Structural analysis with finite elements

Paolo Rugarli
Castalia

thomas telford

Published by Thomas Telford Limited, 40 Marsh Wall, London E14 9TP, UK.
www.thomastelford.com

Distributors for Thomas Telford books are
Australia: DA Books and Journals, 648 Whitehorse Road, Mitcham 3132, Victoria

First published 2010

Text by Paolo Rugarli
Castalia Srl, Viale Abruzzi 1, 20131 Milano, Italia
www.steelchecks.com; paolo.rugarli@castaliaweb.com

Translated from Italian by Kim Williams
Via Cavour, 8; 10123 Torino, Italia
www.kimwilliamsbooks.com; kwb@kimwilliamsbooks.com

First Italian edition 'Calcolo structturale con gli elementi finite' by Paolo Rugarli
ISBN: 88-8184-295-8
EPC Libri Srl, Roma, Italia, 2003

www.icevirtuallibrary.com

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

ISBN: 978-0-7277-4093-9

This book is published on the understanding that the authors is solely responsible for the statements made and opinions expressed in it and that its publication does not necessarily imply that such statements and/or opinions are or reflect the views or opinions of the publishers. While every effort has been made to ensure that the statements made and the opinions expressed in this publication provide a safe and accurate guide, no liability or responsibility can be accepted in this respect by the authors or publishers.

FSC
Mixed Sources
Product group from well-managed
forests and other controlled sources
Cert no. SGS-COC-2953
www.fsc.org
© 1996 Forest Stewardship Council

Printed and bound in Great Britain by Antony Rowe Ltd, Chippenham

To Roberta, Francesco and Stefano.

Text by Paolo Rugarli

Castalia srl
Viale Abruzzi 1
20131 Milan (Milano) Italy
http://www.steelchecks.com
paolo.rugarli@castaliaweb.com

Translation by Kim Williams

Kim Williams Books
Via Cavour, 8
10123 Turin (Torino) Italy
http://www.kimwilliamsbooks.com
kwb@kimwilliamsbooks.com

First Italian edition:

Paolo Rugarli
Calcolo strutturale con gli Elementi Finiti
EPC Libri Srl, Rome (Roma), 2003
ISBN 88-8184-295-5

Table of Contents

PREFACE TO THE ENGLISH EDITION

This edition of my work on finite elements is substantially identical to the first Italian edition, but has been improved by some emendations to the text and the correction of a few errors and misprints. I wish to thank, in this regard, my friend and colleague Engineer Giorgio Nieri, who carefully read the text in order to find the errors that had escaped notice in the Italian edition and so allowed them to be corrected here.

The Italian version of the book includes a CD with a limited version (fifty nodes) of the finite element program "SARGON", which I developed over the course of some twenty years of work. Without disparaging the other programs available, I believe that free access to a program which is perfectly functional and can serve as a trial is extremely helpful, both for those who are learning to work with finite elements for the first time and those who already own other programs and wish to try another one. Before undertaking complex models with many degrees of freedom, it is important for beginners to have a clear understanding of how the finite element method works on small models. The English demo version of Sargon is available as shareware at this website: www.steelchecks.com/pro/sr/demo.asp.

I wish to thank the publisher, Thomas Telford, for their willingness to publish this work, and for their openness to new ideas, and in particular I sincerely thank Matthew Lane and Daniel Keirs, who showed an interest in my work from the very beginning.

The present English edition of this volume is the result of the scrupulous translation from the original Italian by Kim Williams, a specialist in the translation of scientific works from Italian to English: the author has found in her a valuable collaborator. To her go my most profound thanks.

Paolo Rugarli

Milan, February 2010

PREFACE TO THE ITALIAN EDITION

This book grew out of the desire to provide an easily understandable guide illustrating some of the principal aspects of the Finite Element Method, in order to facilitate awareness of the considerable power of the tools at the engineer's disposal today. It is designed as an intermediate link between books that provide no theoretical information whatsoever and specialised textbooks: the aim during its composition was to arrive at a clear articulation of the essential core of the problems, without allowing concerns about rigorous formal orthodoxy to obstruct that object. Ironically, in some case a simplification would have required another book (for instance, chapter 3, references to simplified theory, or chapter 11, the in-depth description of basic finite elements), and thus in this context greater formality was required for brevity's sake.

This present volume presumes to be nothing more nor less than an aid. While writing it, I have seen more clearly its limits, and the limits of my effort: I am well aware that works of this kind can easily be accused of oversimplification, approximation, incompleteness and presumption. Be that as it may, it is still necessary and urgent to provide engineers with a text on the finite element method that is less irksome and thorny, and thus of greater advantage to the quality of structural calculations, than is usual.

This book is an effort in that direction.

I wish to thank all of my colleagues who, over the years, through the questions and observations that arose in the course of actual practice, as well as their trust in me, have made it possible for me to examine carefully and better understand many problems which are dealt with here. I also wish to thank Mrs Clea Nardi, who helped with drawing the illustrations. Finally, I wish to thank my school, the Politecnico di Milano.

Milan, July 2003

1 INTRODUCTION

The 1980s witnessed a progressive spread of the use of the finite element method (or more briefly, the FEM) by professionals, engineering societies, and construction firms. In fact, the spread of software programs for finite elements began in the 1970s with the famous SAP (Structural Analysis Program[1]), the mother of all acronyms in this field. However, the difficulty of use and the scant availability of instruments for calculation outside of universities posed notable limitations on the spread of the FEM.

It should be noted, however, that the *demand* for analyses of finite elements was also limited: the standards and regulations did not require calculations of this kind, and – except in very special cases – clients did not ask for reports of calculations containing the results of finite elements computations. This was the era of handbooks of all kinds, of those works which had been growing in number since the nineteenth century, whose object was to give a sampling of the most frequent cases in order to provide a rapid guide for the engineer who was not up on the principle of virtual work, and was justifiably reluctant to solve by hand 'a small system' of five or ten unknowns. It was the glorious age of graphic statics.

Of course, this all changed with the advent of personal computing.

