Morton and Mayers, 1994). If the problem also involves transient, rather than steady-state modelling, finite difference codes may run substantially faster than a finite element realisation of the same problem. In cases where the conditions set out above do not apply, or if the use of commercial software is being considered, finite element methods are likely to be preferred.

For those seriously considering the finite element method, NAFEMS produce a range of booklets which provide useful advice. The titles are:

- *Why do Finite Element Analysis?*
- *How to choose a Finite Element Analysis system*
- *How to get started with Finite Elements*
- *How to choose a Finite Element Pre- and Post-processor*
- *How to model with Finite Elements*
- *How to plan a Finite Element Analysis*
- *How to interpret Finite Element Results*
- *How to understand Finite Element Jargon*

If the problem of interest involves more than one area of physics and the interaction of systems governed by partial differential equations of different types, such as the interactions between a structure and fluid flowing past it, or combined thermal and structural interactions, it is likely to be necessary to consider the use of one of the commercial finite element codes that advertise themselves as “multiphysics” codes, such as ANSYS or ABAQUS. Such software often allows the user to combine different solution methods for different parts of the problem, such as using finite element methods and computational fluid dynamics methods to model an interaction between a structure and a fluid .

2.3.2 Which software?

Commercial finite difference codes are available to tackle specific modelling problems or specific types of physical problem and the potential user is advised to identify

software that has a track record of application in the field of interest. Much software for the solution of systems of partial differential equations can be found by searching websites such as the Guide to Available Mathematical Software (GAMS) at NIST (<http://gams.nist.gov/>), and Netlib (<http://www.netlib.org>). Furthermore, the implementation of the finite integration technique and the finite difference method are often very similar, especially when staggered grids are used for finite difference approximations. Thus, for electromagnetics problems, the MAFIA software package, as mentioned in section 2.2.4, may be of interest.

Finite element software is often marketed as being of general applicability, or at least applicable to a wide range of problems. For example, ANSYS divides its software into the following categories:

- *mechanical*, which can also tackle non-linear contact, acoustic, piezoelectric, thermal/structural and thermal/electric problems;
- *structural*, which includes linear and non-linear materials and inelastic materials;
- *computational fluid dynamics*, including particle transport and computational electromagnetics;
- *multiphysics*, with the ability to handle thermal/mechanical, fluid/structural and electrostatic/structural interactions.

It is, however, often the case that finite element software was developed to tackle a specific problem or class of problems and has then been modified or extended to allow it to be used for problems in other areas of science or engineering. For instance, ABAQUS was first developed to study problems in the nuclear power industry and in offshore engineering, where there were a number of complex, non-linear engineering problems. It is now used for a wide range of non-linear and linear finite element analyses that go beyond the applications for which it was originally established, including drop tests, crushing, and many manufacturing processes. Chapter 11 of this report provides information and website links for a range of software packages that the reader may find of interest.

It may also be helpful to look at the range of test and example problems that software suppliers often include in tutorial and validation manuals. Many of these will be routinely used by the software supplier itself to test and validate its own software, such as after an upgrade or the issue of a new software version. These example problems are often a good indicator of the range of problems which the software in question can tackle. If a problem type is not included among the available examples then it may be that the software has not been tested on that class of problems. The potential user may also find that example problems can be adapted to solve the problem in which he or she is interested, and may be useful in the early stages of developing a new model.

2.3.3 How long to develop the model?

Early in the process of identifying software it is beneficial to consider how much effort will be required to develop the model or models of interest. To a certain extent the answer to this question will depend on the knowledge and previous experience of the staff who will be preparing and using the model. In fact, the availability of staff with specific expertise in a certain software package or range of packages may be the single most important factor in deciding what approach to take to the problem, especially if it is necessary to obtain results quickly.

At this stage it is valuable to consider how the model will be validated and to draft the programme of tests that will be employed in validation. In this way, a more realistic estimate of the time it will take set up a model which is fit for purpose will be obtained. There is more to developing a model than simply coding it up. How the work will be documented should also be considered. If the work is being carried out as part of a metrology project, the documentation of the model itself, the test programme and the test results should form part of the traceability requirements for the software. If these questions are considered in the early stages of the project, then it is likely that a realistic understanding of the workload involved in preparing the model will be developed.

2.3.4 How much will the model be used?

The question of frequency of usage needs careful consideration at the start of the modelling process. Will the model be used to find a solution to a single problem, or will it be run many times, for example to investigate a range of material properties or to identify the sensitivity of the model output to changes in the input? If there is to be substantial re-use of the basic model, then it is here that the benefit of the time and effort put into setting up the original model can be reaped. In such cases, finite difference methods, which may take longer to set up and program than finite element approaches, may be a competitive solution. In such cases it may also be possible to separate the process of developing and validating the basic model from the process of carrying out repeat runs of the model and recording and analysing the output from such runs. It may therefore be possible to use less experienced or more junior staff to obtain many of the required results, thus freeing more experienced staff for developing further models. If this division of labour is likely to occur in practice, this should be built into the project specification stages so that it is possible to ensure that topics such as the usability of the software and the ease of carrying out repeated analyses are addressed early and are taken into consideration when specific software is being considered.

2.3.5 Can the modelling task be simplified?

A final consideration before embarking on the acquisition of software is whether a problem can be simplified, perhaps by reducing the number of dimensions or by transforming the governing equations into an alternative form, so that the problem can be solved semi-analytically, say, or be transformed from a non-linear to a linear problem.

For instance, a model has been developed (McCartney and Saunderson, 1995) of the stress and displacement distributions within a damaged cross-ply laminate under biaxial loading conditions. The full set of equations describing the displacements when the laminate is in equilibrium are a coupled set of second order partial differential equations. These equations do not have an analytic solution. Whilst it would be possible to approach this problem using finite element analysis, it would require a very fine mesh owing to the ply thicknesses being small compared to their length.

To avoid this, an assumption is made about the behaviour of one of the stress components, turning the problem into a semi-analytic one. It is assumed that the normal stress in the direction along the ply is independent of the through-thickness coordinate, so that the normal stress in the i^{th} ply can be written

$$\sigma_{yy}^i = \frac{[C_{i-1}(y) - C_i(y)]}{h_i} + \sigma_i.$$

This leads to a set of recursion relations for the unknown functions $C_i(y)$, and these recursion relations produce a set of coupled fourth-order ordinary differential equations. The fourth-order ordinary differential equations can be solved using techniques for generalised matrix eigenvalue problems. The software produced to implement this model takes significantly less time to run than the equivalent finite element model does, and the semi-analytic model requires a much shorter set-up time as it does not need a mesh.

An example of the benefits of transforming the equation of interest into a different form arises in solving Burgers' equation. This equation is a non-linear partial differential equation with the following form:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \lambda \frac{\partial^2 u}{\partial x^2}$$

Burgers' equation is often used as a simplified model for turbulence, boundary layer behaviour and shock wave formation. It can be reduced to the linear diffusion equation by the Cole-Hopf transformation (Whitham, 1974, p 97) by making the substitution

$$u = -2\lambda \frac{(d\phi/dx)}{\phi},$$

so that a linear form is obtained:

$$\frac{\partial \phi}{\partial t} = \lambda \frac{\partial^2 \phi}{\partial x^2}.$$

This form avoids the need to solve a non-linear equation and this may have computational benefits, especially if it allows the use of existing software rather than requiring the purchase or development of new software.

It is also important to consider whether finite element and finite difference methods are really necessary for the problem in question. In the experience of one of the authors (Esward, 2002), the use of finite element modelling to determine the distortion coefficient of oil-operated pressure balances of simple construction at pressures up to 200 MPa gave no advantage over simpler analytical solutions. Differences between the distortion coefficients predicted by finite elements and by the analytical solution were of the order of 0.2% to 0.3%. The experimental uncertainty on the determination of pressure balance distortion coefficients is approximately 10%.

3 Constructing a mesh

3.1 Introduction

This chapter provides some practical advice on what issues to consider when constructing a mesh either for finite difference or finite element problems. It reviews the different meshing problems encountered in the two modelling approaches and discusses some aspects of choice of element type in finite element modelling.

The mesh that is used for a particular problem has a fundamental effect on the numerical solution, in terms of its accuracy and the computing resources that are required. In the case of finite element modelling some general meshing rules that should be observed include: limit the number of required nodes to a minimum to reduce demands on computing resources; try to employ elements that do not need to be distorted too much from their basic shapes; ensure that regions in which there will be steep gradients of the field quantity of interest or in which boundaries have to be defined in detail are finely meshed.

The choice of co-ordinate systems to be employed in generating meshes is of importance for both finite difference and finite element methods. Although many problems can be tackled using right-handed Cartesian co-ordinates, other systems such as circular, cylindrical or spherical polar co-ordinates may allow more efficient computation of a range of solutions, especially those involving circular or axial symmetries. When meshing complex engineering structures for finite element analysis, it may be more convenient to use a number of different co-ordinate systems to generate different domains within the complete structure. Thus Cartesian co-ordinates may be used for components of the structure that are formed from straight lines, rectangular or brick shapes, but polar co-ordinates may allow curves and arcs to be defined more efficiently. Most modern commercial finite element software allows the user to change co-ordinate systems during the pre-processing or mesh generation phase, and some also allows users to define new co-ordinate systems for themselves.

The analysis of transient behaviour presents further problems when meshes are being designed. To obtain reliable results it is likely to be necessary to employ denser meshes than would be used for the equivalent static or steady-state analysis of the same structure. In addition, a time step has to be chosen that will capture the required features of the structure's response and give a stable solution for the chosen mesh size. The bandwidth of the transient excitation of the structure will therefore have to be taken into account when choosing step sizes. For example, the accuracy of the Newmark method for the solution of transient wave propagation, which is one of the popular methods of modelling transient behaviour, has been analysed by Wang et al (1992). They point out that accuracy in transient analysis of wave propagation problems depends both on the spatial and temporal discretisation. In addition it is not sufficient to consider mesh size and time step separately but their interdependence must be taken into account when modelling transient wave propagation problems. This is the case for many transient problems, since poor interaction between the time step and the mesh size can lead to unreliable results arising, for example, from unstable solutions and exponential error growth. More information on instability in transient models can be found in Esward, Lord and Wright (2003).

3.2 Finite difference meshes

As has been explained in section 2.2.1, finite difference methods are usually based on regular grid structures with simple boundaries. This is the approach most likely to be adopted if the aim is to code up a finite difference problem from first principles. Nevertheless, it is possible to employ staggered grid, multi-grid, grid refinement and hyperfine meshing techniques to improve upon the regular grid methodology.

3.2.1 Staggered grids

Intuitively it would seem likely that the most suitable approach to meshing and solving a partial differential equation using finite differences would be to locate all the variables on an evenly spaced grid, represent the variables at the grid points, and use the central difference approximation for the required derivatives. However, this method has a serious limitation. Consider the central difference approximation to the first derivative, which was defined above in chapter 2. The equation is:

$$u'(x) = \left(\frac{du}{dx} \right)_{x=x_0} \cong \frac{1}{2h} \{u(x+h) - u(x-h)\}$$

Note that although this is an approximation to the derivative at the point x , the value $u(x)$ does not appear in the equation. Note also that if $u(x+h) = u(x-h)$ the derivative is zero whatever value $u(x)$ takes. The extreme case is an alternating data set of the type: 0 1 0 1 0 1 ... etc., which would return a derivative of zero throughout its length.

In the staggered grid technique, instead of locating all the variables on one grid, different variables are located on different grids, these being shifted half a grid point. In two dimensions one variable may be placed at the centre of a cell in the mesh and other variables in the middle of the vertical and horizontal edges of the cell. In three dimensions a cubic volume is used and in this case further positions become available in addition to the centre of the cube – the centres of cube faces and also edges, for example. Staggered grid methods are commonly used for any problems that require simultaneous solution for pressure and velocity, and are often used to solve acoustic, seismic wave and elastodynamic wave propagation problems. The geophysics literature, in particular, is a rich source of advice and guidance on the use of staggered grid techniques.

3.2.2 Multigrid methods

To improve computing times for large finite difference calculations, multigrid methods are frequently employed. The introduction of these methods is due to Brandt (1977). These methods attempt to solve systems or equations more efficiently or more quickly by interpolating between grids of different density. Conceptually, they rely on the idea that a system on a coarse grid may be solved quickly but with a large error, and then the results of the coarse solution can be interpolated onto the more accurate fine grid as an approximation to the fine grid solution. The solution proceeds by iterating between the coarse and fine grid, obtaining estimates of the errors on each grid. Naturally, the technique is not limited to a simple coarse and fine grid system and additional layers are possible.

Multigrid methods appear to be efficient for linear and non-linear elliptic partial differential equations with smooth boundaries. However, for non-elliptic steady-state problems they appear to be slower than conventional methods. Press et al (1992, pp 862-880) has some useful advice on multigrid methods. In addition, the NAG library routine D03EDF uses multigrid methods to solve elliptic PDEs on a rectangular region.

3.2.3 Grid refinement

It is possible to vary the refinement of a mesh locally, so that part of the region of interest may be meshed coarsely and other parts meshed finely. One reason for the adoption of such an approach is that the use of grids with cells of identical size and geometry may lead to oversampling in models in which there are large contrasts in the properties of different materials, with consequent high demands on computer memory and long computation times. A problem to be overcome in the use of locally refined grids is, of course, the numerical mismatch in the region of transition between the coarse and fine grids. An example of how this problem can be overcome can be found in Schubert (2001) who presents an iterative grid refinement algorithm. This algorithm was tested by modelling the one-dimensional propagation of an elastic wave in a homogeneous medium and showed that an inadequate interpolation in the transition regime between the refined and non-refined region produce numerical dispersion and attenuation of the wave, but that his method could avoid these problems. The method can be applied iteratively so that meshes may be successively refined in the region of interest.

3.2.4 Sparse grids

Numerical methods in two or more dimensions which are based on rectangular grids suffer from the problem that mesh refinement leads to a rapid growth in the number of grid points in the problem. If N is the number of grid points in one direction, then the number of grid points is N^d where d is the number of dimensions in the problem. This law has consequences for computational run times and for computer memory and storage. In the last decade mathematicians concerned with developing solution techniques for finite difference and finite element problems have attempted to circumvent this problem by implementing sparse grid methods. This technique has been borrowed from numerical quadrature and approximation theory and decomposes the full grid into a hierarchical set of subspaces with successively fewer grid points. Control of the accuracy of the approximation becomes important with these methods. More information and discussion of suitable algorithms can be found in Zenger (1991), Bungartz and Dornseifer (1998) and Griebel (1998).

3.3 Finite element meshes

Unlike finite difference meshes, which are defined at specific points in the domain of interest, the finite element method interpolates the field quantity of interest over the whole of the region being modelled from the values these quantities take at nodal points. Thus, the finite element method allows the use of irregular meshes and different shapes and types of mesh elements in the same model. In addition, this flexibility allows complicated shapes to be modelled in a more straightforward manner than is possible with finite difference methods, and it makes it easier for the user to increase the concentration of mesh points at key locations in the model, such as regions of high strain.

3.3.1 Specifying the mesh

The first stage of the implementation of a finite element model is the construction of the mesh and the choice of appropriate element types. For all but the simplest structures, mesh generation is a skilled task which requires the practitioner to have substantial experience to be able to produce meshes which will give accurate answers. However, much modern finite element software incorporates automatic mesh generation routines

that can assist in defining meshes. For large problems these have the advantage that the user does not have to define the co-ordinates of each node in the model but can leave the finite element software to generate large parts of the detailed mesh. In addition, specialised pre-processing and mesh generation software is often available that allows graphical generation of meshes, or that will take input from computer-aided engineering software. The numerical description of the graphically-generated mesh is then made available to the finite element solver itself. Many finite element software packages, and especially the large commercially-available ones, have mesh checking routines built in to them, which will identify gross errors in meshing such as failure to close regions or failure to join different types of elements together correctly.

As NAFEMS (1992, p85) emphasises, it is not sufficient simply to begin mesh generation once the geometry of the model is known. NAFEMS argues that there are four pre-requisites for mesh generation – knowledge of the geometry, the boundary conditions, the loadings and the required results. In addition, some engineering or scientific appreciation of the problem being tackled is essential so that the modeller understands what simplifications and approximations it is reasonable to make and how the structure may be expected to behave under the applied loads. For example, it may be useful to consider if it is possible to reduce the complexity of model by working in two dimensions rather than three, or by assuming that a structure possesses axial symmetry. Such a decision requires an understanding of what information might be lost if such an assumption were to be implemented in the proposed mesh.

Knowledge of the boundary conditions and the applied loads is necessary so that nodal points for the application of the conditions are defined correctly and that the chosen element types allow the required conditions to be implemented. The boundary conditions and loading can also affect the simplifications that are possible, since a symmetric structure may deform in an unsymmetric manner under some loads, making a reduction in complexity less straightforward. Understanding what the peak values are likely to be and where they are likely to be located is also important in helping to design meshes, so that it can be ensured that the behaviour of interest is captured accurately by the model. Knowing what results are needed, including their required accuracy, is probably the most important single piece of information needed before beginning to build the model. It will define the choice of element and the nature of the mesh, as well as the solution technique which is chosen.

Mesh generation begins with the discretisation of the structure in a series of finite elements. This process typically requires the definition of the co-ordinates of the nodal points. This is followed by the topology of each element, that is, the manner and order in which the nodes are connected to form the element shapes, which must be carried out using the rules required by the finite element software being used. For example, the choice of a clock-wise or anti-clockwise method of joining nodes may be used in defining the direction of outward-pointing normals to element surfaces. Next it may be necessary to define certain aspects of the element's geometry, such as the thickness of plate and shell elements. Finally, the material properties of the element are required. Which material properties are necessary will depend on the type of finite element analysis that is to be carried out, but typical properties include density, Young's modulus, Poisson's ratio, and thermal expansion coefficient, for example.

Defining the mesh geometry is not always straightforward, especially when the structure of interest is not formed from straight lines, but contain arcs of varying radius, or less well-defined curves. If the curved line cannot be defined by an equation, it is

likely that some form of interpolation, perhaps using splines will be used to approximate the shape of the curve. In such cases, difficulties may arise when it is necessary to ensure that regions which should be connected together are joined properly. Not only must nodes be connected, but sides of elements that are required to meet should both be described by the same interpolation function, so that the structure remains connected as it is deformed under load.

As has been mentioned in section 2.2.2, finite element analysis could be thought of as being based on piecewise polynomial interpolation, and the order of the polynomial that is used in the interpolation can strongly affect the results obtained. First order elements use a linear interpolation so they cannot capture higher-order behaviour accurately without a denser mesh and they lead to derivatives (such as stress and strain) being constant within an element. Second order elements use a quadratic polynomial and so can capture higher-order behaviour but they require more nodes to define an element so the computational cost is higher. Higher order elements also exist, and some packages offer elements that guarantee continuity of derivatives so that stress and strain can be continuous across element boundaries.

3.3.2 Mesh density

The key to efficient finite element analysis is the correct choice of mesh density for the problem of interest. Too dense a mesh will make an analysis too costly and time-consuming, but too coarse a mesh will lead to incorrect results. To define a suitable mesh it is important to have an understanding of the distribution and gradients of the variables in the structure being modelled. A fine mesh is needed in regions in which there are large gradients, whereas a coarser mesh may be used in regions of constant, or at least slowly varying stress. Large gradients occur at discontinuities, which may be structural, or may be discontinuities in boundary conditions, loads, or material properties. Meshes should be graded in density so that they gradually become less dense over the range in which the effect of the discontinuity dies away.

It may be the case that results in the region of peak values or discontinuity are not of interest for a particular analysis. If the parts of the structure in which results are required are sufficiently far from the peak value regions a coarse mesh in the region of the discontinuity may be suitable, provided that the overall behaviour of the model is correct and that the coarse meshing implements the correct boundary conditions and can transmit loads to that part of the structure that is of interest. It must, however, be remembered that results in the region of the discontinuity itself will not be reliable.

All elements are defined in relation to a basic shape such as a square, an isosceles triangle or a cube. The further an element is distorted from the basic shape, the less accurate it becomes. In a regular mesh it is almost always possible to avoid large departures from basic element shapes, but in irregular meshes and in complicated geometries it is harder to achieve this aim. The amount of distortion that can be tolerated will depend on the element type – elements with mid-side nodes may be able to tolerate more distortion than those without mid-side nodes. Some commercial software packages give the user warnings when element distortions are unacceptable. Typical distortions include those due to poorly-selected aspect ratios for quadrilateral elements, angular distortion, where angles between adjacent sides are either very small or very large, and positional distortion of the mid-side node. Hannaby (1987, 1988b) discusses the problem of representing a circular domain with rectangular and triangular-shaped elements, especially in relation to the solution of Laplace's equation.

As the NAFEMS Primer points out (NAFEMS, 1992), one of the most difficult aspects of defining geometries and meshes is associated with the intersection of solids and surfaces, especially when attempting to define lines of intersection or blending intersections together. Ideally a mesh should not include regions where large changes in stiffness occur across element boundaries. NAFEMS suggest that for uniform materials and continuous geometries this implies that similar-sized elements be used either side of a boundary and that the volume of one element should be no more than two to four times the volume of an immediately adjacent element.

3.4 Choosing a type of element

In general, elements have simple one-, two-, or three-dimensional geometric shapes. They can also be classified by the loads which they can carry. For example, membrane elements are used for loads in the plane of the element as they have no stiffness normal to the plane. Bending elements carry loads which are applied in the plane normal to the plane in which the element lies. Shell elements combine the properties of membrane and bending elements. Solid elements are used for continuous three-dimensional structures and axisymmetric elements are used to represent solids which are generated by revolving a plane figure about an axis in the plane. In addition, there are various specialist elements, such as point masses, springs, pressure-based elements for acoustic propagation in fluids, and temperature-based elements for heat flow problems.