The increasing application of the FEM (and of the software related to it) brought with it such consequences as an evolution of the standards and regulations, which today require calculations that are increasingly sophisticated and less capable of being dealt with by hand; a greater demand for results that are general and comprehensible in concise forms such as colour maps, deformations, internal force diagrams, etc.; the rapid obsolescence of tools such as values arranged in tables, and many others.

One might expect that this would have led to an improvement in the quality of engineering works, and an increase in their level of safety. Unfortunately, with regard to quality, the situation is all too obvious.[2] With

[1] The SAP program was developed at the University of California Berkeley by Klaus Jurgen Bathe and Edward Wilson. Various versions of the program were released, the last being SAP V. Following SAP V the authors developed a code for nonlinear calculations, NONSAP (1973), and then split up. Prof. Wilson developed with others SAP80 and then SAP90, while Prof. Bathe developed ADINA.

[2] Among the many possible citations, I quote G. Sarà and R. Nudo, who say, for example, "it might appear surprising that the elevated quantity of recent structural failures regard not only old, decayed structures but newly built structures as well" (in "Fallimenti strutturali e deficienze previsionali"[Gori R. Ed. 2001]). Regarding the causes of this disastrous situation, along with the greed of those who profit at the expense of safety, ignorance and abuse, he says, "It should also be noted that the increasing recourse to computer instruments tends to induce in the designer a passive reception of data obtained by increasingly sophisticated programs for calculation, with the consequent progressive reduction of recourse to traditional but fundamental tools for evaluation, such as experience and critical judgement".

Structural analysis with finite elements
978-0-7277-4093-9

regard to an increase in the level of safety, for various reasons it has yet to be demonstrated that this has been actually achieved.

Even though the FEM is greatly in demand and its use widespread today, an awareness of its foundations and the rules underlying the calculations used does not appear to be equally widespread. This is both because there has not been ample time (many professionals have not been able to take university courses in the subject) and because the management of a finite elements program is often only one part of the work to be performed: there are a vast number of professionals who perform calculations but also do many other things as well, and so don't have time to dedicate themselves to learning the FEM and the software program they own that makes its use possible.

Specialised texts on the subject[3] require a significant commitment of time and concentration, and it is rare that the professional who already has so many claims on his time is able to get through them. Some of the textbooks are quite fine, but unfortunately they are not as well known as they should be, requiring as they do the passion and constancy of those who can devote months or even years to their study.

While it is understandable that specialised texts present an obstacle that not everyone can overcome, it is also true that the complete lack of study is quite dangerous: objectively, the use of a finite element software program without possession of basic notions about the FEM can give rise to a series of oversights and errors which, even if they remain undiscovered, nevertheless pose a formidable barrier to progress in *real* structural engineering.

Even today only a small minority of students of engineering and architecture attend courses that specifically deal with this subject.

The idea for this present book came to me during the course of experience *in practice* that spanned eleven years. The work during those years was providing explanations and assistance for users of the finite element software program named SARGON, which I had been writing since 1991. It was precisely while doing this that I realised that many explanations that were asked of me, regarded not the operative aspects, but those that were intimately connected to theory (the vast case histories of lack of proper constraints, which finite elements were appropriate for a given problem, on modal analysis, etc., etc.).

This book is arranged as follows:

- Chapter 2 is a panoramic overview of the FEM, and serves as an initial introduction to the material that will be dealt with. The idea is to provide from the very beginning a frame of reference into which the topics that

[3] The texts [Zienkiewicz and Taylor 1989], [Cook, Malkus & Plesha 1989] and [Bathe 1996] are genuine "bibles" that are cited universally and are fundamental for anyone who wants in-depth studies written by some of the authors who "created" the FEM.

are dealt with in successive chapters are inserted. Although a maximum of clarity has been sought, various concepts that will be illustrated later are just sketched out here: the reader should understand that these will be explained in greater depth in later chapters.

- Chapter 3 is a summary of the main points of the various simplified theories that must be known at least in broad terms by those who want to produce models. Those already familiar with the various theories might look at this in order to review the notation used. Those who do not know anything abut the theory of elasticity will encounter some difficulties that might lead them to want to study this in greater depth (Appendix IV is also intended to help with this). Most will find in this chapter a brief summary of pertinent facts to refresh their knowledge. Knowing the differential equations by heart is not indispensable, but mastery of the basic hypotheses is.
- Chapter 4 goes into the details of what the FEM presumes about a *generic finite element*, introducing all of the most important concepts. The emphasis has been placed on describing the problem in general terms, with reference to the displacement method, which was the first to be established historically and which represents the main route for understanding the FEM. This chapter is fundamental.
- Chapter 5 describes the assembly that brings together all the information about various finite elements.
- Chapter 6 deals with forces, and Chapter 7 with constraints, in the attempt to describe all of the main errors that can occur.
- Chapter 8 explains the typical problems related to the numeric solution of the final algebraic problem (solving), as well as the stress recovery of the elements. This chapter will reveal some of the tricks that make beautiful (and inaccurate) colour maps possible.
- Chapter 9 deals with the techniques that can be used for error evaluation (checking), for verifying every analysis for correctness and reliability.
- Chapter 10 briefly illustrates the problems typical of each of the various theoretical models examined in Chapter 3, from the point of view of finite elements, in order to relate the theories to problems encountered in actual practice.
- Chapter 11 describes the characteristics of finite elements most used in the various models. It does not pretend to provide detailed descriptions of all the elements (though it does provide descriptions of beams, rods, and basic elements of plane stress and plane strain), but does however present information on their effectiveness.

- Chapter 12 deals with the most typical problems of modelling, in order to provide guidelines for identifying and overcoming them. Listed are numerous examples of erroneous models and tables useful for creating new ones. The study of the errors made by others and of problems of mesh refinement can be helpful in understanding how to proceed, bearing in mind that use of a medium-level software was envisioned, such as those in widespread use among professionals.
- Chapter 13 mentions typical problems related to the implementation of automatic checkers, aimed at providing enough information to make it clear that these are not miracle programs but present problems and difficulties of their own.
- Chapter 14 is a collection of some real cases of calculation of finite elements.
- Appendix I is dedicated to notation and a review of some notions of calculus.
- Appendix II is dedicated to a very brief review of the theory of elasticity.
- Appendix III lists the principal symbols used in the book.
- Appendix IV is an essential glossary for quick reference.