At an early stage in the modelling process it is advantageous to consider the cost of the proposed analysis. How much staff time will be needed to develop the model and how much computer time will be needed to solve it? Mesh density, choice of element type and model complexity are all factors which contribute to cost considerations. In general the simpler the element chosen, the shorter the time required for the solution. One possible rule-of-thumb is to choose the simplest element that will do the required job.

A question to be borne in mind throughout the discussion of element types which follows is: how many nodes per element? As the number of nodes in an element increases, the quality of the analysis can increase dramatically as extra mid-side nodes are added. This increase in quality is caused by the use of a higher-order polynomial for the interpolation. Elements with too few nodes tend to be too stiff. It could also be argued that in general quadrilateral elements are to be preferred to triangles and brick shapes are better than wedges or tetrahedra.

The remainder of this section reviews the properties of some of the more common element types.

3.4.1 Beams and bars

Beams, trusses and frames made from assemblies of beam-like structures are common engineering structures. It is therefore of interest to appreciate the manner in which beams and bars are represented in finite element analysis.

The simplest bar element is a tension bar, which is a straight uniform element that carries loads only at its ends. The element has only two nodes, one at each end, and one degree of freedom at each node, which relates to displacements parallel to the length of the beam. When it is loaded the axial force is constant along the beam length. The output from a stress calculation is the axial force in the beam, usually with the convention that tensile forces are taken to be positive. To define the element geometry it is only necessary to specify the co-ordinates of the end of each node and the cross-sectional area of the bar. Applications of such simple elements are limited to structures

with pin-joints or to beams with negligible bending and high torsional stiffness.

More complicated beam behaviour is often modelled with specialist beam elements which do not require a full three-dimensional specification. Some simple straight uniform beam elements which are defined by only two nodes can model bending in two directions, axial forces and twisting about the shear centre. Such elements require their properties and cross-section to be defined by the principal second moments of area about the y and x axis (the x axis lies along the beam), the cross-sectional area, and a torsional constant, which relates the torque and the shear modulus to the gradient of the angle of rotation with respect to x . This type of element can also model tubular bars and beams, provided that two of the inner and outer diameter, and wall thickness are also defined. Elements of this type will not represent shear deformation or rotary inertia and therefore should only be used for beams where the length is greater than at least five times the largest cross-section dimension. These limitations provide an example of why it is important to understand the results that are required from a finite element analysis before beginning work on defining a mesh and choosing element types.

Curved two-noded beams are also often used. In this case the element is assumed to be part of a circle, where the end nodes are positioned at the centres of the cross-sectional area of the two ends of the beam. Three-noded beam elements may also be used to model both curve and straight beams.

3.4.2 Membranes and plates

It is possible to regard membranes and plates as thin solids so that they could be represented by three-dimensional solid elements. However, such an approach introduces an extra computational burden as transverse normal stresses and transverse shear stresses will be computed, and these are regarded as negligible in plates. In addition, there is a risk of ill-conditioning, because the stiffness in the thickness direction will be much larger than other stiffnesses (Cook, 1995, p 176). To model plate-like structures efficiently, two-dimensional elements which carry loads only in their own plane are employed. In such a case it is assumed that stresses do not vary throughout the thickness of the element, so that a two-dimensional representation is adequate. The two geometries that are employed for plates are triangles and rectangles. These may possess nodes only at corners so that the sides of elements remain straight. If nodes exist at mid-side positions then curvilinear shapes are possible for element edges. Each node has only two degrees of freedom allowing displacements only in the plane and the main limitations of such elements are that they must be flat and that bending and twisting effects acting out of the plane of the element are ignored.

Commercial finite element software often allows plate elements to be used in modelling axisymmetric structures when loadings are also axisymmetric, provided that appropriate flags are set and that the axis of symmetry is defined in accordance with the correct protocols. Note that it is possible to model circular plates by using quadrilateral elements in axisymmetric mode. Further observations on modelling axisymmetric structures are included in section 3.4.4 below.

Users of this type of element need to understand how accuracy varies with aspect ratio. In addition, it is important to appreciate how far the triangle or rectangle may be distorted from its basic shape before results become unreliable.

3.4.3 Shell elements

In-plane effects can be modelled by the type of element described in the previous section. However, if both in-plane, and out-of-plane effects are important, shell elements should be considered.

For very thin plates only membrane effects are important. For thicker plates bending effects also have to be taken into account. As Woodford et al point out (1992, p86), plate elements may meet in such a way that bending in one plate is coupled to both bending and membrane effects in adjacent elements. Shell elements aim to combine both effects within a two-dimensional geometry. Such elements will have five or six degrees of freedom at a node, (three translations and a rotation about two or three axes, depending on whether rotation about the axis normal to the shell is incorporated). If the third rotation is not included in the element formulation, it will still have to be taken into account, as the global stiffness matrix will be singular if elements that share the same node happen to be flat and coplanar (Cook, 1995, p 186).

Limitations on the use of shell elements, and especially on those that are curved rather than flat, will depend on what form of shell theory the element has been based. For example, PAFEC's eight-noded facet shell element has the limitation that is based on thin plate bending theory and should not be used if the thickness of a plate is greater than about one fifth of typical plate sizes or radii of curvature (PAFEC, 1994).

Cook (1995, p 187) demonstrates how isoparametric shell elements can be developed from three-dimensional solid elements, specifically how a 20-noded block can be transformed to an eight-noded shell element. He also has good advice on the use of shell elements (1995, p 188). He points out that shells are probably the most difficult elements to formulate and that the behaviour of shell structures can be difficult to predict. Shell elements should be tested for validity, and sensitivity to shape distortion. Sample problems should be solved in which the results are known in advance. For flat geometry a shell should be able to model plane stress and plate bending. It should therefore pass tests designed to evaluate plane and plate elements. Shell elements may also be used to solve plane arch problems, which often have well-documented solutions.

3.4.4 Three-dimensional solid elements

In finite element modelling, a three-dimensional solid is one which has no restrictions on loading, material properties and boundary conditions. Thus, all possible stresses, including shear stresses are accounted for. Typical shapes for these kinds of element are tetrahedra and hexahedra.

Unless it is necessary for an adequate representation of the structure being modelled, it is advisable to avoid three-dimensional elements. Such models are the most difficult to prepare and to check for errors, and require greater computing resources (Cook, 1995, p143). It is also preferable that three-dimensional elements should only be employed with others of the same dimensions. Mixing elements of different dimensionalities should only be attempted by very experienced finite element practitioners.

A typical brick-shaped three-dimensional element may possess 20 nodes, one at each of the eight corner nodes and 12 nodes at the middle of each edge, with three translational degrees of freedom at each node. For three-dimensional elements, the best results will be obtained when the stiffness of the element is approximately equal in the three orthogonal directions.

As was mentioned in section 3.4.2, axisymmetric structures are three-dimensional

structures formed from a solid of revolution, and they are modelled using plane elements. It must be understood that plane shapes of this kind are actually cross-sections through the three-dimensional structure and that nodes are in fact nodal circles. What appear to be point loads when applied to a node in an apparently two-dimensional structure are in fact loads applied around the whole circumference of the axisymmetric solid. Specialised techniques exist for axisymmetric structures to which non-axisymmetric boundary conditions and loads are applied. One approach is to describe the load in terms of an infinite series, such as a Fourier series, calculate the response to a finite number of the terms of the series and sum the results in post-processing. The number of terms used in the series will be related to the accuracy produced in the solution. Similarly, plane strain and plane stress elements can be used to model two-dimensional cross-sections of three-dimensional structures under certain restrictions about the behaviour in the third direction. This is explained more fully in section 4.3.

3.4.5 Thermal elements

The rod or bar elements, two-dimensional and three-dimensional elements discussed so far have all been assumed to be structural elements. All these geometries have equivalents in thermal analyses. For example, a one-dimensional, two-noded thermal rod element may be used to model one-dimensional heat transfer, where heat flow is assumed to be proportional to the temperature difference between the ends of the rod. The heat flow equivalent of a plate or membrane element would be one in which temperature variations throughout the thickness of the structure can be neglected. Note that elements that are used simply for temperature calculations have only one degree of freedom, temperature, at each node.

When modelling thermal distortion of structures and stresses that arise from temperature variations it is necessary to use structural elements in which, in addition to structural parameters, the coefficient of thermal expansion and the thermal conductivity are built into the behaviour of the element. In practice, it may be necessary to perform a two-stage analysis. The first stage thermal run derives a temperature distribution and then this distribution is used as the input to a run to calculate stresses. For problems that involve surface heat transfer from the surface of a solid structure into a fluid, for example, specialist heat transfer elements exist. In two dimensions one side of the element would be defined by structural nodes and the opposite side would have nodes located in the fluid (these non-structural nodes may actually be defined at fictitious locations in some FE software). The only property that needs to be defined for such an element is the heat transfer coefficient and the degree of freedom at each node is temperature.

3.4.6 Some other specialist elements

Commercial finite element software packages may also offer a range of other more specialised elements, depending on the physical or engineering problems which the software is designed to solve. One type of specialist element which is often used is the piezoelectric element. These will have the common structural forms - planes, axisymmetric and three-dimensional solids - but will include an extra degree of freedom to define the electrical potential at each node. In addition the material properties will be orthotropic and it will be necessary for the user to define the direction in which the material has been poled. Piezoelectric elements may be earthed by restraining the electrical degree of freedom for the point or surface in question.

Some elements may be able to represent pressure fields of some kind. For example, the

pressure distribution in oil films in bearings derived using Reynold's equation may be of interest. Here an element with one degree of freedom, pressure, at each node is required. A similar requirement exists when acoustic wave propagation problems in fluids are being solved. Other specialist elements allow the addition of point masses to structures or the use of springs without having to model the spring explicitly.

Occasionally software may allow the user to define at least some aspects of element behaviour. One such example is provided by PAFEC finite element software, which has a user-supplied general element that allows the user to input stiffness and mass matrices relating to specific degrees of freedom. These may have been derived from specific measurements on materials or they may relate to degrees of freedom which have been modelled using an alternative software package, where values now have to be transferred to the PAFEC software.

3.5 Automatic finite element mesh generation

An example of what can be achieved with automatic mesh generation software is provided by FEMLAB. More information about this software can be found at <http://www.euro.femlab.com/>. FEMLAB is a general purpose finite element solver that operates within a Matlab environment. The figures presented in this section of the report relate to the study of a structural mechanics problem: a wall-mounted lug with a circular hole. The loading studied was a uniform downward pressure around the lower semicircle of the drilled hole. The lug was 0.125 m long, 0.5 m in height and 0.02 m thick. The radius of the hole was 0.015 m. FEMLAB's automatic mesh generation routines were used to mesh the model. In the first instance a two-dimensional version of the structure was analysed. Figure 1 shows the default mesh offered by FEMLAB. Note that FEMLAB offers only triangular or tetrahedral elements and that the elements displayed in this figure are six-noded triangles, so each side has a mid-side node and the interpolation functions for the element are quadratic. The mesh shown in figure 1 has 220 nodes and 332 elements. Note, too, the variability of the mesh density across the structure, with the coarsest mesh occurring at the edge that formed the junction with the wall and the densest mesh being around the circular hole. The mesh parameters were then altered and a mesh with increased density was generated. The results are shown in figure 2.

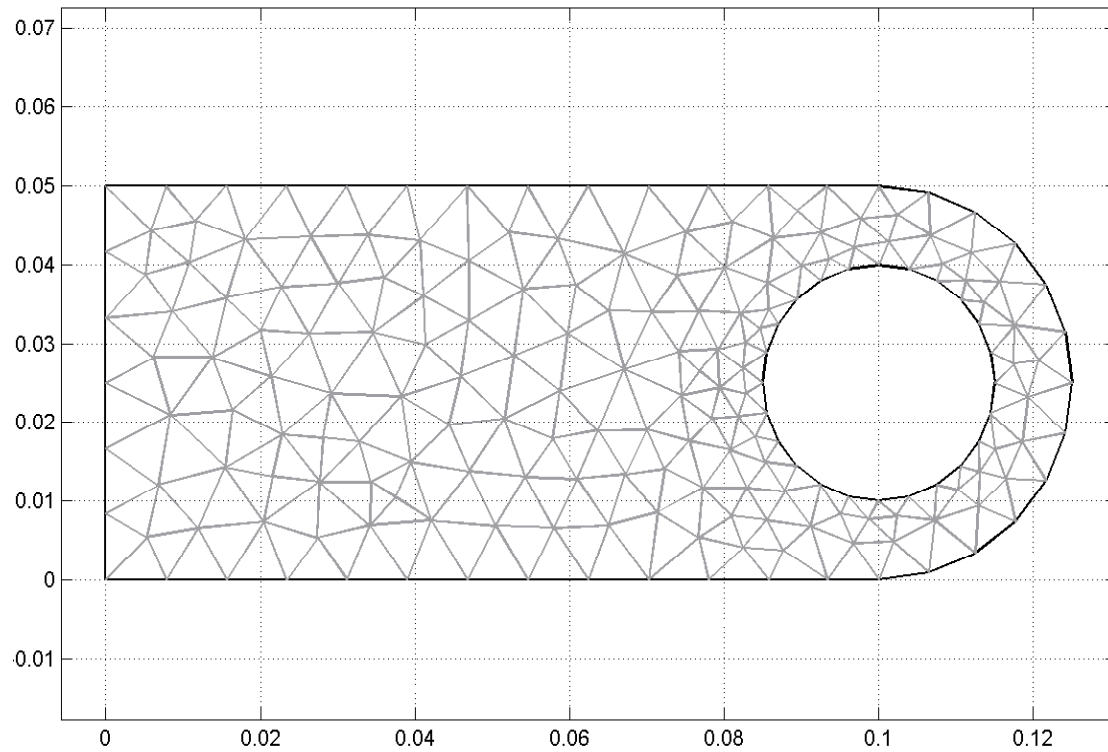


Figure 1: FEMLAB default mesh for plane stress analysis: 220 nodes, 332 elements

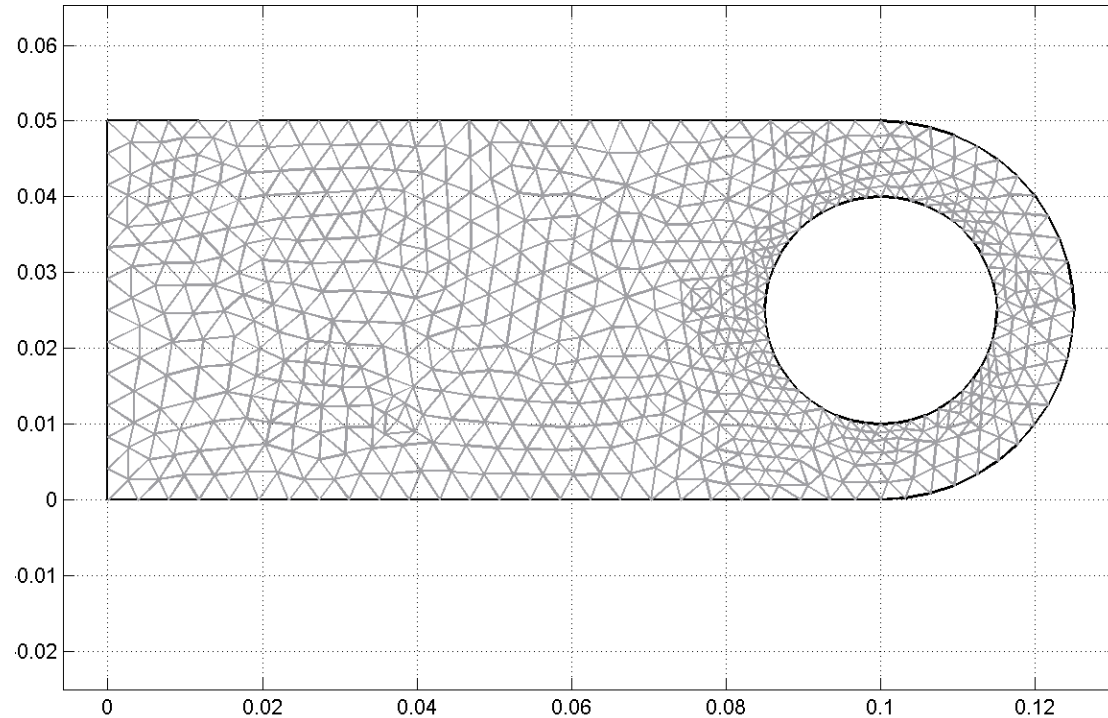


Figure 2: FEMLAB denser mesh for plane stress analysis: 732 nodes, 1328 elements

The mesh in figure 2 demonstrates the same tendency as that shown in figure 1. The automatic mesh generation system produces a denser mesh in the region of the circle, which is the area where the load was to be applied.

A three dimensional version of the problem was also studied. As before, the default three-dimensional mesh was generated. Figure 3 shows the result of this process. Here there are 705 nodes and 2519 elements.

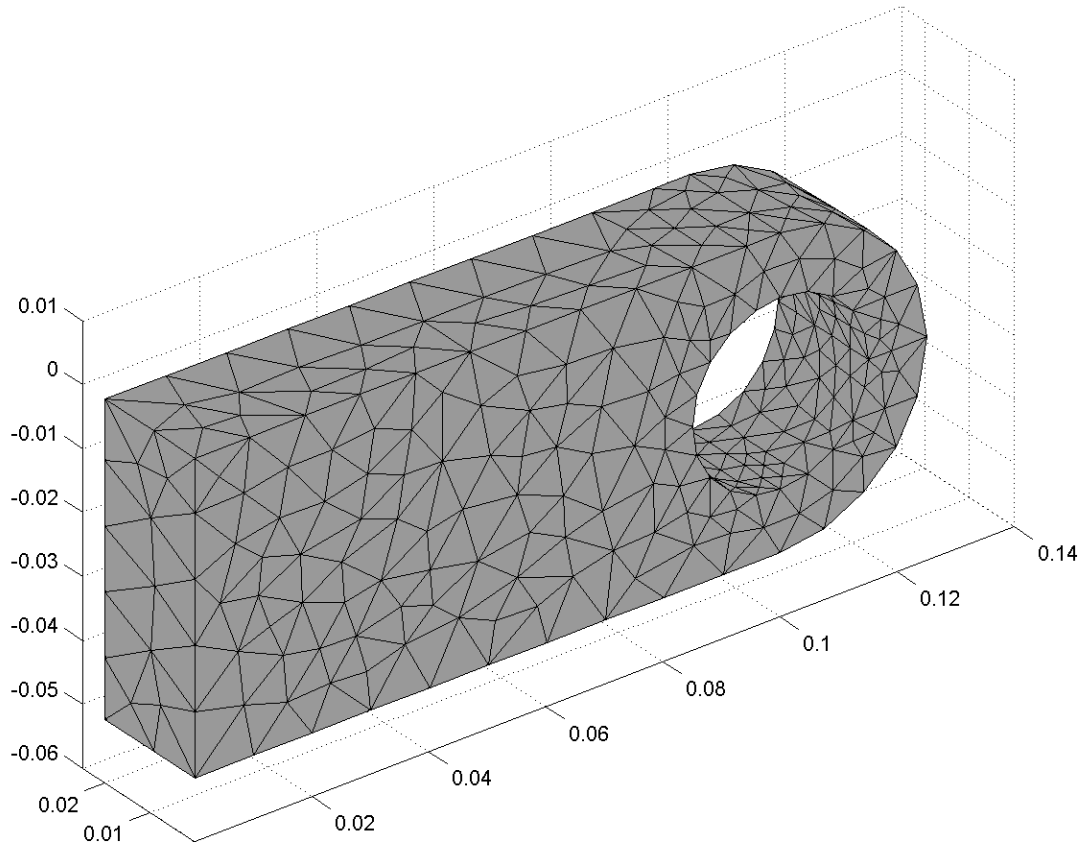


Figure 3: FEMLAB default three-dimensional mesh for structural analysis: 705 nodes and 2519 elements

Note the interesting manner in which the automatic mesher solves the problem of adding the third dimension to the model. Not only can the same gradation from a finer mesh near the hole to a coarser one at the edge next to the wall be seen, but the meshing of the thickness direction varies from two elements across the thickness to a maximum of nine around the volume of the hole.

Figure 4 shows a mesh of higher density generated using the same geometry.

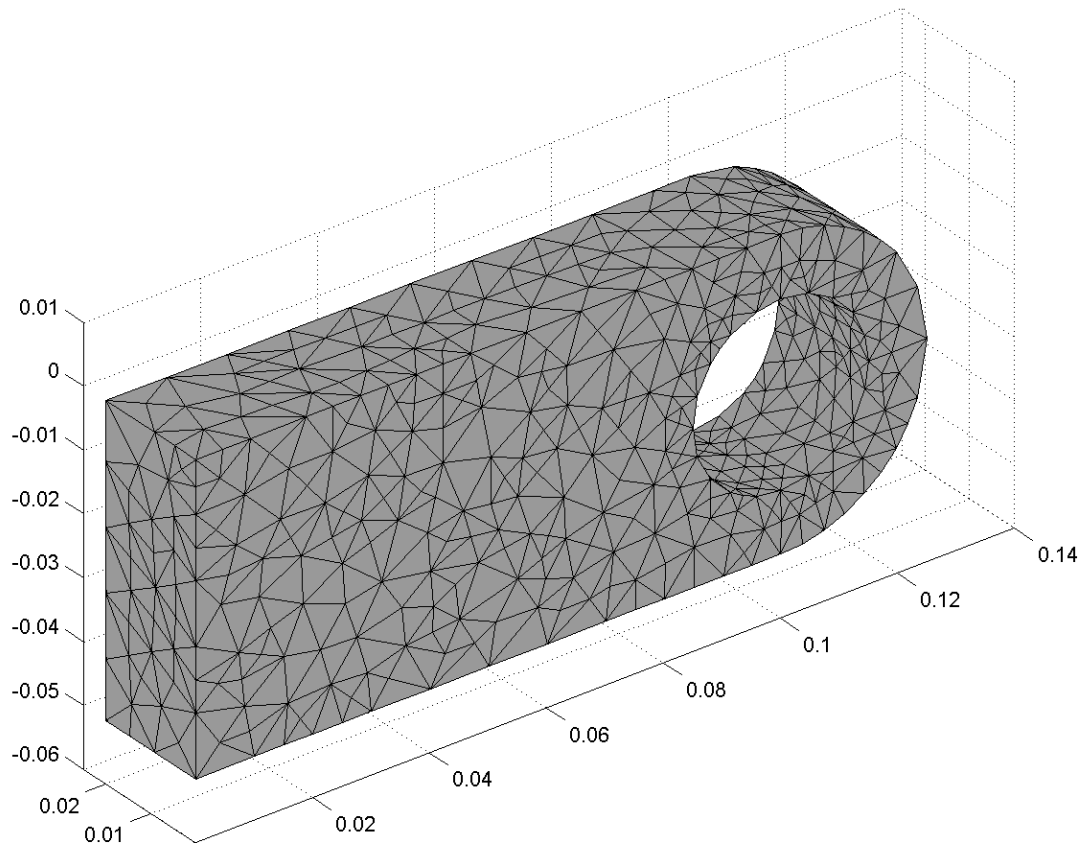


Figure 4: FEMLAB denser three-dimensional mesh for structural analysis: 2162 nodes and 9391 elements

The differences between figures 3 and 4 are not as visually striking as those between figures 1 and 2. However, in the case of figure 4, the main contribution to the increased node and element count arises from the increased element density through the thickness of the structure.

It is instructive to consider the computational cost of two and three dimensional models of the same structure. The lug considered here was meshed to a range of mesh densities in both two and three dimensions and a plot of the number of elements versus the number of nodes was produced for both the two-dimensional and three-dimensional model. Figure 5 shows the result. If a model can be restricted to two dimensions there are clear benefits in terms of model size and computation of results will be quicker. A 5,000 node three-dimensional model will have approximately 2.5 times the number of elements as a 5,000 node two-dimensional model.

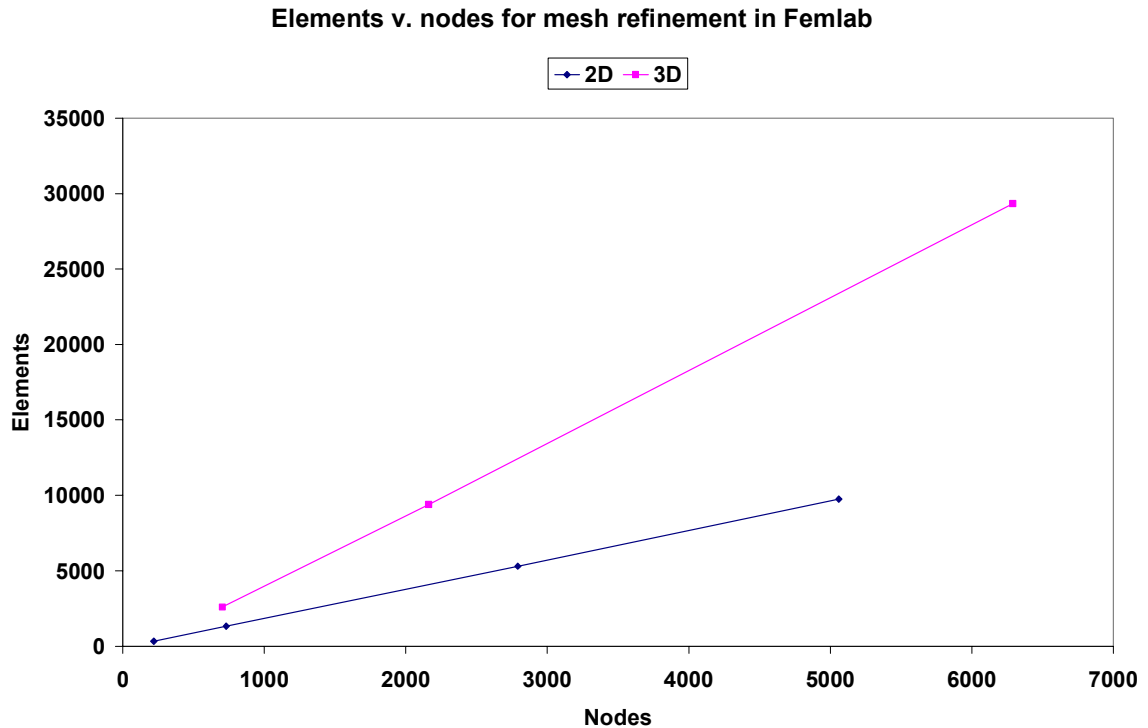


Figure 5: Mesh refinement for two- and three-dimensional models.

Some packages offer adaptive meshing tools. These tools alter the mesh as the calculation progresses, by using various error estimation techniques to identify regions where the errors are largest and refining the mesh in those regions. The refinement can be in the form of an increased mesh density (*h*-refinement), a higher order of interpolating polynomial (*p*-refinement), or both at once (*hp*-refinement). Generally the refinement continues until the estimated errors fall below some user-specified tolerance. These tools offer a good way of improving the accuracy of a solution without the user having to refine the mesh by hand. The main drawback of these tools is the increased run time, although often the improvement in results compensates for this delay. Another drawback is that time can be wasted by unnecessary refinement in regions where the solution is not of interest and does not affect the important results, although in some cases it is possible to limit the regions where adaptive meshing takes place.

4 Choosing approximations

4.1 Introduction

It must be recognised the whole process of producing a finite element or finite difference model is a process of approximation. Nevertheless, there are often computational advantages in introducing simplifications that may make it easier to generate a mesh or to reduce computation times. In fact, it may be helpful in the initial stages of studying a problem deliberately to produce simplified models.

If a complex structure is being modelled, the NAFEMS Finite Element Primer (NAFEMS, 1992) recommends that a series of small parametric studies should be conducted, in order to explore combinations of structural and element behaviour related to the problem of interest so that it can be ensured that the chosen elements and mesh are adequate for the proposed task. Another approach to analysing a complex structure is to start with a very simplified model and redefine it gradually as the design process develops. Thus a simple model is used when details of geometry and loadings are not well-defined, but as they become more precise the finite element model is refined iteratively. Such studies also help identify key features of the model where it is essential to capture all behaviour accurately and those features that are less important, where approximate representations or coarse meshes may be used with little deterioration in the quality or the results. If the ability to compare the results of finite element models with analytical solutions is of interest, then it is almost always likely to be the case that a much-simplified version of physical reality will be modelled.

4.2 Symmetry

The exploitation of symmetries in a problem is an effective way of reducing the size of a computational task. The previous chapter has already explained the advantages of axisymmetric representation of three-dimensional structures to reduce the number of degrees of freedom in a problem. Other symmetries can also be employed to advantage. Again, these are explained well in the NAFEMS Primer (NAFEMS, 1992, p 116), where, in addition to axial symmetry the following symmetries are identified:

- Mirror symmetry, in which one half of the structure is the mirror image of another in some plane of reflection.
- Cyclic symmetry, in which a structure is composed of repeating sectors around an axis rather than the rotation of a cross-section about the axis (which is axisymmetry). It may be possible to model only one of the repeating sectors. Where commercial finite element software allows this possibility, it is necessary to define the rotation axis, the cyclic order (the number of segments in the structure), and the nodes that define the adjacent boundaries of the segment. In PAFEC, for example, parameters identified as ROTATION.AXIS, CYCLIC.ORDER and CYCLIC.NODES must be defined to set up this type of analysis correctly. The boundary conditions and loads on each part of the repeating structure must be identical.
- Repetitive symmetry, where a structure is formed from sections that are repeated continuously. Physically, this is unrealistic as the structure must have a beginning and an end, but it may be possible to ignore end effects to analyse the

main structure. One way in which repetitive symmetry may be generated is by the employment of periodic boundary conditions.

It is possible the models may possess more than one symmetry. For example, there may be two or more mirror planes, allowing a quarter or an eighth of the structure to be modelled. In such cases generation of the mesh and the definition of boundary conditions and loads are easier if the user can define the planes of symmetry so that they are normal to the axes of the global co-ordinate system.

Exploitation of symmetries in a problem is simplest when both the structure and the loadings and boundary conditions have symmetry. Reference has already been made in section 3.4.2 to techniques for solving axisymmetric structures with non-axisymmetric boundary conditions. In the more general case of non-symmetric loads applied to a symmetric structure it may be necessary to analyse the structure by treating the load as the sum of a series of separate load cases in which each term is either symmetric or anti-symmetric (NAFEMS, 1992, p 116). This approach can only be applied to linear problems. If a non-linear problem is being analysed, then symmetry in both the geometry and the load is required to make it possible to take advantage of symmetry.

4.3 Reducing the dimensionality of a problem

The discussion of meshing and of element types in section 3 pointed out the benefits to be gained from avoiding three-dimensional modelling whenever possible. The definition of a plane body is one which is flat, lying in the (x, y) plane and of constant thickness. Following the NAFEMS Finite Element Primer (NAFEMS, 1992, pp 113-114), the stress-strain law for an isotropic material is defined by two material properties, Young's modulus, E , and Poisson's ratio, ν . Strains are represented by ϵ and stresses by σ .

$$\begin{bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ \epsilon_{xy} \\ \epsilon_{yz} \\ \epsilon_{zx} \end{bmatrix} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & -\nu & 0 & 0 & 0 \\ -\nu & 1 & -\nu & 0 & 0 & 0 \\ -\nu & -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2(1+\nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & 2(1+\nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & 2(1+\nu) \end{bmatrix} \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{xy} \\ \sigma_{yz} \\ \sigma_{zx} \end{bmatrix}$$

For a two dimensional system, two possibilities exist. In both cases the out-of-plane shear terms are zero, that is, ϵ_{yz} , ϵ_{zx} , σ_{yz} and σ_{zx} are zero. However, plane stress assumes that σ_{zz} is zero. If the thickness strain ϵ_{zz} is assumed to be zero, this is known as plane strain.

In the plane stress case there are no z direction stresses or shears and the thickness can increase or decrease in response to stresses in the (x, y) plane. In the plane strain case, changes in thickness are not allowed. Thus if it is possible to assume that strains perpendicular to the plane of interest are negligible, then there is little value in a three-dimensional model. A similar conclusion applies if there are no stresses normal to the surface of interest. Of course, in both cases bending stresses are also ignored.

4.4 Simplifying materials properties

In general the simplest method of representing a material is to assume that it is homogeneous and isotropic. For many common engineering materials whose behaviour is non-linear under large strains – metals and glasses, for example – these assumptions are perfectly adequate for many linear statics, dynamics or heat flow calculations. In such cases, and depending on the analysis, it may only be necessary to define Young's modulus, Poisson's ratio, density, and thermal conductivity. Some commercial finite element solvers include a library of standard materials properties which are available for common materials such as steel, iron, aluminium, glass, concrete and so on. It is important to recognise that these are often generic, book values and that for any specific analysis, the user should ensure that suitable material properties are employed. In some cases, measurements on material samples may be the best way of obtaining reliable data.

The challenges presented by modelling non-linear materials and materials with frequency dependent or strain-rate dependent properties are discussed in chapter 5. However, in the context of a discussion on methods of simplifying models, it is important to note that in some cases, such behaviour need not be modelled, for example, if only a limited frequency range is of interest or if properties vary only slightly in the region of interest. In addition, the use of simplified materials properties may be of benefit when testing or validating a modelling method prior to tackling the real problem of interest.

4.5 Boundary conditions

It is almost always the case that boundary conditions cannot be represented exactly in a model, indeed they may not be known in great detail or be incapable of exact definition especially at edges and corners, so that their representation in the model is often approximate. For structures the most commonly applied boundary conditions are to restrain or clamp movement in one direction or combination of directions. Where two edges meet, there will be a single point that belongs equally to both edges and representing the real behaviour of the structure at this point may not be possible. The same considerations apply to loads, both point loads and distributed loads, which are to be applied to the structure.

Where symmetry is being exploited to limit the size of the model it is essential to ensure that the applied boundary conditions reflect the symmetry requirements. In the case of axial symmetry, the axis itself must be restrained in a manner which ensures that the model remains truly axisymmetric.

Further discussion of the problem of representing boundary conditions adequately is set out in chapter six.

4.6 Avoiding infinity

In many areas of modelling, such as those involving heat flow, transport processes, or wave propagation, the domain of interest may be the whole of space external to an object such as a heat source or a transducer. By definition, finite elements cannot represent infinite regions, although, for some applications, infinite elements have been developed, which can be regarded as finite elements with one edge removed to infinity, and within which an interpolation function has been defined that allows the field quantity of interest to decay to zero. PAFEC, for example, uses this type of element, known within PAFEC as the wave envelope element, to model infinite regions of fluid

for acoustic wave propagation problems. Other methods of truncating a domain to avoid this difficulty include the use of boundary elements (these can, of course, only be applied where the material they are modelling can be regarded as homogeneous), perfectly matched layers (Berenger, 1994), which provide a reflectionless interface between the region of interest and the perfectly matched layers at all incident angles, Engquist-Majda boundary conditions for wave propagation problems (Engquist and Majda, 1977), and what are known as hybrid methods, which combine finite elements with the method of moments (Ali, Hubing and Drewniak, 1997). Earlier work at NPL on the use of infinite elements can be found in Hannaby (1988) and Hannaby (1994).

The methods mentioned in the previous paragraph are employed as a means of providing an artificial boundary to a finite region of space that has been modelled using finite elements. Although, to limit the computing task, an attempt to make the region modelled by finite elements as small as possible may be made, it is important to ensure that the boundary between the finite elements and the infinite elements or the perfectly matched layer is not too close to the source of the waves. Essentially, this boundary is introducing a mathematical discontinuity into the representation of the properties of the medium of interest and this may introduce artefacts, such as spurious reflections back into the finite element domain. If field quantities close to this artificial boundary are of interest, then particular care should be taken to ensure that the effects of the artificial boundary are minimised. The simplest form of artificial boundary is to place the boundary at a sufficient distance from the region in question that the behaviour of the event being modelled could be considered as having no effect on the behaviour at the boundary. For instance, a light bulb will affect the air temperature 1 mm away from it, but may not be significant 10 cm away, so an artificial boundary at that distance, fixed at a constant temperature, may be a good enough approximation to reality.

4.7 Lumped elements

Occasionally it may not be necessary to model the whole of a structure especially if it is known that the part of the structure under consideration does not play an important role in defining the behaviour of the structure. In this case, the use of simplified elements such as mass elements may be considered to model the unimportant regions as concentrated masses. Typically these may consist of two nodes, one defining the point of attachment to a structure and the other defining the location of the centre of mass. Thus the element will also incorporate a moment of inertia about a specific axis. Its main use is in dynamics problems to model concentrated masses. Note that such an element has no static stiffness.

Similarly, two-noded spring elements may be used to introduce a discrete stiffness into a model, typically to represent a more complicated structure that is not to be modelled completely, such as a connection to a foundation or to the ground. Translational and rotational stiffnesses may be defined for such elements.

5 Material properties

5.1 Describing linear materials

For many analyses of engineering structures it is often assumed that materials behave in a linear manner, that is, for small strains the material has an elastic response so that it returns to its original shape when the load is removed. In such cases the material is assumed to obey Hooke's Law, that the strains which are observed are proportional to the applied load, provided that the load does not exceed the elastic limit of the material.

At any point in a material there may exist up to six components of stress – three direct stresses and three shear stresses, and therefore six components of strain. Kolsky (1953, pp 7-8) defines Hooke's Law as: "each of the six components of stress is at any point a linear function of the six components of strain". This leads to a 6 by 6 matrix of coefficients which link the strains to the applied stresses, but these 36 coefficients are not independent in a real material, so that the number of independent coefficients is reduced to 21. Only in a material without any form of spatial symmetry would this number of coefficients be required to define the behaviour of the material. For cubic symmetry only three independent constants are required.

In an isotropic linear elastic solid, the values of the coefficients are independent of the axes chosen and only two independent coefficients are needed to describe the material. These are known as Lamé's constants, which are commonly written λ and μ . In finite element software, it is not likely that these constants will be entered directly. It is more common to use combinations of four other elastic constants: Young's modulus, Poisson's ratio, bulk modulus and rigidity modulus. The rigidity modulus is equal to μ ; the others can be expressed as functions of λ and μ . For a succinct discussion of Lamé's constants the reader is again referred to Kolsky (1953, p 9).

5.2 The limitations of linear materials

As NAFEMS point out in their guide to non-linear finite element analyses (NAFEMS, 1998), all structures exhibit non-linear behaviour to some extent. Materials such as rubber or plastic do not have constant moduli. The structure may operate over a range of temperatures and high temperature properties may differ from those at low temperatures. Linear approximations to materials behaviour may be adequate for many purposes, with the benefit that the finite element analysis is easier to perform.

The advantages of linear behaviour are that loads and their effects can be scaled. A doubling of the load should produce a doubling in the displacement. Similarly, the loading history of the structure can be ignored, i.e. residual stresses are not considered. The principle of superposition allows the results from the applications of different loads to be added together to calculate the combined effect of all the loads.

NAFEMS (1998) list the limitations which arise from the assumption of constant material stiffness and it is worth quoting their remarks in full:

- *The material properties must remain constant. This precludes the analysis of any structure in which the material might yield and undergo plastic deformation or in which the material stiffness is dependent on the strain, the strain rate or a temperature field which varies with time.*

- *Changes in the structural stiffness that may arise from contact between parts of the structure and any subsequent frictional sliding cannot be taken into account in a linear analysis.*
- *The linear analysis is invalid if the structure undergoes large deflections, because significant changes in the shape or orientation of a structure can alter its stiffness.*

5.3 Non-linear material behaviour

All materials are affected by their environment and by the forces and loads which are applied to them. In addition, the same material can exhibit apparently different behaviour when it is formed into different shapes. For example, a metal block may return to its original shape after a load is removed, but the same metal formed into a thin wire may be bent or twisted and not return to its original shape once the applied force has been removed. In other cases, behaviour may depend on how long a load is applied, so that we may observe creep, the gradual deformation of a material under a constant force.

In other cases it may be the manner in which the load is applied which may be important. Glass exposed suddenly to a high temperature may fracture, whereas a slow application of heat may not produce the same effect. Forces applied suddenly – impacts – may cause fracture when the same force applied slowly may not. Repeated application of a small load may lead to fracture through fatigue.

Materials may behave differently in different environments. High temperatures often soften materials, whereas low temperatures may make material more brittle. Brittle materials may be more susceptible to fracture from impact loads. Other environmental factors which affect the behaviour of materials include ionising radiation, corrosion, wear and humidity.

Many of the phenomena described above may be ignored for the purposes of finite element modelling. Nevertheless, it is possible to represent a wide range of non-linear behaviours in analyses.

The simplest non-linear behaviour to understand in common engineering materials such as metals is that of the elastic limit or yield stress. As has been pointed out, the linear, Hooke's Law assumption is that for small loads a material returns to its original shape when unloaded – this behaviour is elastic deformation. However, above a certain stress the material will not recover its size and shape and becomes permanently deformed. The stress at which this occurs is the elastic limit of the material. For real materials it can be very difficult to determine the point at which the onset of permanent deformation occurs, but for most practical purposes a yield stress can be defined below which behaviour is elastic and above which some plastic deformation occurs. If a specimen is loaded and unloaded repeatedly, the phenomenon of strain hardening may also be observed. Because of previous plastic deformation a greater stress may be required to produce additional plastic deformation. Thus the yield stress for the plastically deformed material exceeds that of the material prior to any plastic deformation. To describe such materials it may be necessary to specify no more than the material's Young's modulus, Poisson's ratio, yield stress and hardening modulus. In practice, it is of course possible that metals may demonstrate strain rate dependence, so that the yield stress and the hardening modulus depend on the rate of deformation of the material. These properties may in addition be temperature-dependent.

Elastomeric materials – rubbers and polymers – may have no permanent deformation after loading and unloading and so can be regarded as elastic. However, their stiffness is not constant. Behaviour may be strain rate dependent and also display hysteresis effects.

Many commercial FE packages include a wide range of pre-programmed material models. Many of these models were designed for use on a specific type of material (for instance some packages have models specifically designed for modelling soils), and others are more general models of specific types of behaviour (for instance the Drucker-Prager plasticity model can be used for a range of metals, adhesives and polymers by choosing the appropriate parameter values). The majority of these models are non-linear and so will take longer to run due to the need for solution of complicated equations, but if the correct model has been chosen the improvement in results compensates for the increase in run time.

Some packages offer users the opportunity to program their own material formulations, but in general this is a complicated undertaking and should only be attempted by expert users.

5.4 Other sources of variability in materials properties

Even if non-linear deformations of materials are not of interest, there are other circumstances in which materials may have to be described in a relatively complex manner, especially where dynamic behaviour is being modelled. For example, in acoustics analyses, properties such as attenuation and damping are frequency dependent. When damping ratios are entered as a function of frequency by the user it is important to understand how the finite element software calculates damping for frequencies that were not entered by the user. Does the software simply use linear interpolation or does it interpolate in some other more complicated way, or is it possible for the user to define the method of interpolation? This information should be available in the manuals that accompany good quality commercial software.

In thermal analyses properties may vary with temperature or time, or with both temperature and time. Software should at least allow the user to enter tables of data that define the way in which non-constant properties vary. In some cases it may be possible to define functional relationships that the solver can use for non-constant behaviours. Once again, refer to manuals for guidance on specific software.