2 OVERVIEW

2.1 A SIMILE

According to astrology, it is supposed to be possible to predict and interpret the destiny of an individual (or a group of individuals) on the basis of the relative motion of a set of celestial bodies such as stars, planets, etc. All facts of life, small and large, are supposedly governed by the movement of about twenty bodies in space.

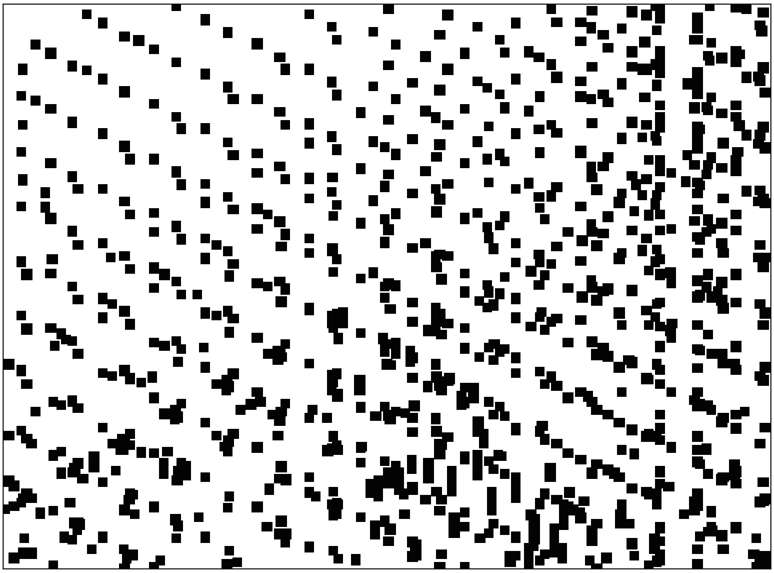


Figure 2-1: Partial view of a structure without a representation of the elements: the firmament of structural analysis.

This idea is ancient and lasting: still today magazines and newspapers publish horoscopes, and certain kinds of professional seers – astrologists – pocket hefty profits.

We wouldn't even go into that, except that there is a kind of point of contact between the finite element method (FEM) and astrology: like astrology, the FEM presumes knowledge of the details of a complex situation – what takes place within a given piece of a structure (a single *finite element*) – on the basis of the relative motion of a few points that are part of it. The points are called *nodes*, and the prediction is made – not unlike that of astrology – on the basis of an arbitrary judgment: the hypothesis that all the points within the finite element obey a certain mathematical law (for example, that the displacements vary

Structural analysis with finite elements
978-0-7277-4093-9

linearly or parabolically). We will call this assumption briefly and unofficially the *astrological principle*.

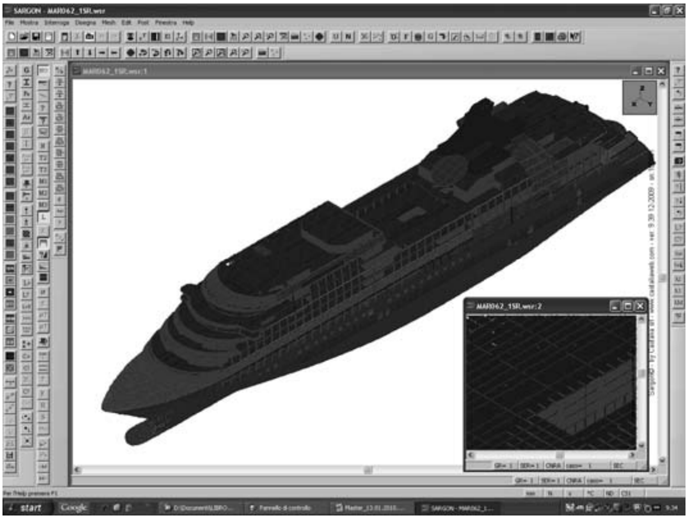


Figure 2-2: Model of the finite elements of the cruise ship Seabourn Odyssey, some like 1,066,000 degrees of freedom (Castalia srl for Studio Engineering).

Unlike astrology, the FEM is used to solve a system of differential equations of a known form, and thus it is possible to evaluate the efficiency – or if you prefer, the exactitude – of the solution quantitatively. In doing so, it is possible to see that the method works: it can be proven that by increasing the number of elements (which thus become progressively smaller), the approximate solution converges to the exact solution.[1]

Taking as it does its point of departure from an arbitrary judgment, how does the FEM work?

The answer lies in the fact that the finite elements can be made as small as desired, so that the "exact" answer is sufficiently comparable, within the element, to the displacement field whose shape has been arbitrarily hypothesised. It is a little like creating a complex image through the use of many coloured rectangular pixels arranged on a grid: if the pixels are sufficiently numerous, the image will have a high resolution, up to the point where the image is so clear that we

[1] It is also possible to prove that by increasing the complexity of the interpolating functions in keeping with the dimension of the element, the method converges to the exact solution. Something similar happens with the series that transform a complex function – such as sin(x) – into the sum of a given number of elementary functions: $1 - x^2/2 + \ldots$.

completely forget how it was created. If instead there are not enough points, then the image appears visibly "erose" and the means of its creation evident.

This kind of approach is very similar to interpolation, with the added advantage that it is possible to choose the points where the unknown function is to be sampled. As we said, the points are called *nodes*, and we effectively decide on their collocation in the structure.

In describing how the FEM works and what must be kept in mind when working with an analytical software program, we will proceed by successive approximations, exploring the concepts set forth in increasing depth.

Above all it is necessary to clarify what the general characteristics of the problems that we intend to resolve are: we will be dealing with problems of linear elasticity in a static environment. Further, we will presume the validity of the hypothesis of small displacements, thus allowing us to write the equilibrium equations with respect to the undeformed configuration (in many cases this is not possible: think of a cable in tension due to the force of its own weight).

In these terms, the problem is thus: *given a body or a set of bodies that are constrained in various ways, subject to a set of forces applied almost statically, constructed with materials that satisfy the theory of linear elasticity, and supposing that the displacements of the system are small enough to allow writing the equilibrium equations with respect to the undeformed configuration, what is the response of the system to the application of forces, in terms of displacements, stresses and strains?*

Complementary to this, but not less important, is the next problem: *Is the structure capable or not of resisting the displacements, stresses and strains consequent to the application of forces?*

A first question arises spontaneously: What must be done to describe correctly the problem to be solved? Then: What are the successive steps to be taken to arrive at its solution?

Here is the briefest possible summary.

The finite element method consists in dividing the structure into parts, on the assumption that it is possible to describe, by means of simplified rules, the displacements, stresses and strains within each part (which becomes therefore a *finite element*). The application of these rules requires only the knowledge of the displacements of a few points that belong to the elements (the *nodes*).

The loads applied to the structure are then transformed into forces and couples that act on the nodes. Once this is done, equilibrium is imposed on all nodes of the structure (or, equivalently, the total potential energy is determined to be stationary in the equilibrium configuration; we shall see better in what follows what this really means).

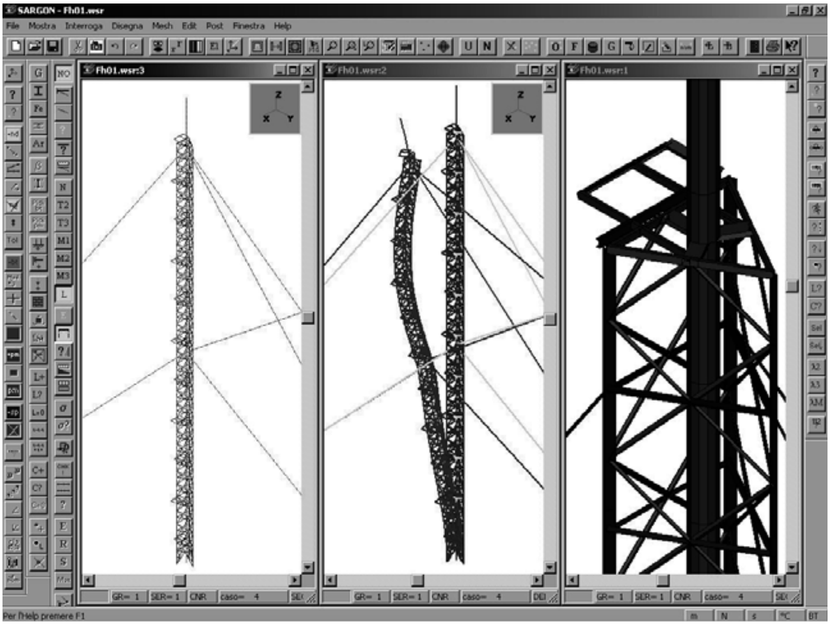


Figure 2-3: Wireframe, deformed and solid representations of a braced structure (finite elements model created with SARGON by Eng. Paolo Gustin, Milan, reproduced with permission).

The presumptions are:

1) the external forces applied to the nodes and the reactions of the distorted elements of the nodes are in equilibrium;

2) the reactions of each element on its own nodes depend on the relative displacements of the nodes of the element itself; these are evaluated on the basis of the simplified rules that typify the finite element in question (*the astrological principle).*

Bearing these in mind, in order to satisfy the equilibrium of all nodes it is necessary to solve a system that has as a known term the vector of forces and couples applied to the nodes (**p**), and as an unknown vector the set of the translations and rotations of the nodes (**u**). The coefficient matrix of this system (**K**) is called the *global stiffness matrix*.

$$\mathbf{Ku} = \mathbf{p}$$

Once the system is solved, and thus the displacements of the nodes are known for each element, on the basis of the simplified rules that characterise the element in question, we know what happens within the element itself in terms of displacements, strains and stresses. All that remains to be done is to save these results in the appropriate files for all the elements that make up the structure.

♣

In what we have just discussed, it is fundamental to understand the fact that in displacing any node whatsoever, (re)actions are communicated to nodes both nearby and very far away, if these are connected to the node of origin by the elements. Imagine being in the last car of a train: without warning you can be thrown against your seatmate. Effectively all the reactions of all the other cars are released into the last car, reactions due to the fact that the locomotive, two hundred metres ahead, has been jolted.

This extremely brief synthesis requires an in-depth explanation. In order to do this, we will examine one by one the necessary steps from start to finish.

2.2 THE PRINCIPAL STEPS

In order to describe the basics of what normally takes place in a finite elements software program, it is useful to state the following assumptions:

a) the problem can be described by means of a unique set of equations (in other words, it is a problem concerning plates, membranes, solids, etc.);

b) a single type of finite element is used.

In what follows we shall see how this scheme can be generalised to treat the cases in which these two assumptions do not hold.

This said, the principal steps are:

1) Choosing the most suitable mathematical model for the problem at hand;

2) Choosing the most suitable type of finite element for the problem at hand;

3) Subdividing the structure into finite elements (meshing);

4) Assigning the material and geometrical characteristics to the finite elements;

5) Assigning the constraints;

6) Describing the loads for each load case;

7) Renumbering of the nodes and creation of vectors;

8) Assembly;

9) Solution;

10) Stress recovery;

11) Output and **interpretation of the results**.

Each of these steps is fundamental and indispensible: the steps in bold are performed by the engineer, while the others are performed automatically by the computer. Step 3 appears in italics as a reminder that today it is partly done automatically by software programs, in accordance with rules that are in any case established by the engineer.

The requirement that steps 1-6 and 11 are performed by the engineer indicates that the FEM is still not completely foolproof. It is a fact that in each of these steps there exists the possibility that errors will be introduced that can result in unreliable output or can even make it impossible to solve the problem.

2.2.1 CHOOSING THE MATHEMATICAL MODEL

Strictly speaking, all problems of elasticity are three-dimensional problems, and the equations that are applied are the general ones of the theory of elasticity. However, in a great many cases the problem is formulated so as to make possible the use of one of the various approximate or specialised theoretical treatments, making many simplifications possible. We can briefly list the theories that are most often used:

- **technical beam theory;**
- **plane stress;**
- **plane strain;**
- **axial symmetry;**
- **thin plate theory (Germain-Kirchhoff);**
- **thick plate theory (Mindlin-Reissner).**

The choice of one of these models is inevitably linked to approximations, which must be correctly evaluated by the engineer. For example, the condition of a slab might possibly quite closely resemble that of a thin plate, while on the other hand this is highly unlikely in the case of a foundation plinth. The good engineer should know how to deal with the effects of the choice of formulation.