5.5 Piezoelectric materials

Piezoelectric materials present their own specific problems in finite element modelling. They are orthotropic by nature and it is necessary to define material compliances in the principal directions in the material, the cross compliances and the shear compliances. For thermal models, the coefficients of thermal expansion along the principal axes of the material and the thermal conductivities in the principal directions will also be required. Piezoelectric constants d (strain/electric field at constant stress or charge density/stress at constant electric field; units: m/V) and e (stress/electric field at constant strain or charge density/strain at constant field; units: C/m²) will also be required for the principal axes of the material. In addition, the software will either assume a direction of poling or the user will be required to define the direction of poling of the material. Piezoelectric ceramics are assumed to be isotropic in the plane perpendicular to the polarisation axis.

Manufacturers of piezoelectric materials often do not publish sufficient data for all the parameters required in finite element modelling to be defined. Even where data are

available the associated uncertainties may be large to the extent that values may only be accurate to within $\pm 20\%$. In addition, for some ceramics the electrical properties may be frequency dependent, and these may also not be well-defined.

5.6 Testing non-linear models

How should the non-linear finite element model results be validated? NAFEMS (1998) make a fundamental point in their non-linear modelling guide: both linear and non-linear analyses should give the same result when a linear problem is being analysed. This approach can be very useful in testing non-linear FE models.

In general it may be advisable, when tackling non-linear problems, to begin with studies of simple cases – for example, analysing only weakly non-linear cases in the first instance. If these tests are successful and the user is confident that the software is producing reliable results in these simpler cases, then it may be safe to proceed to the analysis of more complex non-linear behaviour. Note also that non-linear analyses may require a finer mesh than the linear analysis of the same structure. A useful approach may be to study problems that require only a few elements and to which loads can be applied in a straightforward manner – tension and compression loads along a principal axis of the element, and shear loads along a principal axis. The non-linear behaviour of many materials, especially polymers and viscoelastic materials, has been extensively studied experimentally, and comparisons with experimental behaviour are likely to be an effective way of checking that finite element analyses are producing plausible results.

Some analytical solutions and some well-documented numerical solutions exist for a range of non-linear problems, such as elastic perfectly-plastic materials, strain hardening materials and rigid perfectly-plastic materials. These are usually limited to specific geometries, such as thick-walled cylinders with internal pressure loads, or plane structures with cracks, notches and holes. Cook (1995, p 289) summarises these cases and the reader who wishes to test non-linear finite element software may find these of use.

6 Boundary conditions

6.1 Introduction

This chapter is concerned with boundary conditions in the most general sense. In finite element analyses, two main types of boundary condition exist: restraints, which limit the behaviour of nodes or elements in some way (for example, not allowing displacements in a particular direction), and loads, which are the application of forces, pressures, temperatures and so on. In most cases boundary conditions are idealised or simplified. An edge of a structure may be clamped completely or restrained from moving in one particular direction. Local variations along the edge may be ignored. Loads may be applied in a similar way, so that the structure sees a homogeneous application of the load on the surface or edge of interest. Similarly there may be abrupt transitions from loaded or restrained regions to unloaded and unrestrained regions. In fact a corner node may belong both to a restrained edge and an unrestrained edge of a structure.

Another common idealisation, particularly for engineering structures, is to model them *in vacuo*, that is, to ignore the effects of the environment in which they find themselves. While this may be reasonable in the case of large structures in air, it is likely to be unreasonable in the case of underwater structures, for example, when the water load is likely to play an important part in determining the behaviour of the structure. Indeed, the determination of the effect of water loading may be the purpose of the analysis. In the case of wave propagation problems, it may be the propagation of the wave in the environment surrounding some source or transducer that is of interest, rather than the behaviour of the transducer itself. Obviously in such cases the environment in which the structure finds itself cannot be ignored.

Commercial finite element software typically allows restraints and loads to be displayed graphically in some way. Usually such displays allow identification of the nodes to which restraints or loads have been applied, and if appropriate, the displacement directions that have been restrained. However, it is unlikely that the display will allow the representation of the numerical value of loads. Nevertheless, such facilities allow the user to make a visual check of the model to ensure that boundary conditions seem reasonable and have at least been applied in the intended locations.

6.2 Representing loads in finite element models

The NAFEMS Finite Element Primer (NAFEMS, 1992, pp 257-261) gives good practical advice on the accurate representation of loads and it is worth quoting their advice in some detail. They classify loads in three ways:

- *applied displacements*, that is, values of displacements that are given as initial conditions (some of these displacements may, in fact, have values of zero);
- *mechanical loads*, such as point loads, inertia forces, surface pressure and the self-weight of the structure;
- *initial strain or initial stress loads*, for example thermal loads, or loads associated with lack of fit or non-linear materials properties.

The Primer points out that mechanical loads are likely to be the most common form of loading in finite element modelling and that the finite element method only takes into account loads applied at the nodal point, so that the simplest form of loading is to define

specific loads and specific nodal points. For distributed loads, such as pressures applied to surfaces, the question of how to approximate the distributed load by loads applied at specific nodal points arises. This is best performed by the use of what are known as kinematically equivalent nodal loads (the point loads at the nodes that do the same work under any actual set of nodal displacements as the actual load). For a more detailed discussion of this point, readers are referred to the Primer itself (NAFEMS, 1992).

In thermal analyses involving radiative heat transfer it will be necessary to define absolute temperatures rather than relative temperatures to ensure that the model is physically accurate. In addition, it will be necessary to define the direction of a surface flux as either inward or outward. Thermal analyses also provide the most common case in which boundary conditions may be both temperature and time dependent.

6.3 Boundary conditions for symmetry

Throughout this report we have emphasised the benefits to be gained from exploiting symmetry in finite element models. In such cases it is essential to ensure that the boundary conditions applied are consistent with the requirements of the various symmetries that may be being used. The simplest case to consider is probably that of axial symmetry. The axis of symmetry will need to be restrained so that nodes along it are not allowed to move radially, otherwise the symmetry condition will be lost. In a thermal analysis heat does not flow across a plane of symmetry, so that such a plane becomes an insulated boundary. In addition, it is worth noting that some commercial software requires a specific axis direction to be used as the axis of symmetry.

More generally, structures can possess symmetries about more than one axis and in addition the boundary conditions may be symmetric or anti-symmetric about such axes. Note that structural symmetries for non-linear problems can only be exploited if the loading has the same symmetry as the geometry. For the three dimensional case there are eight possible combinations of symmetry and anti-symmetry. In the linear case and for a general load which is not symmetric, it may be necessary to solve and sum a series of symmetric and anti-symmetric load cases to derive the general result. For axisymmetric structures where the loading is not itself axisymmetric it may be possible to use Fourier series methods to solve a general linear loading problem by representing the load as a Fourier series and solving the finite element problem for each term of the series. The result for the load in question is then the sum of the individual results for each term in the series.

6.4 Non-linear boundary conditions

Non-linear boundary conditions may also arise. The most common examples of these in analyses of mechanical structures are associated with contact between components, including impacts, friction between a component and another moving component, and metal and polymer shape-forming processes in which there is progressive change in the extent of contact between a material and a die, for example. Models involving contact conditions often simulate cases where components are not in contact initially but interact to an increasing degree as the simulation progresses. Non-linear conditions also occur in heat transfer problems where radiative losses are significant, or when the heat transfer coefficient describing loss from a surface is a function of temperature.

If the reader is likely to be analysing such cases regularly then it might be preferable to employ commercial software that has been developed specifically for the study of non-linear problems, such as LUSAS or ABAQUS.

6.5 Singularities in finite difference models

As Tayler (1986) emphasises, estimates of the error between the solution of a differential equation and a finite difference approximation to the equation at or near a singular point, either on the boundary or in the interior of an integration domain, cannot be valid. Such cases often occur at corners of the domain in rectangular co-ordinates, where some kind of discontinuity in boundary conditions exists. For example, a point force may be applied, or a pressure or temperature may change abruptly from zero to some other value. In the neighbourhood of such points finite difference methods may produce unreliable results, which may spread and adversely affect the results in the region of the mesh close to such discontinuities. However, provided the chosen solution method is stable, the errors introduced by the discontinuity will decay. Tayler (1986) reviews methods of dealing with singularities in finite difference approximations and his text should be consulted by those who wish to investigate this problem further.

7 Solution method

7.1 Introduction

For most users of finite element software, the exact method the software uses to solve the problem is not a question of much interest. Often the software does not allow the user any choice as to which solution method is used. However, sometimes knowledge of the solution method and any approximations it makes can lead to improvements in the model to decrease run time or increase accuracy.

The simplest form of finite element analysis, such as a time-independent statics problem requires the solution of the system of equations described by $\mathbf{k}\mathbf{d}=\mathbf{r}$, where \mathbf{k} , which is square and symmetric, is the global stiffness matrix, \mathbf{r} represents a vector of loads applied to the structure, and \mathbf{d} is a vector of results for each degree of freedom, such as nodal displacements. Typically such systems are solved either by direct methods that invert the matrix explicitly or by indirect, iterative methods that usually require an initial guess at a solution, which is then improved iteratively until sufficient accuracy is obtained. Note that sets of linear simultaneous algebraic equations can be solved by either direct or indirect methods, but the non-linear equations can only be solved iteratively unless they are linearised.

The most suitable solution method depends on the nature of the matrix \mathbf{k} and how it has been formed. The global stiffness matrix is built up from the individual element stiffness matrices, with the software ensuring that the coefficients are placed in the correct rows and columns of the global matrix. For real problems \mathbf{k} will be sparse, that is, it will contain many zeroes. In large models perhaps more than 99% of the individual k_{ij} values will be zero, so that software will employ sparse matrix formats for storing and processing data. Ideally, all non-zero coefficients should be located along a band that follows the diagonal of the matrix. A straightforward method of representing the sparse matrix \mathbf{k} is to store coefficients in a one-dimensional array, which is obtained by working down each column of the matrix and storing the values that are found between the first nonzero value and the matrix diagonal.

Coefficients may be stored in \mathbf{k} either by node numbers or element numbers. This is typically not under the control of the user of the software. Most software is able to re-arrange and re-organise node and element numbers internally to obtain the most efficient structure of \mathbf{k} for future processing. Once the problem is solved the software then converts back to the original numbering so that the user can display and analyse the results using the numbering employed when the model was created. Usually this process of re-ordering and re-numbering will be completely hidden from the user.

7.2 Background to FE solution methods

It is possible to regard the finite element method as a form of the Rayleigh-Ritz method. This is an approximation method which was first described by Lord Rayleigh in the 19th century and developed further by Ritz at the beginning of the 20th century. An assumption is made about the displacement of a body in relation to a set of generalised co-ordinates. The assumed displacement field is applied to the whole body being analysed, rather than to elements in a piecewise fashion, so no elements or nodes are defined. The field must satisfy suitable boundary conditions and constraints on the internal behaviour of the structure. The aim is to minimise an energy expression that includes the strain energy of the body and the work performed by applied loads. This minimisation with respect to the generalised co-ordinates leads to a system of algebraic

equations that is solved for the co-ordinates. The difference between this approach and the finite element method is that the finite element method defines the displacement field in a piecewise manner and, instead of generalised co-ordinates, the nodal degrees of freedom are employed as variables in a system of algebraic equations.

The Rayleigh-Ritz method only allows the structure to deform into shapes consistent with the assumed displacement field. This has a limiting effect on the model and acts to apply constraints that may not allow the calculation of the “true” displacement field. The constraints are considered to have a “stiffening effect” so that the method produces either exact displacements or displacements that are less than those of the “true” solution of the mathematical model in question. The under-estimation of displacements does not occur at every point in the model, rather the work done by applied loads is too small, so that average displacement is too small. Note that this implies that a finite element model is more likely to be too stiff than too flexible.

7.3 Direct methods

The most frequently employed direct method for the solution of systems of equations such as those derived from finite element analyses are the various forms of Gaussian elimination and Cholesky factorisation. The NAFEMS Primer (NAFEMS, 1992) has a short description of both methods. The advantage of direct methods is that the computational size of the problem is predictable as it can be estimated from the size of k and its topology. However, it is not necessarily the fastest method as iterative methods may be quicker if there is a large degree of connectivity between elements, such as arises in three-dimensional structures. For systems with low connectivity between elements, direct methods are likely to be faster than iterative approaches. For cases in which several load cases have to be analysed, the advantage of direct methods is that the reduction of the global stiffness matrix k need only be performed once. The PAFEC Theory manual (PAFEC, 1984) contains a good account of some of the computing problems that arise in direct methods, especially those concerned with efficient use of computer memory.

7.4 Indirect methods

One of the limitations of direct methods is that as the number of equations increases, these methods become more susceptible to rounding problems, so that for large systems errors may become large. In such cases iterative methods may be preferred. These all function in a similar manner: an assumption is made about the initial values of the unknowns, and each unknown is then recalculated in a systematic manner until the solution converges to a pre-defined limit. The main problem associated with iterative techniques is the time taken for convergence to occur.

Two specific iterative methods applied to the solution of linear systems are Gauss-Seidel iteration and successive over-relaxation, which is a modification of the Gauss-Seidel method. In the case of the Gauss-Seidel method an initial value is assumed for each of the unknowns. The equations to be solved are re-organised so that the unknown is expressed explicitly in terms of other unknowns. The most recently calculated values for each of the variables are then used in each of these reorganised equations. One iteration corresponds to one complete recalculation of all the unknowns. This procedure is repeated until the change in value for each unknown between one iteration and the next is less than a pre-determined limit. To ensure convergence it may be necessary to ensure that the explicit equation written for each particular unknown is the equation that possesses the largest coefficient for that unknown. Successive over-relaxation is a

technique for obtaining faster convergence in the Gauss-Seidel method by adding to the previous value of the unknown the difference between it and the value obtained from the present iteration multiplied by a relaxation parameter, usually in the range from 1 to 2. This is effective because it is often observed in iterative solution techniques that the successive values approach the true value in an asymptotic manner. The relaxation parameter accelerates the rate at which the estimates approach the final value.

For non-linear equations in which unknowns are raised to powers other than unity or are multiplied together, the above methods cannot be applied. Techniques such as Newton's method can be applied in such cases and the reader who is interested in learning more about such methods is referred to standard mathematical texts on the solution of non-linear systems of equations (Ames, 1972, Huntley and Johnson, 1983).

8 Results and post-processing

8.1 Introduction

Interpreting and evaluating results is the most important stage of the finite element modelling process as the sole purpose of the modelling is to produce results. Yet it is often the case that users of finite element modelling software do not pay sufficient attention to this stage of the process, but are inclined to accept uncritically the program output. It is therefore important to develop a consistent approach to the evaluation of results which ensures that the user can be confident the results can be accepted. Most commercial software includes post-processing routines, particularly graphical displays, contour plots and so on which can assist in this process, although it is important for the user to appreciate that it is possible to be misled by such plots.

The NAFEMS Primer (NAFEMS, 1992) includes a useful chapter on results processing and NAFEMS has also produced two short booklets that are also relevant here: *How to interpret finite element results* (NAFEMS, 1997) and *How to choose a finite element pre- and post-processor* (NAFEMS, 1994). These texts give useful practical guidance on how to study finite element results in a systematic way. The recommendations could form the basis of a quality assurance scheme to establish good practice in finite element modelling.

8.2 Warning messages

Much commercial finite element software produces output which warns the user of specific problems with the analysis. These may either be messages which are displayed on-screen in a window during and after the analysis or warnings and error messages written to files. The best systems combine both approaches so that all warnings displayed on the screen will also be recorded in files. In addition, there may be separate files of warnings and error messages or these may be interspersed into results files as advice and observations about the results.

It is likely that the warning process will have been encountered during model development prior to the solution process. A common check built into many finite element programs is the testing of the geometry of models to ensure that elements are not being distorted too much from their basic shapes, for example, that aspect ratios are reasonable for quadrilateral and cubic elements and that angles between the adjacent sides of triangular and tetrahedral elements are neither too large nor too small. Other warnings that may be provided during model development may be related to material properties, typically that not all available properties have been defined by the user. It is usually the case that this occurs because the particular property in question is not required for the planned analysis. Software may also carry out checks to ensure that structures are closed and continuous and will warn the user if they are not. If boundary elements are being employed to model an exterior or interior region, the program is also likely to produce a warning if the boundary elements that have been defined do not form a continuous line or surface.

Other information that a program typically provides at some stage during the model development process relates to the statistics of the model – the number of nodes and the number of elements of particular types, for example.

During the solution process commercial finite element software may also provide information concerning data that has been ignored such as the application of a load to

an incorrectly identified node or identifying cases in which results have been requested for nodes that have not been defined. Warnings on ill-conditioning of matrices and insufficient constraints on the structure are also likely to be provided.

It can often be difficult to identify the specific problem which led to the production of the warnings. The warning message itself may be unclear or difficult to interpret, or the same warning may appear to be repeated many times. However, all warnings and error messages should be reviewed by the user. It may be that in the end the warning relates to a matter that can safely be ignored. Nevertheless, this should not be assumed. An analysis run that has completed successfully and has plausible results may not have produced satisfactorily accurate results.

8.3 Studying the results

A systematic approach to analysing the results of the model should be adopted. For structural analyses it may be advisable to begin with a study of the various graphical outputs which the software provides. Displacements should be plotted and scrutinised in detail. Animated displacement plots will indicate the direction of nodal displacements. The predicted displacements should also be compared with the boundary conditions and with the loading of the structure. At restrained boundaries, do the displacements have a zero value? Do sliding, rotating and hinged joints behave as expected? If the model contains symmetries or if symmetry properties have been used to simplify the modelling task, is this reflected in the results?

However, it is important to recognise that displacement plots can be misleading. Components separated by gaps may be shown as interfering in the displacement plot, simply because the display exaggerates the distorted shape. See Esward, Lord and Wright (2003, section 5.1) for an example of this problem. Distorted shapes may also give a misleading picture. The graphical display may show large apparently non-linear distortions of the structure, when in fact the true displacements are small and within the linear regime. These misunderstandings arise because software often displays greatly exaggerated displacements. Here it is important to study the results data files rather than accept the graphical output, as it is the numerical data that are the actual output of the analysis. It is possible that plotting the numerical results in an alternative software package, rather than relying on the graphics built into the finite element software, may give a more reliable representation of the results.

The results should also be viewed from varying angles and through varying cross-sections. This is particularly important for three-dimensional structures. Also, take advantage of contouring facilities as these present an alternative view of the data in question.

We recommend studying predicted displacements first as these are often easier to understand than other plots, such as stress plots. Finite element software typically offers a range of tools for representing stresses and it is important to appreciate fully what the software is showing before studying the stress plots which the program offers. For example, to what co-ordinate system (local or global) are the stresses referred? Is the software showing bending moments rather than stresses? For plates and shells, are the results for upper, middle or lower surfaces of the element? Are stresses averaged in some way, or has the software performed some smoothing process before presenting the results? Which stress is of interest – for example, normal stress, shear stress, or the von Mises stress? Stress contour plots may also be of use and may inform the decision as to whether the stress gradients and stress concentrations are reasonable.

Cook (1995, p135) provides some useful guidance on evaluating stress fields, which is worth quoting at length:

Stress contours should be normal to a plane of reflective symmetry (of loads as well as geometry). At a free boundary, one of the principal stresses should be zero. At a boundary loaded only by a pressure p , one of the principal stresses should be $-p$. Principal stress trajectories should be normal or tangent to free boundaries, boundaries loaded only by pressure p , and planes of reflective symmetry. In an axially symmetric problem, radial and circumferential normal stresses should be equal on the axis of revolution. None of these conditions is likely to be met perfectly. The amount of imperfection is a measure of the discretisation error, or is perhaps a warning of an error in the FE model.

8.4 Model validation

If the results pass the tests based on visual scrutiny, the next stage is to attempt to validate the model in a systematic way. For many applied scientists and engineers, validation or verification of a model consists of no more than comparison with experimental results. This is unacceptable as a validation method, for the following reasons:

- It is necessary to have quantified both the measurement error and the modelling error for a meaningful comparison.
- The experiment may investigate only a limited subset of the results produced by the model; for example, the model may predict an extended temperature, stress or pressure field but the experiment may only give results at a limited number of points in the field.
- The model may be a simplified or idealised presentation of the problem of interest, whereas the experiment is taking place in the real, imperfect world.
- Sources of variation in the experimental results (the laboratory environment itself, for example) may not be present in the model.
- Input parameters to the model may not be in agreement with the experimental case. For example, the model will use an exact value for Young's modulus or Poisson's ratio for the materials it models. Do you know that these values are truly representative of the actual materials used in the experimental apparatus?

It is also important to recognise that one of the mechanisms of scientific advance is the difference between theoretical predictions and experiment. The attempt to understand and take into account these differences leads both to better theory and better experiments. One activity acts to criticise and correct the other. It is likely that differences between the model prediction and the experiment require you to re-visit and re-evaluate both the model and the experiment.

Finally, remember that it may not be the way the model has been set up that is wrong. The software itself may contain errors. All large computer codes, including well-established commercial codes with an extensive track record, are likely to contain errors and bugs. If possible, try to test the same problem on at least one other software package. If you believe that the software is wrong, don't hesitate to contact the supplier.

9 Getting support and help

9.1 Introduction

The observation concerning software errors at the end of chapter eight provides an important link to the topic of the current section on sources of help and support in finite element and finite difference modelling. Most commercial software vendors provide support services and access to help desk facilities with their software. Usually there is an annual charge or fee for access to this type of service, which may also include the right to receive regular upgrades of the software. For some of the large commercial finite element codes the cost of this facility can be several thousands of pounds every year. This is often seen by management and finance staff to be a large cost item that they believe may not be justified. However, it must be appreciated that advice and help from the supplier's programming and help desk staff may save many hours and even days of the time of scientific and technical staff who may otherwise have to try to find a solution to the problem on their own.