The errors due to an inappropriate model remain unaffected by a variation in discretisation, and do not generally depend on the type of finite element initially chosen. No matter how many elements are used to subdivide the plinth, if the plate element is not the appropriate one, the calculated results will only vaguely approach the "exact" ones. Many inexperienced users do not realise this and, for example, believe that they can study a contact problem with beam elements a micrometre long.

There are two reasons why more general elements – that is, elements that subdivide the solid into small solids (brick elements) – are not used. One is because they involve a greater quantity of calculations that have to performed, and a larger volume of data that has to be dealt with. The other is that the answer would naturally be more complicated, even when the part of the solution that interests the engineer is coherent and organised: no one would create a three-dimensional model to evaluate the axial forces of a truss.

Although this may appear obvious, it should be recalled that, as mentioned, many errors in modelling derive from an inappropriate use of the theories listed above. Not all users of finite elements can provide a ready response to the question, "Should a membrane or a plate be used?".

Chapter 3 contains a summary of and helpful observations about the models listed above.

2.2.2 CHOOSING THE MOST SUITABLE TYPE OF FINITE ELEMENT

Once the theoretical context of the problem, and thus the equations that must be solved, have been established, it is necessary to choose the type of finite element to be used in performing the analysis.

The software programs most widely used allow little choice in the finite elements: they usually propose at most two kinds of elements, and the user must choose one. Programs that are more refined (but also more costly and difficult to use) provide complete freedom of choice within an ample range of possibilities.

For technical beam theory, the choice is between the truss element, which models only the axial behaviour of the beam, or the more general beam element, with six components of stress. Although these elements are quite distinct from each other and their use fairly intuitive, we have found that inexperienced professionals using the FEM sometimes confuse them. It suffices for now to note that – as was the case in the choice of the model – the erroneous or inappropriate choice of finite element has irreparable consequences: the model can be vague, or worse, whole parts of the solution can be lost, so that the final calculated result is quite distant from the exact one. An extreme example is a Vierendeel girder whose horizontal members are modeled with beam elements, and the verticals modeled with truss elements: the entire typical stress regime of a girder thus constructed (as regards the bending of the struts) would be lost.

The choice of the correct finite element usually involves problems of either plane stress or plane strain or plate-shells. In this last case in particular, the presence of many kinds of elements makes this a delicate problem indeed. For plane conditions the choice is often between elements with three nodes (constant strain) and those with four nodes (linear strain), with or without drilling degrees of freedom.

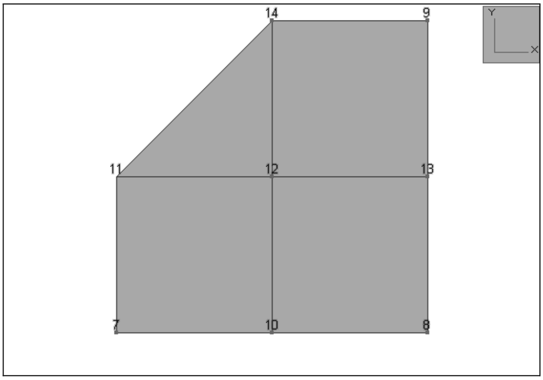


Figure 2-4 : 4-node quadrangular and 3-node triangular elements.

In an actual problem various kinds of elements can be used in different zones of the model. Indeed, the good engineer takes advantage of this possibility as he sees fit, with the aim of optimising the model. Chapter 11 deals with the elements most often used.

2.2.3 MESHING

Once the appropriate model and finite elements have been chosen, it is necessary to divide the structure into parts, or *finite elements*. The *mesh* cannot be defined randomly, but must be created in accordance with a given set of rules.

The first rule concerns the fact that the shapes of the individual finite elements must be compatible with their type. For example, it is not possible to represent an arch of curved elements using ordinary (straight) beam elements because beam elements must have a rectilinear shape. In addition to the requirement that their shape be possible, the finite elements must not be overly distorted, or present internal anomalies.

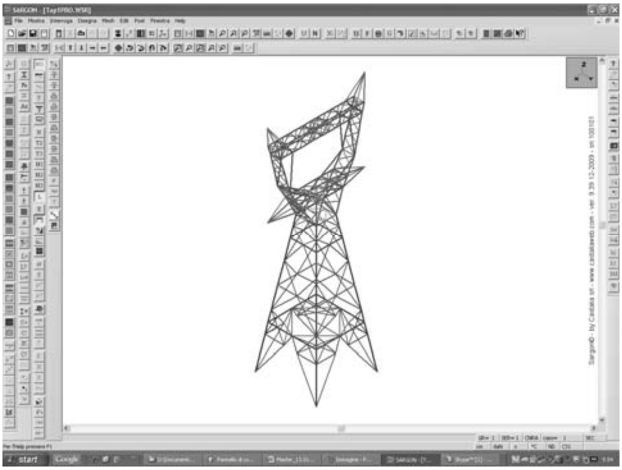


Figure 2-5: The mesh of an electricity pylon (Sargon model by Eng. Verrillo, reproduced courtesy of the engineering firm Colombo S.p.a.).

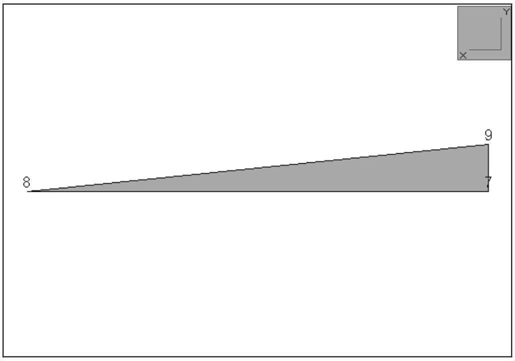


Figure 2-6: An excessively distorted triangular element.

For example, a triangular element with angles 89.5°-89.5°-1° is inefficient because it is too distorted.

To identify the nodes in the model, a progressive number is assigned to each one. The set of nodes in a model and of their coordinates with respect to a given reference system is called the *coordinate table.*

There cannot be more than one node with a given number.

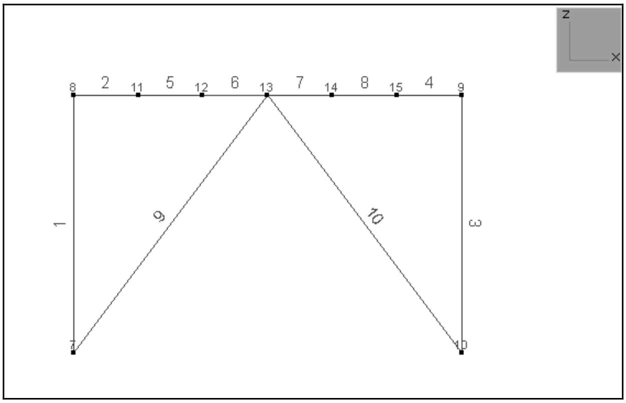


Figure 2-7: Numbering of nodes and elements.

There can, however, be two different nodes occupying the same point in space; these are called *double nodes.*

The finite elements can be one-, two- or three-dimensional. Some finite elements have the form of a point (0-dimensional), but this will be addressed later.

Each one-dimensional element is defined by a given numer of nodes and by a type. Identifying the type makes it possible to connect the nodes in order to describe the element's geometry properly. Rectilinear elements (the most frequently used) generally have an axis defined by two nodes. Thus the form of the element is defined by two nodes.

Two-dimensional elements (such as plates or membranes) are defined by a given number of consecutive sides. In addition to the two extremes, each side is defined by a given number of definable points, called *intermediate nodes* (typically midside nodes). It can be seen that, when the type of element and the nodes on the sides are known, the element is unequivocally defined, thus when the type of element is known, its geometry is unequivocally defined by its nodes.

Three-dimensional elements are defined by a given number of faces delimited by sides defined by nodes. Each face shares a side, defined by two or more nodes, with another face. When the type of element is known, each face is unequivocally defined by sides, and thus, it can be said definitively that the shape of each element whose type is known is unequivocally determined once the nodes of the element are known.

Thus, in general, given a type of element (beam, truss, CST, etc.) *its form is unequivocally defined by its nodes and their ordering.*

The table for each element of a given type specifies the ordering of its nodes, and is called the *connectivity table.*

Two elements are said to be *connected* if they share at least one node. However, in some cases the connection can be incomplete or faulty.

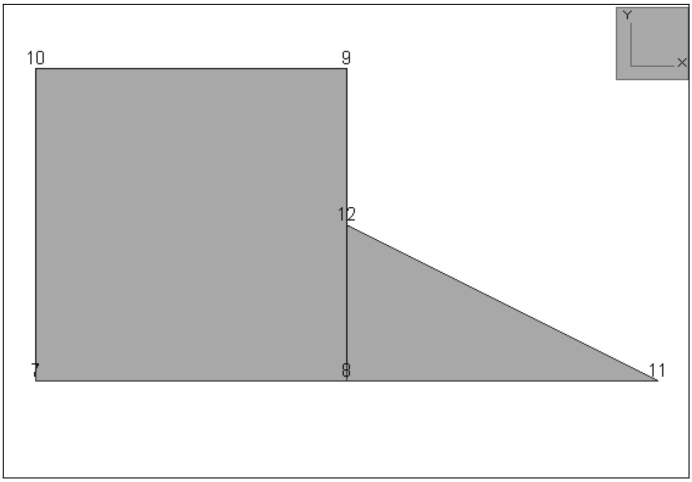


Figure 2-8: An example of an faulty connection between two elements: the sides are partially connected.

One side of a two- or three-dimensional element is said to be connected to the side of another two- or three-dimensional element of the same type[2] if and only if the nodes of the side of the first element are also the nodes[3] of the side of the second element. If only some of the nodes are identical, the side is said to be partially connected, or equivalently, partially disconnected.

A face of a three-dimensional element is said to be connected with the face of another three-dimensional element if all the nodes of the face of the first element are exactly the same as the nodes of the face of the second element. If only some of the nodes are identical, the faces are said to be partially connected,

[2] The connection between elements of different types will be covered in a later chapter.

[3] It should be noted that it was not said that the nodes had to coincide: in fact, two nodes can be coincident but different, that is, they can have different progressive numbers. In such cases, the nodes are called *double nodes*. Later we will see how double nodes can constitute a serious problem if they occur by chance and not on purpose.

or equivalently, partially disconnected. If two faces do not have any nodes in common they are said to be disconnected.

The above statements indicate that elements that have sides with different numbers of nodes can only be partially connected. Likewise, elements that have faces with different numbers of nodes can only be partially connected.

If two elements are partially connected, they are generally not congruent. Thus partial connections should be avoided, unless the model is intended to simulate cracks, construction joints or, more generally, situations which present discontinuities.

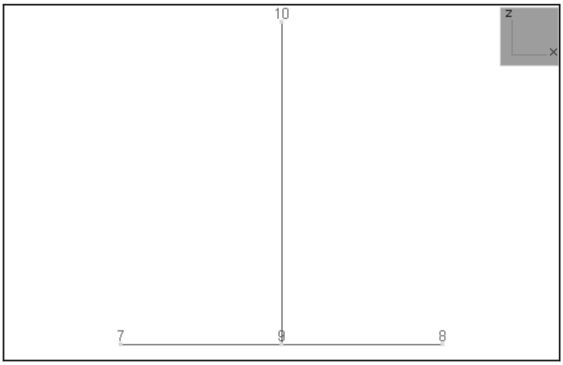


Figure 2-9: Facing elements. Note that since the elements are not numbered it is impossible to understand whether they are connected or not.

If a node of one element lies on the edge of another element without having a connection with it in that point, the two elements are simply said to be *facing elements*. Elements that are simply facing are not connected, and thus are not congruent.

As has been seen, a mesh is not a drawing, but is rather a set of relationships between nodes and elements satisfying requirements of congruence.