Failure to pay annual support fees is one of the main ways in which users find themselves with out-of-date versions of commercial software and thus unable to take advantage of new developments, including the correction of software errors and bugs that have been identified.

9.2 Other services provided by vendors

Commercial software suppliers typically provide a range of services in addition to help desk support and upgrades. Many run training courses in the use of their software, either at their own premises or on the premises of the purchaser. They may support user groups or encourage the formation of independent user groups and may also sponsor conferences where users present the results of work carried out with their software. Keeping in touch with other users of the same software is almost always of benefit. Such contact shows the range of problems that can be tackled with the software and the modelling challenges these present, and also develops links with individuals and organisations who may be able to provide advice and help when problems that cannot be resolved personally are encountered or when an alternative view of a modelling task is required.

9.3 Help from colleagues

In large scientific and engineering organisations it is often possible to find colleagues with experience of finite element and finite difference modelling, both with the software of interest and perhaps with other software packages. They are likely to have experience that can be of benefit, especially when the model building and coding need to be reviewed or it would be useful to compare the same problem with alternative software. In addition, colleagues may be able to provide an independent view of the quality and credibility of the modelling results even if they do not have direct experience of modelling themselves. Company quality schemes may also require the consultation of colleagues with the appropriate scientific and engineering knowledge as part of the model validation process.

Mathematicians and computer scientists on the staff are also an important source of advice, particularly in relation to validating and testing software. They should also be consulted during the early stages of specifying the hardware that will be required to run the software under consideration, as their advice in relation to the choice of hardware

platform, computer memory requirements, problem run times, and the possibility of accessing more computing power through parallel processing and distributed computing techniques may have important consequences for the choice of software package and software supplier. Company IT specialists will also give advice on maintaining data security and ensuring effective storage and back-up of data. In addition it is important to recognise that the speed of developments in computer technology means that applications that in the past had to be run on dedicated high-quality workstations can now be run effectively on PCs. The advantage of moving to PCs means that the user may be using a familiar operating system and computing environment, rather than being required to learn a new set of commands. Another practical question on which to seek advice from computing specialists is the export of data from the finite element software to other packages. This is especially important for graphical output where the user may wish to incorporate results into reports produced with standard word processing software.

9.4 Consultants and academics

One disadvantage of relying solely on the help desk and support teams of a software supplier means that there is a risk of being presented with only one point of view of the problem of interest. In addition, there may be a tendency to try to force the problem into a shape that the software can handle, rather than look for possible non-standard or imaginative approaches. Working with private consultants and with academic groups may be a means of overcoming this limitation. Even though this route may appear to be expensive at first sight, it may become economic if it saves a substantial amount of time otherwise spent on developing solution techniques or in trying to validate and test models. The main challenge in working in this way is that the problem of interest will have to be specified in an unambiguous manner so that all parties involved in the process know exactly what is required, in terms of the problem to be solved, the results that are required and how the results are to be tested.

9.5 Other sources of help

Finite element and finite difference modelling are mature areas of applied mathematics and the range of text books dealing with these topics is large and growing rapidly, especially as the techniques are applied in new areas. There are even texts on applications of finite element modelling in quantum mechanics. There are also a number of Masters courses in applied mathematics and in engineering and applied science disciplines which contain modules on finite element methods. In addition many academics publish course notes on the Internet, and the use of search engines will soon lead those interested to some useful resources.

Journals, including for example, *Finite Elements in Analysis and Design* published by Elsevier and *Finite Element News* published by Robinson and Associates, Okehampton, are a useful source of case study examples and of advice on validation and testing methodologies. The excellent series of NAFEMS booklets and the NAFEMS Primer have already been mentioned extensively throughout this guide.

Do not overlook the sample problems that are provided with much commercial software. These are often used by the software suppliers themselves as tests of the software, especially after upgrades have taken place. However, they can also be a useful guide to how to tackle a problem in particular application areas. It may be possible to begin by adapting an existing test example to resemble the modelling problem of interest to you.

9.6 Other software

Often it may be more efficient to separate out the stages of producing, solving and analysing a model and to perform each stage with dedicated software, rather than rely on one software package to perform all these stages. This is particularly the case when there is a need to take design data from an engineering design or drawing office as the basis of the model's geometry. The modelling process consists in such cases of four stages:

- the production of the design;
- the transfer of the geometry data to the pre-processing software, which then produces the finite element model;
- the analysis and solution software then solves the finite element model;
- finally the post-processor presents and displays the results.

In the extreme case, four different software packages may be needed. In practice, it is likely that the final two or three stages will be performed by a single commercial finite element software program and it will be the generation and transfer of the geometry data which may be performed by other software. Obviously, the interlinking of the various processes can only be carried out successfully if the data translation and linking is performed accurately at each stage.

Readers are referred to the NAFEMS Guide "How to choose a finite element pre- & post-processor" (1994) for a detailed discussion of this topic.

10 Future trends

Future trends for finite elements are being driven by two main requirements: the desire to tackle larger and more complicated problems, and the desire to model more complex materials.

As computer power has grown, finite element users have found they are able to run larger simulations more quickly than before. However, in many industries this does not mean that all problems can now be solved, it just means that the degree of approximation being used in many models of large-scale objects is reduced. Additionally, many problems that are physically on a small scale require detailed meshes that make them computationally intensive, and so there is still a need for more computational efficiency. This has produced two growth areas: parallelisation of code, and *hp*-adaptive meshes.

Parallelisation of code means that a computationally intensive problem can be distributed across several processors. Some packages offer this solely for multi-processor machines, and some offer a “distributed computing” version of their software that can distribute the computations over several machines connected to a network. At the moment, there are often restrictions on what type of problems can be solved using such codes, but as the technology becomes better established, the range of problems that can be distributed is likely to increase.

Adaptive *hp*-meshing is a technique where the software automatically alters its mesh (both geometry definition and element properties) so as to have detailed higher-order elements in areas where change in the calculated quantity is rapid, and less detailed elements away from these regions. It leads to accurate results with an appropriate level of detail. The mesh is refined either by dividing a mesh element up into geometrically similar sub-elements (*h*-refinement), or by increasing the order of polynomial approximation within an element (*p*-refinement). Some packages have already implemented these techniques, but as more research is done into efficient refinement and solution techniques, it is likely that their use will become more widespread.

Another growth area that is now possible due to the increase in computer power is coupled problems. More and more packages are offering “multiphysics” options, where users can couple together solutions of several different equations (e.g. stress analysis and thermal analysis, electromagnetic analysis and thermal analysis, etc.) to model several aspects of a problem simultaneously.

The most common material models (linear elasticity, Von Mises’ plasticity) were originally developed to describe the behaviour of metals. Advances in materials technology have led to a much wider range of materials being used in engineering situations. In particular plastics and composite materials are now strong, durable, and in common use for products that benefit from stress and impact testing at the design stage. For finite element modelling to produce reliable results, accurate materials models are needed. Additionally, finite element methods are now being applied to a wider range of application areas than before, and in particular their use in the biosciences is growing. All of these applications involve materials that do not behave in a “linear elastic-perfect von Mises’ plastic” way, and so if the finite element technique is to be used reliably in these areas some packages will have to widen their range of materials.

11 Survey of popular finite element and finite difference software

This chapter aims to summarize the range of commercial and non-commercial finite element and finite difference packages available, and to give a brief description of the most popular commercial packages. It draws on the results of the questionnaire circulated during the early stages of the research for the production of this report, the authors' experience of the packages, and web searches. Common general and application-specific finite element and finite difference packages are described, and reasons why finite element packages are much more common than finite difference codes are discussed.

11.1 General-purpose commercial finite element packages

The following is not intended to be an exhaustive list of commercial packages. The packages described were those commonly identified in the questionnaire results mentioned in chapter 1.2. All the packages are general-purpose finite element codes, but the additional features of several of them may mean that the package has a core market in a particular industry or application area (e.g. PAFEC has many features useful for acoustics problems). Where this seems to be the case, it is mentioned.

11.1.1 ABAQUS

(<http://www.hks.com/>)

The standard version of ABAQUS solves static and quasi-static problems, and a separate package (ABAQUS Explicit) solves non-linear dynamic problems using explicit time integration. Between them, the packages can be used for:

- Static and transient stress/displacement analysis which includes combined viscoelastic/viscoplastic response,
- Transient or steady-state heat transfer analysis,
- Transient or steady-state mass diffusion analysis,
- Coupled problems (thermo-mechanical, thermo-electrical, pore fluid flow-mechanical, stress-mass diffusion, piezoelectric, acoustic-mechanical),
- Eigenvalue buckling load prediction,
- Determination of natural modes and frequencies,
- Explicit dynamic response with or without adiabatic heating effects,
- Fully coupled transient dynamic temperature-displacement procedure,
- Annealing for multistep forming simulations,
- Acoustic and coupled acoustic-structural analyses for the simulation of wave propagation in an acoustic medium.

The package has a wide range of element types to cope with different types of loading and different levels of approximation, and can handle two dimensional, two-dimensional axisymmetric and fully three dimensional models. A large number of material models are supplied, many of which can include dependence of parameters on state properties such as temperature or strain rate. A wide range of boundary conditions

and surface interactions (e.g. contact between surfaces, heat radiation between surfaces, hinges, etc.) can be applied. In addition to its normal capabilities, users can define their own routines for material models, special elements, generalized loading conditions, and friction models.

11.1.2 ANSYS

[\(http://www.ansys.com/\)](http://www.ansys.com/)

ANSYS supplies a number of different finite element analysis products, each of which solves a different group of problems. These products are:

- Multiphysics: used for solving coupled combinations of structural, thermal, fluid flow, acoustic, and electromagnetic static and quasi-static problems. Combines tools from all the other products. Can also be coupled with LS-DYNA to solve dynamic problems.
- Mechanical: used for structural, thermal, and acoustic static and quasi-static simulations and some coupled problems. Can solve contact problems and cracking and crushing problems in which material may be destroyed.
- Structural: used solely for structural analysis. Can solve linear and non-linear static and quasi-static problems including contact and buckling.
- Emag: used for low-frequency electromagnetic calculations. Includes a circuit builder and calculation of derived quantities (e.g. force, torque, inductance, impedance, etc.).
- FLOTRAN: used for computational fluid dynamics simulations. Includes a range of fluid models, several different turbulence models, and natural or forced convection.
- Professional: contains “most requested set of solvers”. Aimed at finite element novices, and includes basic static structural and thermal analysis plus design sensitivity and optimisation capabilities.
- LS-DYNA: see LS-DYNA section.
- ED: introductory version of Multiphysics package with a maximum node limit.

Most of these packages have a wide range of element types and orders and numerous material models, although some packages offer a more restricted range, either due to the complexity of the underlying problem (e.g. Emag) or because the product is cheaper owing to a limited range of physical applications (e.g. ED, Professional). Most have capabilities for user-defined elements and subroutines, sensitivity analysis, and design optimisation.

11.1.3 FEMLAB

[\(http://www.uk.comsol.com/femlab/\)](http://www.uk.comsol.com/femlab/)

FEMLAB does finite element calculations within the Matlab environment. It solves problems, including coupled combinations, using the following equations:

- AC and DC electromagnetics,
- Convection-diffusion,
- Electrostatics magnetostatics,

- Heat transfer,
- Helmholtz's equation,
- Navier-Stokes equations,
- Schrödinger's equation,
- Solid structural mechanics,
- Structural mechanics, plane stress or plane strain,
- Wave equation.

Three toolboxes are available for specific application areas: Chemical Engineering, Electromagnetic, and Structural. These toolboxes use familiar terminology for the application areas concerned and extend the functionality in some areas (e.g. the Structural toolbox includes standard plastic material models).

The software has an automatic mesh generator that only uses triangular elements (tetrahedra in three dimensions), but can increase the order of elements to improve accuracy. Additionally it uses adaptive meshing during calculation to refine the mesh in areas where convergence is poor.

The links to the Matlab environment mean that material properties and boundary conditions can be defined in terms of Matlab functions of spatial position, time, temperature, or some other variable, which increases flexibility. Additionally, instead of specifying the application area of interest, the user can specify the partial differential equation (PDE) that they wish to solve, by supplying coefficients for use in the most generalised form of a second-order pde. This capability can be used repeatedly to define a set of coupled generalized PDEs. The Matlab environment can also be used to set up repeated simulations for design sensitivity analyses and optimisations.

The flexibility and general approach to problem definition in FEMLAB mean it is used widely in academia.

11.1.4 LS-DYNA

(<http://www.lstc.com/>)

LS-DYNA is a general-purpose non-linear dynamics code. It is a popular package for sheet-metal forming problems and for impact simulation (particularly in the automotive and aerospace industries). The code was originally developed to carry out stress analysis under impact loading, but the application area has now been broadened out to cover solids, fluids and interactions and heat transfer between them.

The package has over 130 materials models, including biomechanical materials, geological materials, various composite models, and a user-defined material option. It also provides a wide variety of elements, including specialized seatbelt elements for vehicle occupant safety tests, and a variety of different joints and contact mechanisms.

11.1.5 LUSAS

(<http://www.lusas.com/>)

LUSAS offer four main products, all of which can have their functionality extended by addition of options adding non-linear analysis (covering both material and geometric non-linearities), dynamic analysis, extra modal dynamics capabilities, and thermal modelling to give a broader range of tools. The main products are:

- Bridge: used to analyse bridge structures under a wide variety of loading conditions the structure could experience in reality.
- Civil and Structural: used for structural analysis including analysis of behaviour of structures such as buildings, dams, and tunnels under extreme conditions such as seismic, blast, buckling, impact and fire.
- Composite: used for composite design and analysis, including a range of failure criteria.
- Analyst: used for general engineering analysis, including non-linear, contact, impact, and thermal problems.

Although LUSAS is a general finite element package, many of its special features are useful for civil and structural engineering.

11.1.6 NASTRAN

(<http://www.mssoftware.com/products/>)

The basic version of NASTRAN solves linear static, buckling, and eigenmode problems (including acoustic problems), and various additional options can be added to the standard package to enable non-linear, thermal, and transient analyses. These are:

- Non-linear Statics: used for geometrical and material non-linearities,
- Non-linear Transient: used for non-linear dynamic problems,
- Aeroelasticity: used for analysing structures in an air stream (includes subsonic and supersonic options),
- Superelements: used for interaction of objects where the detailed behaviour of each object does not affect the overall interaction,
- Optimization: used for sensitivity analysis and design optimisation,
- Heat Transfer: used for thermal problems, including convection,
- DMAP: programming language that can be used to add functionality by automating particular sequences of commands to carry out analyses the user runs on a regular basis.

Several of NASTRAN's additional features suggest it is used for aerodynamic design problems.

11.1.7 PAFEC

(<http://www.vibroacoustics.co.uk/feusers.htm>)

PAFEC has a wide range of element types including boundary element capabilities for some problems, and can solve problems in linear statics and dynamics (including crack tip analysis), acoustics, heat transfer, lubrication, magnetics, and electrochemicals. The package has various options for user-defined files (e.g. for boundary and initial condition definition). Material models are fairly limited: only linear elastic, a small range of plastic models, and two hyper-elastic models are provided for stress analysis.

The new owners of the IPR to PAFEC, PACSYS Ltd., are developing its acoustic capabilities, and its core application areas are underwater acoustics, audio, ultrasonics, and automotive noise (<http://www.vibroacoustics.co.uk/>).

11.2 Other commercial packages

This list aims to summarise other commercial packages available, particularly where they are strong in one particular problem area. These are not packages that were frequently mentioned in the questionnaire responses, but they may potentially be of interest. Many packages have limited-size trial versions that can be downloaded for free.

11.2.1 General packages

ADINA (<http://www.adina.com/>)

ALGOR (<http://www.algor.com/>)

JL Analyzer (<http://www.autofea.com/>)

CAEFEM (<http://www.caefem.com/>)

I-DEAS (<http://www.eds.com/products/plm/ideas/>)

NISA (<http://www.emrc.com/>)

PAM (<http://www.esi-group.com/products/index.php>)

Fastflo (<http://www.cmis.csiro.au/Fastflo/>) (generalized finite element PDE solver, distributed by the Numerical Algorithms Group in the UK)

StressCheck (<http://www.esrd.com/products/default.asp>)

COSAR (http://www.femcos.de/femcos_e.html)

PERMAS (<http://www.intes.de/>)

LISA (<http://www.lisa-fet.com/indexbut.htm>)

MARC (<http://www.marc.com/Products/default.htm>)

Diffpack (<http://www.nobjects.com/>) (generalized finite element PDE solver)

FlexPDE (<http://www.PDEsolutions.com/>)

ProMECHANICA (<http://www.ptc.com/products/proe/sim/index.htm>)

FEVA (<http://www.3dfeva.com/products.htm>)

ELFEN (<http://rsazure.swan.ac.uk/elfen.htm>)

SAMCEF (<http://www.samtech.fr/>)

SOLVIA (<http://www.solvias.se/>)

AFEMS (<http://www.femengineering.com/index2.html>)

Strand7 (<http://www.iuk.co.uk/>)

AutoDesign (<http://www.sat-i.com/products.htm>)

COSMOS (<http://www.cosmosm.com/>)

QuickField (<http://www.quickfield.com/feat.htm>)

11.2.2 Packages with a particular strength

SLIM:

(<http://www.technology.alstom.com/en/rdpolicy/consultancy/slim/docs/index.htm>)

Electromagnetics

Hyperform (<http://www.altair.com/03520000.htm>) Sheet metal forming

Optistruct (<http://www.altair.com/035a0000.htm>) Design optimisation

HFSS (<http://www.ansoft.com/products/hf/hfss/>) High frequency electromagnetics

STRAP (<http://www.atir.com/aboutstrap.html>) Structural analysis, especially civil engineering

COBUS (<http://ourworld.compuserve.com/homepages/cobus/frames.htm>) Linear stress analysis

DIANA (<http://www.diana.nl/>) Civil engineering

DIE (<http://www.die.de/>) Statics. German only.

DrBeam, DrFrame (<http://www.drsoftware-home.com/>) Beam and frame modelling

TriComp and Maze (<http://www.fieldp.com/>) Electromagnetics, shock and detonation modelling

FinitES-Plast (http://www.finites.de/Engl/about_engl.htm) Metal forming and pressing

Geoslope (<http://www.geo-slope.com/products/>) Geotechnical modelling

AxisVM (<http://www.axisvm.com/>) Civil engineering applications

Infolytica (<http://www.infolytica.com/en/products/>) Electromagnetic and thermal analyses

FEWaves (<http://www.mstrtech.com/>) Structural acoustics

Zebulon (<http://www.nwnumerics.com/Zebulon/>) Material behaviour, with many user-defined options

PLAXIS (<http://www.plaxis.nl/ie.html>) Geotechnical calculations

RADIOSS (<http://www.radioss.com/>) Crashworthiness and impact simulation

Robot (<http://robot-structures.com/uk/>) Structural analysis

GCM (<http://rsazure.swan.ac.uk/gcm.htm>) Glass container modelling

SAGECRISP (<http://www.crispconsortium.com/program.html>) Geotechnical modelling

SDTools (<http://www.sdtools.com/>) Structural dynamics Matlab toolbox

Nolian (<http://www.softing.it/eng/pr/ms/macsap.html>) Structural analysis

S-Frame (<http://www.s-frame.info/spframe.htm>) Structural analysis

Soilvision (<http://www.soilvision.com/products.shtml>) Soil seepage, contaminant transport, and thermal modelling

FEA-Crack (<http://www.srt-boulder.com/commercialsoftware.htm>) Crack modelling

FEMForm (<http://www.femengineering.com/index3.html>) Metal forming

GTSTRUDL (<http://www.gtstrudl.gatech.edu/>) Structural design

ThermoAnalytics (<http://www.thermoanalytics.com/products/index.html>) Thermal modelling

AdvantEdge (<http://www.thirdwavesys.com/>) Metal cutting and processing

Vector Fields (<http://www.vectorfields.co.uk/>) Electromagnetic simulation

Genesis (<http://www.vma.com/software.htm>) Design and optimisation

ZSoil (<http://www.zace.com/zsoil.htm>) Geotechnical calculations

11.3 Freeware and shareware

There are many freeware and shareware FE packages available: for lists, see the links mentioned below. Most of these packages do not have high-quality graphical user interfaces, but they offer increased flexibility and source code availability for the user. Often they are of more use to experienced users who know that their usual code cannot cope with the problem they are trying to solve rather than inexperienced users who wish to solve relatively routine problems.

11.4 Finite difference packages

Finite difference packages are comparatively rare, and where they do exist they tend to be application-specific. Finite difference methods are still used extensively in academia and research, but methods tend to be developed to solve a particular problem rather than being sufficiently general to solve any partial differential equation problem. The main reason why finite elements are more widely used is that finite difference schemes generally need to have a regular rectilinear grid of node points. This causes problems when a user wishes to model a structure that is not a rectangle, sphere or cylinder. These problems can be avoided to some extent using remeshing and multigrid techniques, but these techniques add an extra level of complexity. Finite elements can be generated to fill any shape since they do not need uniform grid spacings.

The NAG library (<http://www.nag.co.uk>) has a wide range of routines for solutions of partial differential equations in the D03 chapter, several of which have automatic remeshing to improve accuracy. Netlib has a number of libraries for solution of different partial differential equations, including <http://www.netlib.org/fishpack/> for elliptic boundary value problems on a rectangle, <http://www.netlib.org/pltmg/> for elliptic equations in more general regions, and <http://www.netlib.org/PDEs/> for other types of problem (not all of which are finite difference based). The sources also provide the basic units needed for a user to develop their own partial differential equation solver, as both have extensive collections of solvers for large sparse systems of linear and non-linear equations. There are various applications where finite difference methods are more commonly used, particularly electromagnetics (in the form of finite difference time domain methods), and computational fluid dynamics (particularly for porous flow problems).

11.4.1 Particular finite difference packages

MODFLOW groundwater flow

http://www.scisoftware.com/products/modflow_details/modflow_details.html

UNIVDIF financial modelling <http://www.mbrm.com/htmls/ufda.html>

FIDSOL/CADSOL general packages,

<http://www.uni-karlsruhe.de/Uni/RZ/Forschung/Numerik/fidisol.cadsol/>

SMS surface water modelling <http://www.flowpath.com/software/sms/sms.html>

MCTMIG depth migration, <http://www.3dgeo.com/mctmig.html>

MUDPACK multigrid for elliptic PDEs,

<http://www.scd.ucar.edu/css/software/mudpack/>

FLDWAV flood routing, <http://www.nws.noaa.gov/oh/hrl/rvrmech/fldwav1.htm>

SINDA/FLUINT thermal and fluid, <http://www.crtech.com/sinda.html>

XFDTD electromagnetics, <http://www.remcom.com/html/index.html>

EMPIRE electromagnetics, <http://www.empire.de/>

11.5 Other resources

As well as finite element solvers, there is a wide range of pre- and post-processors available to develop finite element meshes and to visualise and interpret results of analyses. No attempt has been made to survey these, since most commercial packages are supplied with their own pre- and post-processors. Additionally, several companies offer libraries of components for development of finite element software (e.g. mesh generators, matrix solvers, etc.) so that users can custom-build their own applications. Further information on such resources is available from the links below.

11.6 Useful links

http://www.engr.usask.ca/~macphed/finite/fe_resources/fe_resources.html contains an extensive list of finite element Internet resources, including links to freeware and shareware sites, links to commercial code sellers' sites, and links to more general information about finite element theory and methods. Each link is accompanied by a short description of the material the site contains.

<http://sal.kachinatech.com/B/2/> contains links to sites with commercial, shareware, or freeware finite element and finite difference codes.

http://dmoz.org/Science/Technology/Software_for_Engineering/Finite_Element_Analysis/ has a list of finite element analysis links including shareware and freeware.

<http://home.worldonline.nl/~fenned/xfemi.html> (the Robinson FEM Institute) has information about finite element modelling courses run by the Institute and links to other finite element resources.

<http://www.nafems.org/>: the website of the National Agency for Finite Element Methods and Standards, quickly shortened to the acronym NAFEMS, was founded as a special interest group in 1983 with a specific objective "to promote the safe and reliable use of finite element and related technology".

12 Users' testimonies

This chapter is made up of contributions from a variety of metrology users of finite element packages. They were asked to describe their experiences of the packages they use, and in particular to address the following points:

- Name and version number of package,
- Computer system (i.e. PC/Workstation, etc),
- Broad description of application area,
- Any special features that led them to choose this package,
- Good points and bad points they have found in using the package,
- Anything else that may be of interest to prospective users of the package.

In some cases a detailed description of the operation of a package or of the problem being examined has been included. This detail has been left in as the issues being described are likely to be of interest to users with similar problems, and the details of operation may be of interest to those currently using other packages. The text of the individual contributions has not been edited, except where it has been necessary remove repetition and to edit references to specific organisations to retain anonymity.

12.1 ABAQUS

Supplier

ABAQUS UK Ltd, Genesis Centre, Warrington WA3 7BH

ABAQUS User Testimony 1

Version

ABAQUS Standard and ABAQUS Explicit version 5.8-1

Platform

Sun workstation accessed remotely from a PC

Applications

Stress analysis of nonlinear materials including material model development

General comments on package

ABAQUS has been used as a tool for a wide variety of stress analysis problems involving linear and non-linear materials. ABAQUS has a wide variety of material models, including many of the common plasticity and hyper-elasticity models. Often model parameters can be functions of solution variables such as temperature or strain rate. Boundary condition input is generally very flexible, and there is a range of element types that are suitable for different conditions that may occur.

In addition to its existing material models, ABAQUS has the capability for including user-defined materials. This has been used recently to produce a modified version of the Gurson model (Gurson, 1977) for porous metal plasticity with some success. The user develops a FORTRAN code that calculates the stress increment for a given strain

increment, along with an approximate expression for the Jacobian if the model is to be used with the Standard code.

ABAQUS has two main solution codes: Standard for static and quasi-static problems, including vibrational analysis and heat transfer, and Explicit for modelling problems where inertial effects are likely to be significant. Explicit has a reduced range of material models and tends to take longer to run than Standard.

Usability comments

The format of the ABAQUS input file makes it straightforward and intuitive to set jobs up, and the manual explains the requirements of the various sections of the file clearly. The flexibility in terms of element type, boundary condition description, and material models makes it possible to model a variety of problems.

The user-defined material option is a useful feature. It is not particularly user-friendly, but it should not be used by novice users anyway so that is not really a problem. Other features such as boundary conditions and element formulations can be defined through user routines as well, although we have not tried these. User routines do add to the running time, but this is bound to happen as the code will not have been optimised to the same extent as that of the models supplied with Standard and Explicit.

The output of results can be tailored to give files of a reasonable size that contain all the information required about a model, particularly for a large model where only a subset of elements is of interest. Results can be written directly to a text file, which can be useful if operating the software remotely with only a text link available.

The post-processing packages that come with ABAQUS (Post and Viewer) are largely user-friendly and well-documented. It is simple to produce graphical output and contour plots, and Post produces text files of results easily. The contours can be switched to display discontinuous element-by-element values instead of smooth curves, which can be useful for troubleshooting etc.

There have been some problems with convergence of some of the problems we have attempted to model. In particular, we have experienced problems with the Drucker-Prager plasticity models. Some jobs have not been able to complete a solution using the Standard solver, but have run satisfactorily with the Explicit solver. Additionally, there have been some problems using strain rate dependent data. Generally, we gather data at test speeds that produce strain rates that differ by orders of magnitude. If these are used in the Drucker-Prager models, the interpolation between different curves does not seem to produce the expected results. It is not easy to know in advance which sets of parameters are likely to be problematic, which can be inconvenient.

ABAQUS User Testimony 2

Version

ABAQUS Standard and ABAQUS Explicit version 6.x

Platform

Sun workstation accessed remotely from a PC

Applications

Stress analysis of nonlinear materials and geometries

General comments

A version of Python (interpreted scripting language) is embedded within ABAQUS to provide an interface to ABAQUS data. This allows automation of repetitive tasks and also extended functionality such as enhanced visualisation or performing calculations outside the scope of ABAQUS.

In post-processing, common tasks can be automated by using journal (.jnl) files (ABAQUS/post) or Python script (.py) files (ABAQUS/viewer). This can speed up repetitive tasks. In V5.8 the results file (.res) is used for post-processing, while in V6.x it is the output database (.odb) file.

Usability comments

The latest version (V6.3) comes with CAE, which contains a pre-processor as well as FE solver and post-processor. This has a form based graphical user interface that will reduce the need to modify the input file directly. The pre-processor is relatively new and so lacks some of the functionality of well-developed pre-processors. A big advantage is that it is tailored to ABAQUS and is kept updated with any ABAQUS changes in each release. There would always be an uptake time lag with other pre-processors.

The automation of post-processing tasks means results from repeated model runs such as those produced by sensitivity analysis can be processed quickly and efficiently for inter-comparison.

A few years ago, the release of V6.1 saw significant architectural changes. This has caused some incompatibility between V5.8 and V6.x input files, although any problems are highlighted during data-checking. A V5.8 file can be updated to the V6.x format easily. The results files from the two versions are not compatible so the V6.x results files (.res) cannot be used in Post and the V5.8 output database results file (.odb) cannot be used with V6.x Viewer. Currently, to be able to use Viewer on a PC Hummingbird Exceed V6.2 (an old version) and Exceed 3D are required.

Extensive manuals are available in print or online. The UK customer support is generally good – in most cases you get a fast response.

12.2 ANSYS

Suppliers

ASN Systems Ltd, 1st floor, Waterloo House, Riseley Business Park, Riseley, Berkshire RG7 1NW

Analysis Design Productivity Ltd, PO Box 2993, Kenilworth, Warwickshire CV8 1XZ

IDAC Limited, Airport House, Purley Way, Croydon, Surrey CR0 0XZ

Wilde FEA Ltd, Brindley Lodge, Adcroft Street, Stockport, Cheshire SK1 3HS

ANSYS User Testimony 1

Version

ANSYS 5.4 Mechanical/FLOTRAN

Platform

PC, Windows based. The work described here was performed in a Windows NT environment.

Application

Calculating the pressure distortion coefficients of piston-cylinder pressure balances.

The problem

Oil-operated pressure balances consist of a piston located within a closely fitting cylinder. The radius of the piston is typically a few mm and the external radius of the cylinder may be several times that of the piston. The clearance between the piston and cylinder is 1 μm or less. The gap between the piston and cylinder is filled with oil, which flows along the gap under the influence of a pressure gradient. This pressure gradient arises because the lower end of the piston and cylinder assembly is connected to a pressurised reservoir of oil (in practice this may be another pressure balance) and the top of the assembly is exposed either to the air or to a vacuum. The force on the piston generated by the application of the pressurised fluid is balanced by a mass applied to the top of the piston.

The calculation of the pressure generated by a piston-cylinder pressure balance or deadweight tester requires knowledge of both the forces applied to the piston and the area over which they act. However, the piston and cylinder assembly is itself distorted by the pressure generated in the pressure balance fluid, and accurate determination of the generated pressure requires this distortion to be taken into account. In other words, it is necessary to know the effective area over which the force is applied in practice.

We use ANSYS to predict the pressure distribution along the engagement length of the piston and cylinder, and consequently the local distortion of the piston and cylinder components, and then derives the distortion coefficient by using the outputs of the modelling as inputs to Dadson et al's equations (1982) and other related equations. Currently, we employ an uncoupled, iterative solution technique, so that the fluid-flow finite element problem is solved to derive a pressure distribution, which is then used as the stress input to the finite element calculation of the piston and cylinder strain. The distortion results are then utilised in the next iteration of the fluid-flow calculation and this process is repeated until convergence is achieved. See Esward (2002) for a fuller description of the problem formulation.

Setting up the problem in ANSYS

ANSYS provides the facility to define and carry out analyses using the ANSYS Parametric Design Language (APDL). This high-level language has a FORTRAN-like structure and allows the user to define and re-define parameters, to generate macros, initiate solution routines, and write macros to analyse results, in the same manner as one writes a typical piece of line-by-line computer code. A suite of macros in APDL has been developed to handle all pre-processing, problem solution, and post-processing aspects of the finite element analysis. The method has been standardised so that it is

only necessary to produce a single, new pre-processing macro to model a new pressure balance or to investigate a new pressure balance geometry.

In the case of pressure-balance modelling, the finite element analysis provides the raw data needed for the solution of equations given in Dadson et al (1982). The solution method is an iterative one, as the fluid flow problem and the structural strain problem are solved in an uncoupled manner. Knowledge of the structural deformation is simply a means towards the end of calculating the distortion coefficient and the piston fall-rate (the piston moves downward as fluid leaks out at the top if the engagement length), and such calculations are naturally not part of standard finite element software routines.

The use of APDL allows us to prepare a series of macros to carry out each stage of the analysis. The solution and post-processing stages are common to all the finite element analyses we carry out. To prepare a new model of a pressure balance it is only necessary to generate new pre-processing macros. In practice, the pre-processing macros are all very similar and the differences between balances relate only to geometry, materials properties and boundary conditions. The structure of the macros themselves remains the same from one analysis to another.

The pre-processing stages set up the global variables for the analysis in question - materials properties, co-ordinates of key points, and line pressure value, for example - and turns these data into a meshed model with all the elements and boundary conditions defined. As part of this stage ANSYS carries out a number of important checks to ensure that key features of the modelling are valid. It performs geometry validation, including checking that areas and volumes are closed, that the element types are consistent with the required analysis, and warnings are produced if aspect ratios of elements are unreasonable.

For the models of piston-cylinder pressure balances, it is difficult to separate the solution and post-processing stages as the finite element problem is solved many times, and at each cycle of the iteration some mathematical analysis is carried out in what are technically post-processing routines to define the input for the next cycle of the iteration. This process continues until the model being analysed converges to a solution. The mathematical analysis routines, which are written by us and not part of the ANSYS software package, are technically post-processing routines in finite element terms.

To model the fluid flow and therefore the pressure distribution along the gap between the piston and cylinder we employ an analogy between fluid flow and thermal flow, which allows us to use thermal finite elements within our ANSYS model. For steady-state viscous laminar flow between two plates whose length and width are much greater than the distance separating them, the total fluid flow across a unit width perpendicular to the flow direction is equal to the rate of decrease of pressure in that direction multiplied by the constant $h^3/12\mu$, where h is the gap width and μ is the coefficient of viscosity. In the case of one-dimensional heat flow, the heat flux is equal to the rate of decrease of temperature in the direction of interest multiplied by KA , where K is the thermal conductivity and A is the cross-sectional area. This means that a one-dimensional thermal analysis can be used for the fluid model.

The LINK32 ANSYS finite element is a uniaxial conducting bar element with a single degree of freedom, temperature, at each node point and can be used for plane or axisymmetric analyses. For steady-state solutions, density and specific heat of the bar material are ignored, so that only the thermal conductivity and the cross-sectional area of the element are required. Thus this element can be used to calculate pressure

distributions in the case under consideration provided the thermal conductivity value is replaced by the appropriate value for the fluid flow problem. The “thermal” finite element problem is solved independently of the structural FE problem and its results are used as the input to the next iteration of the structural problem. The solution of the structural finite element analysis then provides the information on the gap width, and therefore area, needed for the next iteration of the “thermal” analysis.

The post-processing calculations include calculation of distortion coefficient and pressure loads for the next iteration, and calculation of other quantities of interest such as the rate at which the piston falls as a result of fluid flow. The pressure and displacement results are written to a file during this stage of each iteration, as are all calculated results such as the effective area, the distortion coefficient, and the fall rate. Graphical displays of results and contour plots of pressure, stress, and displacement can also be generated.

As it may take 100 iterations or more and several hours for an analysis to converge to a solution, it is important to retain as many results of the analysis as possible to avoid the need to repeat calculations. The use of text files to store results means that they can be examined at any point in the future without having to rerun the full job. It also allows data to be imported into other software packages such as Excel or Matlab, thus facilitating alternative methods of data analysis, including the production of graphical output. Another benefit is that it renders the user independent of ANSYS’s own graphics capabilities, if one wishes to prepare plots of pressure distributions or gap profiles.

Usability comments

As can be seen from the above description, pressure balance modelling uses a limited subset of the capabilities of ANSYS, in this case the solution of a linear static stress problem and of a one-dimensional heat flow problem. In addition, almost all the mathematics required to produce a solution is carried out in post-processing routines we have written. Here, the flexibility of ANSYS and APDL and the wide range of APDL mathematical commands available allowed the evaluation of the results of differentiation and integration operations to be carried out quickly and efficiently, often with only a single line of APDL code.

APDL is also well suited to the use of a modular approach to model building. It is relatively simple to develop self-contained macros for each stage of the analysis from pre-processing to post-processing and to test these independently of each other. This method of operation is very effective when, as in the case of the work described here, one is modelling a class of structures, all of which require the same mathematical analysis and which differ from each other only in the details of geometry and materials properties.

The ability to generate plain text files listing modelling results such as displacements of nodes and the coordinates of nodes of interest allowed ANSYS results to be exported easily to other software packages. This had a specific advantage in a particular case – the display of the distorted shape of the piston and cylinder under pressure loading. It was observed that when ANSYS was asked to display the distorted piston and cylinder pressure balance structure, interference between the cylinder and the piston could be occasionally observed in the image at the top (low pressure end) of the engagement length. In fact, although the piston and cylinder were often only 10s of nanometres apart at this point, no contact between the two components was taking place, and this was

clear from an inspection of the nodal distortion data. It may be that for gaps of this size, the ANSYS graphics were incapable of resolving such small differences in geometry. Graphs produced from the same numerical data using Matlab and Excel represented the gap correctly.

12.3 FEAT

Supplier

Serco Assurance B150 Harwell, Didcot, Oxfordshire OX11 0QJ

FEAT User Testimony 1

Version

FEAT 3.5.1

Platform

Unix or Linux

General use

FEAT is used for solving thermal problems, including those involving CFD, to obtain plant component temperatures both for normal operation and during fault transients. The software was originally maintained and developed by British Energy, but it was out-sourced just over three years ago to Serco Assurance. FEAT also has a concurrent stress analysis capability, for undertaking elastic and plastic analysis. This is utilised during the transient analysis of components in which thermal stresses need to be monitored to find the time and state of the most onerous loading. It also has a capability for weld modelling, for which temperatures and stresses are calculated at each time-step. FEAT has mesh generation, pre-processing, solver and post-processing capabilities that are accessed through FEAT's interactive command language interface.

Features used

Mildly compressible CFD (density variation with temperature), on flows strongly influenced by buoyancy and walls; in particular the low-Reynolds number k-ε turbulence model in conjunction with sophisticated wall functions. We use FEAT for CFD mainly because of the validation that FEAT has for this type of flow. However another reason is that its discretisation methods are high order (formally fourth order), and the solutions obtained are accurate and can be trusted to be broadly free from mesh error. Alongside the CFD, FEAT has a terrific capability for analysing temperatures, both steady state and transient, in solid materials. For example, it has a full range of boundary conditions, property variation with temperature, thermal radiation using a ray-tracing technique, including the effects of a participating gas, and a flexible approach which enables the user to always solve the problem that needs to be solved. It is also used to provide predictions of weight loss in our graphite cores. This involves the solution of transport equations for various chemical species, including their interaction, in conjunction with a flow model as gas passes through the pores of the graphite bricks. Finally, we also use it for weld modelling, to predict residual stresses.

Advantages of software

The command language interface is extremely flexible, and enables the user to solve the

problem that needs to be solved in a highly efficient manner. The quality of the software is first class, as measured by comprehensive verification, validation and benchmark guides, which form the basis of the test set.

Usability issues

FEAT's command language is both an advantage and a disadvantage. For those able and willing to get their hands dirty it is an extremely powerful tool. However, for the occasional user, it is likely to prove to be too difficult to use because of the learning curve. The frequency of reported bugs is very low. The technical support is excellent, provided by very knowledgeable people.

FEAT User Testimony 2

Version

FEAT 3.5.1

General information

We have experience in using the FEAT package for over 15 years to perform assessments requiring various types of flow, thermal and structural analyses. FEAT is unusual in that it is operated interactively as a single package, so that no separate pre- and post-processing facilities are required. In addition, different types of calculation and data manipulation, e.g. to perform thermal and stress analyses, can be carried out within a single FEAT session. This gives users great flexibility over the way they solve particular problems

Features used

In recent years many of the analyses using FEAT have been in support of structural integrity assessments. In its CFD mode, the code has been used to calculate the thermal loading of structures. The flows involved are often affected by buoyancy, requiring the use of solution routines, incorporated within FEAT, for the analysis of turbulent buoyancy-influenced flows. Other features used in CFD calculations have included modelling of flow through porous media (e.g. tube banks), and combined modes of heat transfer (e.g. conjugate heat transfer, and the inclusion of thermal radiation using the inbuilt ray-tracing algorithm). Thermal calculations have often involved the implementation of empirical correlations, some of which have been of significant complexity (e.g. when modelling boiling heat transfer or representation of buoyancy-driven thermo-syphon flows). Structural analyses have included linear elastic and elastic/plastic behaviour. In some of these, the mesh generation and manipulation facilities in FEAT have been used to generate locally refined meshes to represent small defects, these meshes then being embedded in larger, less refined meshes of the overall structure.

Advantages of the software

FEAT is driven using a command language, the commands usually being combined together to form macros. Model development and solution is carried out interactively, and this has been found to result in very cost-effective use of engineers' time and rapid model production. The ability to interrogate any aspect of the model either graphically or numerically throughout model build and solution has been of considerable assistance, particularly when developing models to represent several complex processes. Features such as the inbuilt mesh error analyser and the ability to interpolate between meshes of

different sizes have also proved useful. The latter feature has been used when performing both thermal and structural calculations on the same component as a thermal mesh is not generally optimised for structural purposes. The ability, within a single FEAT session, to interpolate temperatures onto a mesh that is able to resolve high local stress concentrations has been invaluable.

Usability issues

Like any language, the high-level FEAT command language needs to be learned, but it provides far greater flexibility than a menu-based system and it avoids the need to code user subroutines in FORTRAN or C++. The command language has shown itself to be powerful and versatile enough to enable users to perform operations not originally envisaged by the code developers, such as the mesh manipulation facilities noted above. Although mainly used on Unix systems, FEAT can also be run on PCs using Linux.

FEAT User Testimony 3

Version

FEAT 3.5.1

Platform

FEAT can currently run on Unix workstations and Linux based PCs.

General use

FEAT is used mainly for thermal modelling. Problems solved include the modelling the temperature of items of plant on nuclear reactors, modelling the temperature reached by radioactive material transport containers under normal conditions and when exposed to a fire and modelling the temperature inside drums of radioactive waste during a fire. These assessments have included the use of the structural capability within FEAT to determine stresses and deformation due to the temperature gradients in the models. FEAT has built-in mesh generation capabilities which we frequently use. However, for complex, 3-D models we prefer to use the FEMGV mesh generation package.

The fluid flow capability within FEAT is only occasionally used within our company. For determining flow in complex geometries we prefer to use the CFD code CFX.