The subdivision of the structure into finite elements is one of the trickiest steps in the entire procedure, since in addition to satisfying all of the requirements of regularity that a mesh must satisfy, the elements must be sufficiently small where there is a larger gradient of displacements or forces (which is somewhat like decreasing shutter speed in order to photograph rapidly moving objects).

A priori, it is not possible to say with certainty whether a mesh is sufficiently fine or not (although in the case of a coarse mesh it is easy enough to predict that the results will be unreliable). Often in order to decide if the discretisation is sufficient a new mesh is constructed that contains all the nodes of the original mesh plus a new set of nodes, in order to compare the results obtained by the two meshes, the one less and the other more refined. If the differences between the results are modest overall, it is reasonable to consider the results acceptable; if, on the other hand, the differences are notable, then a third mesh, denser than the second, is necessary, and so on, until the desired precision is achieved; this is known as *remeshing*. Actually, some software programs are capable of executing remeshing completely automatically, because they memorise the geometry of the problem not only in the form of nodes and elements but also in the form of geometric primitives (solid modelling). If this is not available, remeshing requires the painstaking work of remodelling.

In any case, the subdivision of the structure in elements results in a set of nodes with a coordinate table, and a set of elements with an connectivity table. These two tables, which can be written in various ways, constitute the most part of the input file for all finite element software programs.

2.2.4 ASSIGNING MATERIAL AND GEOMETRIC CHARACTERISTICS

Once the form of the finite elements has been defined it is necessary to establish the geometric and material characteristics of each individual element. "Geometric characteristics" concerns the fact that for some finite elements (the majority) it is not sufficient to specify the form in order to know with precision which material part the element simulates. In many cases it is necessary to furnish additional data, such as the thickness of the plate/-shell or membrane elements, areas or moments of inertia for beam elements, and so forth. It should be noted that in specifying these characteristics it is not as important to simulate the physical reality, as it is to depict the behaviour of an element correctly. For example, for a stiffened slab, the thickness, which must be furnished if a plate element is used, only has a numerical value: it is not the true thickness of the real slab, which is undefined. It is also for this reason that it is not always easy to input data regarding thicknesses, moments of inertia, and cross-sectional areas.

A second difficulty is related to the possibility of making errors (including banal errors of calculation or typing). The latest software programs are furnished with databases that make it possible to attribute characteristics to elements by calling up the appropriate files. Earlier programs of the 1970s, which require writing a formatted input file, can lead to problems.

In addition to geometric characteristics, material characteristics must be attributed to the elements as well. In theory, each finite element might have its own material. More often there are groups of elements with similar materials which simulate different parts of the structure (for example, in a mixed structure of steel and concrete). It is generally for this reason that material characteristics are defined for a *group* of elements of a similar type. The definition of material in the context of the FEM, and in the software programs using the FEM, is not, as might be thought, similar to the theoretical definition of the material (isotropic, anisotropic, orthotropic with all of the constants involved): it has a conventional meaning, due to the fact that by definition certain finite elements make sense with particular types of materials, of which they in any case use some but not all parameters. For example, a rod element uses only Young's modulus of elasticity relative to the direction defined by the axis of the rod. In this case the possible anisotropy of the material is not relevant, since we are only interested in the material's behaviour along the axis of the rod itself. Further, it is not necessary to specify either Poisson's ratio or the shear modulus of elasticity, because the formulation of the element is concerned with neither of these. On the other hand, the material's weight density, which is in fact usually required, can be helpful in evaluating the weight or mass of the element. Instead, in the case of the ordinary beam element, it is implicitly hypothesised that the material is homogeneous and isotropic, even if it is relatively easy to deal with an element whose material is, for example, orthotropic. The element is simply designed in this way.

Generally speaking, the software programs do not cover the full range of the set of possible choices: they furnish some of them, most often the best known or most used. It should also be noted that a correct use of material and geometric constants often makes it possible to model situations that it would appear cannot be modeled. For example, in the case of a beam of orthotropic material, we need only deal with the moments of inertia appropriately in order to model this situation correctly.

This discussion makes it clear that the phase of assigning material and geometric characteristics to the elements requires great care and refinement, and even recourse to *tricks of the trade,* in order to obtain meaningful modelling: it is thus of primary importance to understand completely not only how each individual finite element functions, but also how it has been implemented in the software program being used. For now it is sufficient to repeat that such characteristics are intended to be only indirectly realistic, in order to make it possible to deal with the behaviour of the elements as we wish. Their object is to *model*, not photograph. They should therefore be thought of more as the commands needed to fly a model airplane by remote control than as precise references to theory.

2.2.5 ASSIGNING THE CONSTRAINTS

Once the finite elements have been completely defined, it is necessary to specify the constraints to which the structure is subject. In the context of the FEM the constraints always and only concern the nodes. This does not mean that more general, real situations cannot be modeled: it simply means that in order to do this with the FEM it is necessary to define constraints for a certain number of nodes. In some cases, as we shall see below, in order to model some constraint situations completely it is necessary to add elements as well as to constrain nodes, but again the concept of *constraint* only applies directly to the nodes.

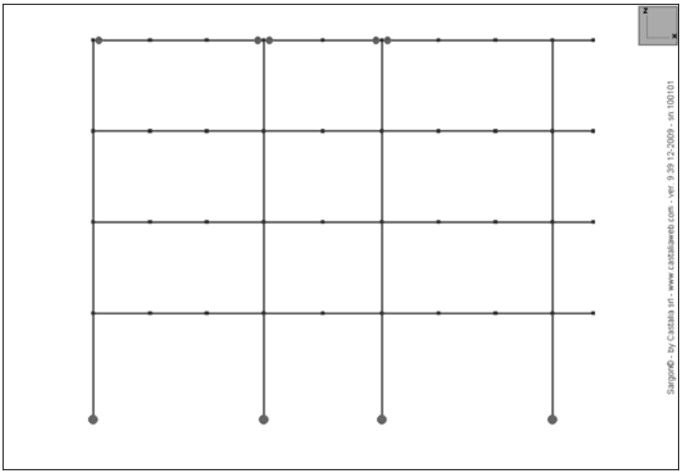

z
x
Sargon© - by Castalia srl - www.castaliaweb.com - ver. 9.39 12-2009 - sn 100101

Figure 2-10: Planar frame (Sargon). The solid dots at the bottom indicate constraints; the smaller solid dots at the top indicate end releases; the squares indicate the presence of a node.