Advantages of FEAT

The advantages of FEAT are its power and flexibility. The solvers (there are a choice) are extremely fast and hence very suitable to large 3-D problems. The flexibility of FEAT is apparent in pre-processing, the solution stage and post-processing. In pre-processing, FEAT can not only import meshes from a range of pre-processing packages but can also manipulate the imported meshes, joining different meshes together, extending 2-D meshes into 3-D, transforming and copying meshes and deleting regions. These facilities are also useful for preparing meshes for use in other FE packages. During the solution stage, the flexibility of FEAT is frequently used to add additional physics. A simple example is the representation of natural convection in an enclosed space using an engineering correlation. To determine the appropriate heat transfer coefficient at each surface of the enclosure both the area averaged surface temperature and the temperature range need to be determined. This is easily coded into a FEAT macro using the command language.

During post-processing, FEAT allows results to be determined at locations not represented by nodes in the model. The entire solution can also be interpolated onto a completely separate mesh (optimised for structural performance for example) or a 2-dimensional model, with results, to be extracted from a 3-dimensional model. Additional scalars can be readily defined if required.

Usability issues:

FEAT is driven by macros that are most readily generated using a text editor (although each line can be tested interactively by copying as a FEAT command). The format of the FEAT command language is best learnt on a training course and, due to the flexibility of FEAT, care needs to be taken to avoid mistakes. FEAT is very well validated and verified. Help is readily available from the help desk run by Serco Assurance.

12.4 FEMLAB

Supplier

COMSOL Ltd, John Eccles House, Oxford Science Park, Oxford OX4 4GP

FEMLAB User Testimony 1

Version

Femlab 2.3, running under Matlab version 6.5, networked PC Windows-based version

Hardware

Windows PC

General use

FEMLAB can be employed with Matlab within the Windows, Unix, Linux and Macintosh environments. FEMLAB is a finite element solver which is used with Matlab and the Matlab toolboxes. However, version 3 which was released in December 2003 can be run as stand-alone software independent of Matlab. An interactive menu-driven interface allows the user to build finite element models either by selecting ready-to-use application modes, by choosing the physical situations and equations of interest, or by writing the equations of interest directly. It is possible to combine both approaches in the same model. This has advantages for multiphysics modelling, which is one of the main strengths of the software. FEMLAB has automatic mesh generation and refinement facilities. Its main limitation is that it only allows the use of triangle (2-D) or tetrahedral (3-D) elements. It has specialist toolboxes for chemical engineering, electromagnetic and structural mechanics problems. Physical properties can be defined as analytical expressions or Matlab functions. FEMLAB can import DXF and IGES files. It also has extensive model libraries and example problems. Training courses run by Comsol Ltd (<http://www.uk.comsol.com/>) are readily available.

The ready-to-use physics applications within the basic FEMLAB software include (depending on the number of dimensions in the model) AC power electromagnetics, DC conductive media, diffusion, electrostatics, magnetostatics, heat transfer, the incompressible form of the Navier-Stokes equation, and structural mechanics. The

ready-to-use classical partial differential equations are Laplace's equation, Poisson's equation, the Helmholtz equation, Schrödinger's equation and the convection-diffusion equation. Additional functionality is available with the three specialist toolboxes. For example, the chemical engineering tool box adds momentum, energy and heat balances to the basic physics models.

Graphical user interface

The model is built up through FEMLAB's graphical user interface (GUI) which takes the user through a series of menus which allow the model to be developed in a structured manner. At the start of the modelling process the user enters the Model Navigator, which allows the dimensionality of the model, the application mode and the element type to be selected. Subsequent menus allow the user to draw the model structure. The drawing process is flexible and intuitive. Components can be drawn separately and then joined into one component, for example. For simple geometric structures such as cylinders the user simply enters a radius and a height and the software produces the three-dimensional geometry immediately. Further menus allow the definition of boundary conditions, initial conditions, material properties, equation coefficients, and so on. The default mesh is drawn by accessing a further menu and the mesh can be further refined. Although the user does not have total control over the meshing process, it is possible to choose appropriate mesh densities, including local variations in density, by skilful use of the meshing menu. Finally, the user defines the required solution method and in the case of transient problems, the required time step.

Post-processing is performed using the GUI and a variety of graphs and plots, including contours, are available. Numerical data are often best obtained using Matlab itself. The FEMLAB model can be stored as a Matlab .mat file or as an m-file script (for more information about Matlab itself (see <http://www.mathworks.com/>) and it is these files that allow the user to process the results with the greatest flexibility. In such cases the user with prior experience of Matlab has a substantial advantage.

Manual

The FEMLAB manuals contain a range of practical examples of the use of FEMLAB to solve specific problems in the basic version of FEMLAB and the three main toolboxes. These examples are often based on industrial and scientific problems which FEMLAB users have themselves tackled with the software. As a result the examples can be relatively complicated. In addition, not all aspects of FEMLAB's potential uses are covered. What is lacking is a set of simple examples which demonstrate some straightforward problems from first principles, which the user can begin to edit and develop to gain experience.

Advantages and disadvantages

FEMLAB's single most important advantage is that it is possible to develop complicated models quickly through the use of the GUI and the drop-down menus. This ease of use can also be a disadvantage in that it is easy to produce models and obtain rapid results, but one needs to check carefully that one has implemented correctly the physical problem and solution method of interest. Have the correct choices been made for all the menu options? Have the default FEMLAB options been accepted without careful consideration of whether they might be appropriate for the model in question? It is also necessary to take into account the limitations of the basic FEMLAB implementation. For example, axisymmetric modelling of structures appears only to be possible in the add-on toolboxes rather than in basic FEMLAB. Similarly, the

application of point loads to structures requires the structural mechanics toolbox. Testing and validation can also be difficult. Owing to the use of triangular and tetrahedral elements only, it is difficult to implement NAFEMS benchmark tests, for example, which frequently require meshes based on quadrilateral or brick shapes.

FEMLAB User Testimony 2

Version

Femlab 2.3, running under Matlab version 6.5, networked PC Windows-based version

Hardware

Windows PC

General use

The software is mainly used for two problems. The first problem is simulation of a microwave radiometry problem which requires solution of Maxwell's equations. The second problem is simulation of a transient heat flow problem involving heating of a cylindrical sample by a pulsed laser.

Features used

Both problems are run using two-dimensional models. The electro-magnetics problem is taken to be constant in the third dimension, and the heat flow problem is axisymmetric.

The electro-magnetics (e-m) problem is made up of several layers of material and an antenna and calculates microwave propagation through the structure. The model has been investigated with the e-m options in the main body of FEMLAB as well as the e-m module which can simulate e-m wave propagation.

The heat flow problem has used three different methods of modelling the heat due to the laser: specification of a temperature profile for those areas directly affected by the laser, specification of a power density on the outer surface of the sample, and specification of a heat flux per unit volume in the affected region. The heat generation has been defined using a spatially Gaussian distribution, oscillating sinusoidally over time. The cylinder is made of several layers of different materials

Advantages

In general the package runs well as a stand-alone entity. The software is user-friendly, particularly for inexperienced modellers. The menu-driven model development procedure makes most stages in setting up a model straightforward (with some caveats: see below). The run time for the jobs mentioned above is short, making sensitivity investigations a possibility in future. The post-processing and visualisation tools are good, making it easy to visualise results in a variety of different ways (contour plots, x-y plots, etc.).

Disadvantages

Some problems have been found with interfacing the package with the main body of Matlab. In particular, exporting data to Matlab and plotting it there (which is useful because Matlab has some graphical capabilities that Femlab does not) has been challenging, and an attempt to call Femlab directly from an M-file has been unsuccessful so far.

Some of the terminology used to define input parameters and boundary conditions can be confusing. For example, the definition of boundary conditions in the e-m problem required definition of a quantity E that was not easy to identify. It is not always straightforward to find answers to such questions in the manual.

12.5 LUSAS

Supplier

Lusas, Forge House, 66 High Street, Kingston upon Thames, Surrey KT11HN

LUSAS User Testimony 1

Version

LUSAS Composite 13.5-2

Platform

PC Windows-based

General comments

The composites group in our organisation has used LUSAS for approximately 12 years and the version currently in used is specifically tailored for modelling composite materials. The company that develops the software is based in Kingston upon Thames and provides excellent training and support services. LUSAS consists of LUSAS Modeller, which acts as the pre- and post- processor and LUSAS Solver, which performs the finite element analysis. Pre- and post- processing is intuitive, easily performed and well documented in the user-friendly Windows interface. LUSAS Composite has been used for a variety of linear and non-linear problems including adhesive lap joints, quasi-statically loaded composite plates, and bolted composite and metallic T-joint configurations. Of particular interest to our usage, the package contains numerous material models (e.g. for metals, composites, rubbers, concrete, adhesives, volumetric crushing of foams etc.)

Features used

Much of our recent work has involved the analysis of composite laminates in various structural applications. With knowledge of the ply properties of a material and the lay-up of a laminate, this is easily performed and viewed in the LUSAS composite material attribute section. The latest version contains an option for modelling symmetric laminates in order to reduce the data input required and provides a composite material property library for a variety of materials. Various composite failure criteria are available including Tsai-Hill, Hoffman, Tsai-Wu (with Cowin extension) and Hashin (fibre and matrix). Many of the models recently analysed using LUSAS have involved components where contact exists between parts of the model either before a load is applied or during loading. The 'slideline' feature in LUSAS was used to model existing and/or potential areas of contact, as it is ideal for modelling situations where there is no exact prior knowledge of the contact process.

Advantages of software

The features mentioned above are extremely advantageous. In general, the package is easy to use and includes many features for effective modelling of composite material problems. Although not commonly used by our group at present, LUSAS includes 2D

and 3D composite delamination interface elements enabling delaminations to be modelled using an incremental non-linear analysis. Mode I and II growth of delaminations can be modelled. The latest release contains improvements to the composite brick elements available in order to compute through thickness effects more effectively, and thermal composite elements have been included to allow thermal-structural problems to be solved.

Usability issues

During recent analyses some bugs in the LUSAS code have been discovered. For example, when using composite material properties with a damage model the input '.dat' file was written in the wrong order and had to be adjusted manually. However, this problem has been fixed in the latest release. Models involving contact (use of slidelines) and composite materials have tended to be expensive in terms of solution time and often models have had to be simplified by using global rather than individual ply properties. The level of support (via e-mail, phone, website postings etc.) is excellent and advice on specific problems is usually offered within 24 hours of a query being submitted. LUSAS also offer consultancy services and during a recent work task a day was spent at the LUSAS offices which proved to be extremely helpful.

12.6 PAFEC

Supplier

PACSYS Ltd. Strelley Hall, Strelley, Nottingham, NG8 6PE

PAFEC User Testimony 1

Version

PAFEC Level 8.7

Platform

PC based, (can also be workstation based)

General use

PAFEC-FE can be used to solve static, dynamic, non-linear, and thermal analysis finite element problems. It also includes vibroacoustic, lubrication, and non-linear piezoelectric analysis. The pre- and post-processor associated with PAFEC is PIGS. The manufacturer also claims some form of compatibility with NASTRAN, PATRAN, and I-DEAS.

Usability

As I have had no specific training in PAFEC, I found it a bit difficult to use at first. The user's manual can also be a bit confusing and not so helpful at times, which is probably true of all FE packages. I have used PIGS as a post-processor and found it to be useful, but I have never used it as a pre-processor.

PAFEC appears to offer all that an FE package should and probably offers more in some cases. In the few instances where I have required help from outside NPL with PAFEC, I emailed their support group and found them to be very helpful and quick to respond. The only disadvantage with PAFEC that I can see is that it doesn't appear to

be as widely used as some other FE packages, which might limit compatibility with other types of software or other companies.

PAFEC User Testimony 2

Version

PAFEC 8.6 Vibroacoustics software

Platform

SUN workstation using UNIX operating system as well as PC Windows-based version. Soon to be transferred exclusively to PC.

General comments

PAFEC is used for modelling the behaviour of ultrasonic and audio-range transducers, with fluid loading or in vacuum, for both continuous and transient excitation.

Features used

The package allows the coupling of vibrating structures to the modelling of finite and infinite regions of a fluid (boundary elements, wave envelope elements). It also allows piezoelectric elements to be defined, which is critical to the simulation of ultrasonic transducers. PAFEC's mathematicians have been willing to work with us both to extend the functionality of the PAFEC software and to provide us with bespoke software to tackle specific modelling problems. This resulted in code designed to allow the prediction of acoustic field pressures resulting from the transient excitation of transducers. PAFEC's mathematicians are currently in the process of developing a viscous element to be used in the modelling of audio range transducers. Quantities routinely obtained from PAFEC include: acoustic pressure, structural displacement and electrical impedance.

Advantages of software

The features mentioned earlier, in particular the ability to account for fluid-structure interaction, make this package ideal for the prediction of acoustic pressure fields of both acoustic and ultrasonic transducers.

Usability issues

There are some bugs in the PAFEC software. For example, fluid-structure coupling is not always straightforward. Fortunately, the support provided is usually good and these problems can be overcome. Also, visualising the data is best done in another software (e.g. Matlab), since the quality of the graphics in PAFEC is average.

PAFEC User Testimony 3

Version

PAFEC 8.7

Platform

Windows-based PC

Application

Heat transfer in measurement equipment

General comments

PAFEC was originally chosen for this work partly due to the accessibility of the package and partly due to the model developer's experience with the package. After initial development, models were passed on to a newer user for repeated use.

Features of package used

PAFEC has been used to model heat flow in a variety of equipment used for measurement of thermal properties of substances. Steady and transient jobs have been run, and boundary conditions have involved convection and radiation.

The convection model is for losses from a surface and does not include simulation of the fluid that the heat is lost to. It is defined by an ambient temperature and a film coefficient, both of which can be time dependent. Radiation is not modelled explicitly, but as it is possible to define a film coefficient that is temperature-dependent, black-body radiation can be simulated by defining appropriate values provided the ambient temperature is fixed.

Material properties can vary with temperature and time. Heat transfer across a gap can be modelled using an element with a known heat transfer coefficient (units $\text{Wm}^{-2}\text{K}^{-1}$). For transient models, the time history of the temperatures or heat fluxes at particular points can be specified, as can generated heat per unit volume. Boundary elements can also be used for steady-state jobs, and thermal stress analyses can be carried out but we have not tried either of these.

Due to the simple structure of the equipment, much use has been made of PAFEC's PAFBLOCKS feature which generates meshes of a given density and form within simply-defined blocks.

It is possible to automate simple post-processing tasks and this feature has been used extensively for these models.

Usability issues

If the geometry of the job is simple, PAFEC is a straightforward package to use. The initial model designs were passed on to a user with no previous FE experience who found that the comment lines within the input file were sufficient to run the file and to carry out simple alterations and parameter studies.

The main usability problem is with the graphical pre-and post-processing. The pre-processor is not that good for mesh generation, and it can be difficult to extract results from the post-processor, particularly if the temperature history of a single node in a large model is required. In general, mesh generation of shapes that are not built up from simple blocks can be time-consuming to mesh. The reliance on screen picking of the pre- and post-processing software makes it difficult to use, particularly for three-dimensional jobs. However, the automation of post-processing tasks has meant that if the geometry does not change, the difficult tasks can be automated to save time when a job involves repeated runs of the same file.

PAFEC does not include an automatic time-step generator for transient jobs which is inconvenient and can lead to repeated runs being necessary to identify a suitable value.

PAFEC support is generally good (i.e. helpful and rapid). However, the manuals are not

always useful if you know what you would like to do but do not know the name of the command. The worked examples are useful, as is the general section on the different uses of PAFEC, but the index could be better.

PAFEC User Testimony 4

Version

PAFEC 8.7

Platform

SUN workstation using UNIX operating system

Application

Simulating the behaviour of a medical ultrasound physiotherapy transducer operating at 1 MHz, and the subsequently generated near-field.

The problem

Understanding the response of medical ultrasound devices and their generated acoustic fields is necessary for treatment safety, efficiency and efficacy. A common design for a physiotherapy ultrasound device has an active element housed in a protective casing and is acoustically well matched to the media it operates in through the use of a matching layer. When the active component is excited it deforms, generating mechanical disturbances in the surrounding components that will propagate into the acoustic region as sinusoidally varying pressure waves – assuming small amplitude, harmonic excitation. The deformation of the structure is likely to be complicated and highly non-uniform, particularly for an un-backed device as considered here. Close to the device the generated field is complex with rapid variations in pressure over distances smaller than a wavelength. At 1 MHz the acoustic wavelength in water is approximately 1.5mm.

We use PAFEC to simulate the response of a variety of acoustic transducers and submerged structures, including medical ultrasound devices, and to predict near- or far-field response due to radiation or scattering. Furthermore PAFEC can be used as a tool to analyse the behaviour of the structure and to investigate significant design features with the intention of proactively effecting device design or considering novel designs and applications. Within this process, as with all numerical modelling applications, model validation is paramount. Here validation is made against experimental measurements on real devices, through comparisons against alternative predictive tools, or against theoretical/numerical predictions for idealised systems.

Setting up the problem in PAFEC

There are principally two ways to generate PAFEC models – using the graphical interface (PIGS), or directly in a text-like data file (.dat). Although the geometry of the problem and its basic features can be laid out directly in PIGS a number of attributes must be subsequently added by editing a PIGS generated .dat file. For problems with a simple geometry, as here, it is reasonable to construct the entire model in text format directly in the .dat file.

The .dat file has two parts – a short CONTROL phase providing global instructions and general information, and a larger section separated into modules that specify full model details. The CONTROL module manages the running of the simulation through a series

of phases. At each phase the simulation can be stopped, stored, analysed and re-started. Phase 1 is essentially a housekeeping stage to check for errors in the text of the .dat file. Phase 2 is required if automatic mesh generation is used, and generates full nodal and topological data. Phase 4 performs geometry checks on the elements and considers the definition of boundary conditions. Phase 6 generates the system of matrix equations that fully describe the simulation in terms of a series of simultaneous equations. Phase 7 solves the matrix equations for the primary unknowns. Additional intermediate and post-processing phases may also be included. To experienced FE users much of the information in the second section will be familiar, as the format is common to many codes. Separate modules specify the positions of fundamental nodes, the geometry and topology for elements, materials data, meshing requirements, restraints, loading, and any specific output requirements.

The system modelled here is a dynamic, harmonic, fluid-structure interaction problem where the projector operates into an unbounded fluid region. Axial symmetry is assumed, including loading and response so only the fundamental circumferential component is considered here. However PAFEC can also accommodate non-symmetric response in an axisymmetric model, 2D planar models, full 3D models, periodic systems and transient analyses. Here model generation requires the specification of several different isotropic components, a piezoelectric component and the surrounding fluid region. The availability of piezoelectric elements is obviously an important aspect of PAFECs capabilities here. The mesh intensity throughout the model is essentially determined by the acoustic wavelength in water, where a rule of thumb for meshing is a minimum of 3 elements per wavelength, and preferably 8-10. It should be noted that even for an axisymmetric model an acoustic wavelength of 1.5 mm places serious computational requirements on the simulation. It should also be mentioned that an operating frequency of 1 MHz is relatively low for medical ultrasound applications.

One of the most significant features of many fluid-structure interaction problems is the choice of technique for approximating an 'infinite' fluid region. A number of finite element and boundary element methods are available within the literature for modelling unbounded fluid regions (Astley, Macaulay and Coyette, 1994; Francis, 1993; Geers and Felippa, 1983; Seybert and Rengarajan, 1987). Another important feature of PAFEC is that it offers several alternative fluid-modelling options, which can be of some considerable significance as each method is restricted in its use by practical or theoretical considerations. The size of the model (in terms of nodes and elements), the frequency range of interest, the frequency-size relationship of the system, the required accuracy of the solution and the nature of the required output are all features that can effect the suitability of the fluid-modelling option.

Some considerable time was spent determining and validating the best solution for modelling an unbounded fluid region here. It was decided that the two most appropriate options were the use of acoustic elements and 'infinite' wave envelope elements (WEEs) and a doubly asymptotic approximate boundary element technique (DAA2c). WEEs simulate outward travelling, decaying pressure response and when applied at an appropriate boundary within the acoustic region they effect radiation to infinity. DAA uses an approximate fluid-structure relationship rather than the 'exact' approach adopted by other boundary element formulations based on the solution of the surface Helmholtz integral. This approximation is expected to be effective at high frequencies – equivalent to an 'early time' for transient analyses on which the DAA formulation is based - and this has been substantiated here. Whilst the use of WEEs was seen to be more stable in terms of model validation it is computationally impractical for

approaching any substantial analysis and large numbers of model runs. By contrast DAA2c, whilst less reliable in terms of validation, was substantially faster. An approach of DAA2c model validation by comparison against experimental measurements, idealised predictions and selected WEE models proved successful and practicable here.

The required output for our simulations was a comprehensive description of the generated near-field, with the majority of model validation through experimental comparison. A number of measurements were carried out at the National Physical Laboratory - in accordance with the IEC standard on physiotherapy transducers (Hekkenberg, Reibold and Zeqiri, 1994; Hekkenberg, 1998), as well as in terms of an assessment of axial, near-axial, cross-axial and planar response close to the face of the transducer. Generally excellent model-experimental agreement was established to within a few wavelengths of the device, where the response is complex and highly non-ideal. Some analysis of the structural model was also made, highlighting the non-uniformity of the response of the device, and the most and least sensitive components.

Usability

In many respects the usability of PAFEC is its biggest drawback. PAFEC and PIGS are not particularly user-friendly because of their limited pre- and post- processing capabilities. This is in comparison with other codes where input and output are presented in a 'standard' CAD or CAE type format which is easy to use and fairly intuitive in terms of model generation and solution. PIGS is simply not as straightforward to use as this, whilst the text generation of .dat files, although fairly standard for the most part, does require the specification of a number of modules that must be understood more through practical experience rather than through obvious reasoning. Difficulties here are compounded by the fact that the PAFEC user manuals are not always as helpful as they could be.

There are, however, several advantages with PAFEC that offset these drawbacks. Firstly few, if any, alternative packages offer the extensive capabilities for solving problems involving fluid-structure interactions, infinite fluid regions or piezoelectric transducers that PAFEC does. Secondly the PAFEC support is excellent and fast. Thirdly is a more general point. The semi-automatic generation of FE models and solutions that is now available in many software packages can lead to the generation of models without a complete understanding of the nature of any approximations used or restrictions on the applied solution techniques. By requiring some good understanding of the solution processes and solution techniques used - actually just to generate models that run - PAFEC has more chance of allowing the user to avoid such pit falls.

Acknowledgements

The authors would like to thank the following people who have contributed descriptions of their work to the section on software packages: Louise Crocker, Chris Fry, Pierre Gélât, Michael Gower, Martin Hughes, Andrew Levick, Jennifer Murphy, Jeremy Noyce, Mike Rabbitt.

They would also like to thank Pierre Gélât and Edward Johnson for their useful comments and suggestions at the review stage.

Annex A: Glossary

Readers who require an extensive glossary of technical terms associated with finite element analysis should consult the NAFEMS guide *How to understand finite element jargon* (NAFEMS, 1994). This guide contains a clear description of the main technical terms encountered in defining meshes and in considering solution methods. It also gives straightforward definitions of a large range of terms associated with finite element analysis itself, mathematics and numerical analysis, and computer hardware and software. Those who are new to the field of finite element analysis might find it useful to begin with a study of this text.

In this annex to the report we present some definitions of key terms which are useful in understanding the arguments that have been presented. It is not intended to be exhaustive, and does not generally include technical terms related to specific problems described in part two of this report. In addition, wherever possible technical terms have been defined when they are first used in the main text.

Adaptive meshing: a technique where the mesh is regenerated and the FE problem re-solved, with the aim of improving the mesh and the accuracy of the solution, at each iteration.

Aspect ratio: the ratio of the side lengths of an element. The aspect ratio of square and cubic shapes is 1.

Axisymmetric element: a type of element that can be used to model a solid of revolution, that is, a three-dimensional structure that can be formed by rotating a planar surface around a central axis. The geometry, materials and constraints must all be axisymmetric for use of this type of element.

Brick: a three-dimensional solid element, typically a hexahedron with eight, 20 or 32 nodes.

Bulk modulus: the change in volume of a substance as the pressure on it changes, given by

$$K = -V \left(\frac{dP}{dV} \right) = \rho \left(\frac{\partial P}{\partial \rho} \right),$$

where K is the bulk modulus, with units of Pascals, P is pressure, V is volume and ρ is density.

Cauchy's equation of motion: a differential equation, used in fluid dynamics, that gives the acceleration of a fluid in terms of the local volume forces and the stress tensor. The equation has the form

$$\rho \frac{Du_i}{dt} = \rho F_i + \frac{\partial \sigma_{ij}}{\partial x_j}.$$

Characteristic equation: an equation which has a solution subject to specified boundary conditions only when a parameter in the equation has certain values. For example $\mathbf{A}\mathbf{u} = \lambda\mathbf{u}$, the eigenvalue equation for a matrix \mathbf{A} , will only have a non-zero solution for \mathbf{u} for certain values of λ (the eigenvalues).

Cholesky factorisation: for a symmetric positive definite square matrix \mathbf{X} , the Cholesky factorisation of \mathbf{X} is the factorisation $\mathbf{X}=\mathbf{U}'\mathbf{U}$, where \mathbf{U} is an upper triangular matrix (it has only zeros below the diagonal), and \mathbf{U}' is its transpose. The advantage of using this factorisation is that once \mathbf{U} has been computed, it becomes easier to calculate the inverse of \mathbf{X} , because $\mathbf{X}^{-1} = \mathbf{U}^{-1}(\mathbf{U}')^{-1}$, and the inverses of \mathbf{U} and \mathbf{U}' are simple to compute.

Classification of PDEs: for a general second-order pde of the form

$$Au_{xx} + 2Bu_{xy} + Cu_{yy} + Du_x + Eu_y + F(u, x, y) = 0$$

the type of equation is classified according to the value of $AC - B^2$. If $AC - B^2 > 0$, the equation is elliptic, if $AC - B^2 < 0$, the equation is hyperbolic, and if $AC - B^2 = 0$, the equation is parabolic.

Compliance: the inverse of stiffness. See stiffness matrix and Hooke's Law definitions.

Contact analysis: a finite element analysis in which more than one structure is present and the structures interact with one another.

Convergence: the tendency of a numerical solution to approach a particular result as the numerical error is reduced.

Cross-ply laminate: a structure which consists of an arbitrary number of layers of the same material and thickness but with alternating orientations of 0° and 90° . Such structures are orthotropic (see definition below), with a Poisson's ratio of nearly zero.

Degrees of freedom: the fundamental variables of interest in a finite element analysis. Typically these are the allowed translations and rotations at a node. A node in a three-dimensional structure will have more allowed freedoms at a node (three translations and three rotations about the axes) than a node in two or one dimensional element. In cases where the variable of interest is temperature or pressure there will be only one degree of freedom at the node.

Eigenfrequency: one of the frequencies at which an oscillatory system can vibrate. Also called natural frequencies or fundamental frequencies. The associated deformations of the structure are called eigen-, natural, or fundamental modes.

Eigenvalue equation: see characteristic equation.

Elastomer: a polymer material such as a synthetic rubber or plastic which can be stretched to at least twice its original length at room temperatures and which returns to approximately its original length when the stress is released.

Elliptic equation: general type of second order partial differential equation, which includes Laplace's equation. See classification of PDEs for a further explanation.

Finite Volume Method: a method popular for the solution of computational fluid dynamics problems, including both fluid flows and heat flows. Unlike the finite element method in which the domain of interest is divided into elements defined by their nodes, the finite volume method divides the domain into a series of cells or control volumes. Flows into and out of the volume or cell are defined in terms of integral equations. The integral equations are converted into a system of algebraic equations for computer solution of the problem of interest.

Fredholm integral equation: an inhomogeneous Fredholm equation of the first kind has the form

$$g(t) = \int_a^b K(t,s)f(s)ds,$$

where $g(t)$ is known and $f(t)$ is the function for which the solution is required. The equivalent matrix equation is $\mathbf{K}\mathbf{f} = \mathbf{g}$, with the solution $\mathbf{f} = \mathbf{K}^{-1}\mathbf{g}$. This has a unique solution when \mathbf{g} is not zero and \mathbf{K} can be inverted. The Fredholm integral of the second kind is

$$f(t) = \lambda \int_a^b K(t,s)f(s)ds + g(t).$$

The equivalent matrix equation is $(\mathbf{K} - \sigma\mathbf{I})\mathbf{f} = \mathbf{g}$.

Gaussian elimination: a computationally efficient method of solving systems of equations of the form $\mathbf{Ax} = \mathbf{b}$, which are common in finite element analyses.

Graded mesh: a mesh in which the element size varies as a function of position.

Helmholtz equation: an equation of the form

$$\nabla^2\psi + k^2\psi = 0,$$

where ∇^2 is the Laplacian operator. In electromagnetic and acoustic problems k is the wave number $k = 2\pi / \lambda$, where λ is the wavelength.

h-method: see *p*-method

Hooke's Law: a law relating the force applied to a linear elastic body to its deformation. For three dimensional anisotropic elastic bodies, Hooke's law states that the six components of stress are linearly related to the six components of strain in the following manner:

$$\begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{zz} \\ \varepsilon_{yz} \\ \varepsilon_{zx} \\ \varepsilon_{xy} \end{bmatrix} = \begin{bmatrix} S_{11} & S_{12} & S_{13} & S_{14} & S_{15} & S_{16} \\ S_{21} & S_{22} & S_{23} & S_{24} & S_{25} & S_{26} \\ S_{31} & S_{32} & S_{33} & S_{34} & S_{35} & S_{36} \\ S_{41} & S_{42} & S_{43} & S_{44} & S_{45} & S_{46} \\ S_{51} & S_{52} & S_{53} & S_{54} & S_{55} & S_{56} \\ S_{61} & S_{62} & S_{63} & S_{64} & S_{65} & S_{66} \end{bmatrix} \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{yz} \\ \sigma_{zx} \\ \sigma_{xy} \end{bmatrix},$$

where $\boldsymbol{\varepsilon}$ is strain, $\boldsymbol{\sigma}$ is stress, and \mathbf{S} is the compliance matrix (the inverse of the stiffness matrix).

Hoop stress: the circumferential stress in a cylinder subject to internal or external pressure.

Hyperbolic equation: general type of second order partial differential equation that includes the wave equation. See Classification of PDEs for further explanation.

Isoparametric element: This is an element in which the same shape function are used for interpolating the geometry and the displacement.

Jacobian: for the functions $f_i(x_1, x_2, \dots, x_n)$, $i = 1, 2, \dots, n$, where the x_j are real, the Jacobian is the determinant of the matrix whose i^{th} row gives all the first-order partial derivatives of the function $f_i(x_1, x_2, \dots, x_n)$ with respect to the variables x_j .

Method of moments: Numerical techniques that are based on the method of weighted residuals are known as moment methods. Within the discipline of computational electromagnetics the “method of moments” is frequently used as a synonym for boundary element methods. Such methods are good at solving unbounded radiation problems, perfect electrical conductor arrangements and homogeneous dielectrics. The method transforms a physical problem defined in terms of integral equations into a dense linear system. For complex three-dimensional objects the boundary element method or method of moments can result in systems of equations with very large numbers of unknowns. Efficient solution of such systems may require advanced computational techniques.

Modal analysis: a finite element study to determine the natural frequencies of vibration of a structure.

Navier-Stokes equations: equations that govern the behaviour of compressible and non-compressible viscous fluids.

Non-linearity: two types of non-linear behaviour are important in finite element analysis. Geometric non-linearity refers to non-linearity caused by large deformations. If there is a sufficiently large change in shape, the stiffness may also change even though the material’s properties do not change since stiffness is a function of both the material and the geometry. Material non-linearity means that there is a non-linear relationship between stress and strain. Such situations commonly arise if a material is loaded above the value of its yield stress.

Orthotropic: a material that possesses three mutually perpendicular planes of elastic symmetry, with each usually having different properties.

Parabolic equation: a general type of second order partial differential equation that includes the heat equation. See Classification of PDEs for a further explanation.

Plastic deformation: a change in the dimensions of an object under load that is not recovered when the load is removed.

***p*-method:** one of the two most common methods of increasing the accuracy of a finite element simulation by increasing the complexity of the problem, the other being the *h*-method. Finite element analyses calculate solutions by using polynomial shape functions to approximate quantities on a mesh. The two simplest ways of increasing the accuracy of the solution are to increase the order of the polynomial approximation (*p*-method) or decrease the mesh size (*h*-method). The use of both methods in combination is called *hp*-refinement. Some software packages will carry out automatic *h*-, *p*- or *hp*-refinement to improve solutions.

Poisson's ratio: a dimensionless constant defined as the ratio of the change in width per unit width to the change in length per unit length of a material during stretching.

Principal stress: a stress that occurs in a direction normal to a principal plane of stress. At any point in an elastic material, a plane at that point across which the shear stress is zero is a **principal plane** of stress. The direction normal to this plane is called a **principal direction**.

Reynolds equations: a set of partial differential equations for studying turbulent flow problems which are related to the Navier-Stokes equations.

Reynolds number: the ratio of inertial forces to viscous forces in a fluid. When the Reynolds number is high, inertial forces dominate leading to turbulent flow. If the

Reynolds number is low, viscous forces dominate, resulting in laminar flow.

Rigidity modulus: see shear modulus.

Shape function: a function describing how a quantity (a degree of freedom) such as displacement varies within an element. There will be a function which applies at each node in the element. In general shape functions N_i are defined such that

$$N_i(\mathbf{x}_j) = \begin{cases} 1, & i = j \\ 0, & i \neq j, \end{cases}$$

where \mathbf{x}_j are the nodal positions. Shape functions are also used to describe how the geometry of the element varies in terms of the co-ordinates of the nodes. Shape functions allow the displacement and the geometry to be interpolated over the element

Shear modulus: the ratio of the shear stress to the shear strain, describing the change in shape of a solid resulting from a tangential stress. It is also called the rigidity modulus.

Singularity: this term has a number of meanings but its most common uses in finite element analysis relate to infinitely large values of a term which arise from attempts to divide by zero, or to an infinite stress occurring at the location of a point load or a sudden change in geometry. A further use arises in the term *singular stiffness matrix* which is a stiffness matrix (see definition below) with a zero determinant.

Spline: a (polynomial) spline of order n (i.e., degree $n - 1$) is a function composed of a set of polynomial pieces of order n . Each polynomial piece joins its neighbours at points known as knots. At the knots, the function value and derivatives of all orders up to and including that of order $n - 2$ are the same for the two polynomial pieces that meet. The most commonly used splines are those of order four (cubic splines), which are composed of polynomial pieces of order four (degree three) and so are continuous in value and first and second derivatives. Splines are often used to provide a smooth interpolation of a set of data points having distinct, increasing, abscissa values.

Stiffness matrix: a matrix relating the displacements of a structure to the forces applied to it. This relationship is of the form $\mathbf{Kd} = \mathbf{F}$, where \mathbf{K} is the stiffness matrix, \mathbf{d} is a column vector of displacements and \mathbf{F} is a column vector of forces. It is called the stiffness matrix because this formula appears similar to Hooke's law of springs. The term stiffness matrix is used even when the quantities \mathbf{d} are temperatures or pressures, say, rather than displacements.

von Mises stress: a stress invariant given by:

$$\sigma_{\text{vM}} = \sqrt{\frac{(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2}{2}},$$

where σ_1 , σ_2 and σ_3 are respectively the maximum, intermediate and minimum principal stresses in the material (see definition of principal stress). It is commonly used as a yield stress for plasticity in metals.

Young's modulus: the ratio of normal stress to normal strain for tensile or compressive stresses below the proportional limit of the material (the greatest stress a material can sustain without any deviation from proportionality of stress to strain). For perfectly elastic materials, Young's modulus is the ratio of the change in stress to change in strain within the elastic limits.

Annex B: References

- ABAQUS v 5.5 User's Manual. Hibbitt, Karlsson and Sorensen, 1995.
- Ali M. W., Hubing T. H. and Drewniak J. L., A hybrid FEM/MoM technique for electromagnetic scattering and radiation from dielectric objects with attached wires, *IEEE Trans. on Electromagnetic Compatibility*, v30, No. 4, pp 304-314, 1997.
- Ames W. F., Nonlinear partial differential equations in engineering, Academic Press, 1972.
- Ames W. S., Numerical methods for partial differential equations, 3rd edition, Academic Press, 1992.
- Astley R. J., Macaulay G. J. and Coyette J. P., Mapped wave envelope elements for acoustical radiation and scattering. *J. Sound Vibrat.*, 170(1), pp. 97-118, 1994.
- Berenger J. P., A perfectly matched layer for the absorption of electromagnetic waves, *J. Comp. Phys*, Vol 114, pp 185-200, 1994.
- Brandt A., Multi-level adaptive solutions to boundary-value problems with boundary layers, *Mathematics of Computation*, pp 333-390, 1977.
- Bungartz H.-J., Dornseifer T., Sparse grids: recent developments for elliptical partial differential equations, Lecture notes in computational science and engineering: European multigrid conference No 3 pp 45-70, Springer, 1998.
- Burton A. J. and Miller G. F., The application of integral equation methods to the numerical solution of some exterior boundary-value problems, *Proc. R. Soc. London Ser. A* pp 201-210, 1971.
- Cook R. D., Finite element modelling for stress analysis, John Wiley, 1995.
- Dadson R. S., Lewis S. L., Peggs G. N., The Pressure Balance: Theory and Practice, London, HMSO, 1982.
- Engquist B. and Majda A., Absorbing boundary conditions for the numerical simulation of waves, *Math. Comput.*, v31, pp 629-651, 1997.
- Esward T. J., Finite element modelling of oil-operated piston-cylinder pressure balances: determining the uncertainties in distortion coefficient predictions using the finite element method, National Physical Laboratory Report CMAM 85, October 2002.
- Esward T. J., Lord G. J., and Wright L., Model Validation in Continuous Modelling, National Physical Laboratory Report CMSC 29/03, September 2003.
- Esward T. J., Johnson E. G., and Wright L., Testing Continuous Modelling Software, NPL report in preparation, 2004.
- Francis D.T.I., A gradient formulation of the Helmholtz integral equation for acoustic radiation and scattering. *J. Acoust. Soc. Amer.*, 93(4), pp. 1700-1709, 1993.
- Geers T.L. and Felippa C.A., Doubly asymptotic approximations for vibration analysis of submerged structures. *J. Acoust. Soc. Amer.*, 73(4), pp. 1152-1159, 1983.
- Gershenfeld N., The Nature of Mathematical Modelling, Cambridge University Press, 1999.
- Griebel M., Adaptive sparse grid multilevel methods for elliptic PDEs based on finite differences, *Computing* 61 pp 151-79, 1998.

- Gurson A. L., Continuum theory of ductile rupture by void nucleation and growth: Part I- Yield criteria and flow rules for porous ductile materials, *J. Eng. Mater. Technol. Trans ASME* 99, pp2-15, 1977.
- Hannaby S. A., Finite element mesh generation and isoparametric elements, NPL Report DITC 97/87, National Physical Laboratory, 1987.
- Hannaby S. A., A comparison of four finite and infinite element techniques, NPL Report DITC 123/88, National Physical Laboratory, 1988a.
- Hannaby S. A., A mapping method for mesh generation, *Comput. Math. Applic.* Vol 16, No 9, pp 727-735, 1988b.
- Hannaby S. A., The use and analysis of infinite elements, Ph D thesis, City University, London, 1994.
- Hekkenberg, R. T., Characterising ultrasonic physiotherapy systems by performance and safety now internationally agreed. *Ultrasonics*, 36, pp. 713-720, 1998.
- Hekkenberg, R. T., Reibold, R. and Zeqiri, B., Development of standard measurement methods for essential properties of ultrasound therapy equipment. *Ultrasound Med. Biol.*, 20(1), pp. 83-98, 1994.
- Hutley I and Johnson R.M., Linear and nonlinear differential equations, Elis Horwood, Chichester, 1983
- Kolsky H., Stress waves in solids, Clarendon Press, Oxford, 1953.
- Lord G. J., Wright L., et al. Uncertainty Evaluation in Continuous Modelling. NPL report in preparation, 2003.
- McCartney L.N., and Saunderson S.E.T., A recursive method of calculating stress transfer in multiple-ply cross-ply laminates subject to biaxial loading. I: Development of the model, NPL report DMM(A)150, June 1995.
- Morton K. W. and Mayers D. F., Numerical solution of partial differential equations, Cambridge University Press, 1994.
- NAFEMS, NAFEMS A finite element primer, Glasgow, 3rd reprint, 1992.
- NAFEMS, How to choose a finite element pre- and post-processor, NAFEMS, 1994.
- NAFEMS, How to interpret finite element results, NAFEMS, 1997.
- NAFEMS, Why do non-linear finite element analysis?, NAFEMS and Hibbitt, Karlsson and Sorenson (UK) Ltd, 1998.
- PAFEC Theory, PAFEC Ltd, Nottingham, 1984
- PAFEC, PAFEC-FE Level 8 Data Preparation Manual, PAFEC Ltd, Nottingham, 1994.
- Press W. H., Teukolsky S. A., Vetterling W. T. and Flannery B. P., Numerical Recipes in FORTRAN: the art of scientific computing, 2nd edition, Cambridge University Press, 1992.
- Schenck H. A., Improved integral formulation for acoustic radiation problems, *J. Acoust. Soc. Am*, 44, p41-58, 1967.
- Schubert F., Iterative grid refinement for elastodynamic finite-difference time-domain methods, 17th International Congress on Acoustics, CD-ROM Vol 11, S12-13 Numerical Methods in Acoustics, Rome, 2-7 September 2001.

Schubert F., Peiffer A., Köhler B., Sanderson T., The elastodynamic finite integration technique for waves in cylindrical geometries, *J. Acoust. Soc. Am.* Vol 104(5), pp 2604-2614, 1998.

Seybert, A. F. and Rengarajan, T. K., The use of CHIEF to obtain unique solutions for acoustic radiation using boundary integral equations. *J. Acoust. Soc. Amer.*, 81(5), pp.1299-1306, 1987.

Smith G. D., Numerical solution of partial differential equations: finite difference methods, Clarendon Press, Oxford, 1985.

Taylor A. B., Mathematical Methods in Applied Mechanics, Clarendon Press, Oxford, 1986.

Wang Y.-C., Murti V. and Valliappan S., Assessment of accuracy of the Newmark method in transient analysis of wave propagation problems. *Earthquake Engineering and Structural Dynamics* (21) pp 987-1004, 1992.

Weiland T., On the numerical solution of Maxwell's equations and applications in accelerator physics, *Particle Accelerators*, Vol. 15, pp 245-291, 1984.

Weiland T., Time Domain Electromagnetic Field Domain Computation with Finite Difference Methods, *International Journal of Numerical Modelling*, v9, pp 295-319, 1996.

Whitham G. B., Linear and non-linear waves, John Wiley, 1974.

Woodford C. H., Passaris E. K., Bull J. W., Engineering analysis using PAFEC finite element software, Blackie, 1992.

Wu T. W. (ed), Boundary Element Acoustics: fundamentals and computer codes, WIT Press, 2000.

Zenger C., Sparse grids, in Notes on Numerical Fluid Mechanics, Vol 31, Parallel algorithms for partial differential equations, editor, Hackbusch W, pp 241-251, 1991.