The constraints of the nodes can affect one or more of the node's possible movements. In general, a node has six degrees of freedom, that is, six different possibilities of movement:[4] constraining its possibility of movement, these can be reduced to five, then four, and so on, until all possibility of movement is excluded. The six degrees of freedom typical of a node are the three translations in directions *x*, *y* and *z* of the global reference system, and the

[4] In some cases the nodes begin with less than six degrees of freedom. These are cases involving a kind of problem that does not require all six degrees of freedom. For example, if a planar rectangular structure is studied using truss elements, each node has at most two degrees of freedom. If the problem involves a flat plate, each node has at most five degrees of freedom, and so on. In these case the constraints would further limit the nodes' possibility of movement.

three rotations about straight axes such as x, y and z. Constraining one of these components means making it equal to zero.[5]

The number of constraints placed on nodes in the structure directly influences the dimension of the mathematical problem to be solved. If all of the nodes were constrained there would be no problem at all to solve (nor would there be, in general, forces or deformations *in the model*). The mathematical problem in question is a system of linear equations, having a number of unknowns equal to the number of degrees of freedom of all the nodes of the structure. Each component of constrained displacement means, as we shall see, that there is one equation less.

If there are insufficient constraints or they are ineptly positioned, the problem can turn out to be impossible to solve, or useless. It becomes impossible to solve when rigid body displacements are present, that is, zones of the structure that can move without giving rise to any internal forces. It becomes useless when the constraints are placed so that they distort structural behaviour, making the model inadequate. Unfortunately, both of these cases occur frequently. The second is most to be feared, because it is possible to obtain absurd results without realising it.[6]

2.2.6 DESCRIPTION OF LOADS

The *loads* – or more generally, the *actions* – present in a structure are usually grouped into *load cases*, meaning that all the forces associated with a load case act contemporaneously. The response of the structure will therefore be, in obedience to the principle of the superposition of effects, the sum of responses of the structure to the individual forces. From this it follows that in order to maintain a distinction between responses due to different causes, they should be put in different load cases.

Actions can be of various types: they can be forces or couples applied directly to points of the structure where nodes are present (*nodal forces*), or they can be forces or couples applied at points other than nodes in the finite elements, such as on their edges (*distributed loads*).

There are also distributed loads of pressure applied to the elements (for example, a uniform load on the axis of a beam element). One particular kind of

[5] The term "constraint" is also used to refer to the reduction of the node's possibility of movement. For example, a node connected to a spring would be considered, in this sense, constrained. Here the term "constraint" means the annulment of one of the components of generalised displacement.

[6] It goes without saying that a good engineer will always be aware, in analysing the results, that "something's not right". Unfortunately, the reality is that the results are not always analysed ...

action is the thermal load, which in general can be either nodal or distributed. Finally, another type of action is fixed displacement (for example, the settling of constraints). Fixed displacement and thermal loads are called *coactions*.

As will be made clear in what follows, all of the actions have to be translated, by means of the appropriate procedure, into nodal forces or couples, with the exception of the fixed displacements: rather than being made equal to zero, the displacement is made equal to the value assigned.[7]

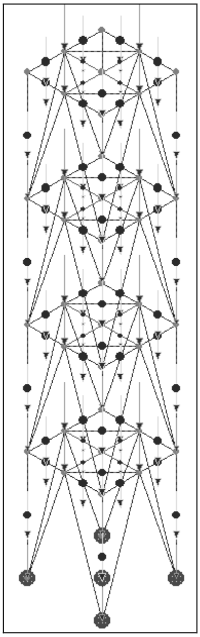


Figure 2-11: Various types of forces applied to elements and nodes.

In all events, the load is made to correspond to a *nodal force vector*, which sums up the set of effects of actions on the nodes (even when the actions are distributed). This vector is the known term of the system of equations that solves the structure for the given load case, and to it corresponds a solution vector that contains the displacements of all the nodes. Thus, a different solution vector corresponds to each load case. Generally speaking, if *n* load cases have been defined, it is necessary to solve *n* linear systems all having the same coefficient matrix and different vectors of known terms: the nodal force vectors corresponding to the load cases in question.

[7] In truth, in the technique most widely used by software programs, the fixed displacements are also transformed into applied forces or couples.

2.2.7 RENUMBERING THE NODES AND CREATING THE VECTORS

Once the load cases have been created and the applied loads assigned to them, the problem is ready to be solved. The first step traditionally performed by all good finite elements software programs is the renumbering of the nodes, and in some cases,[8] the renumbering of the elements. The object of renumbering is to arrive at a system of equations that is identical to the original one but easier to solve because the coefficient matrix has a much smaller *band*: we will see what this means in greater detail in Chapter 8. In solvers of the newer sparse matrix type, the renumbering phase is replaced by the phase of the analysis of the coefficient matrix of the solving system, necessary in order to evaluate the voids and solids,how many non-zero elements there are, or the *sparseness*. Some sparse matrix solvers still require some kind of renumbering in order to optimise the space occupied in memory.

In terms of the time required for calculation, the effort required for renumbering is always more than compensated for by the advantages it brings; this is true even when the solution proposed by the renumberer is not optimal (the optimal solution is that which gives the minimum band in absolute).

Renumbering is done with the use of criteria adopted from graph theory and is an extremely delicate and important step, because its performance – especially its incorrect performance – can lead to insurmountable problems during the solution phase.

Renumbering requires updating the connectivity and coordinate tables: in effect, a node which had number *n* before renumbering will have number *m* after and, consequently, where number *n* occurred it is necessary to write *m*.

After renumbering, the generic solver is able to create the vectors of the unknown terms and the sets of vectors of known terms. In order to do this, usually a table is constructed that associates a given displacement of a given node to each equation of the system, that is, to each line of the coefficient matrix. One such line in the table is shown in Table 2-1.

This line tells us that line 567 of the coefficient matrix refers to translation *x* of the node numbered 113 (which before renumbering was 1280). The number 0 corresponding to translation *y* and rotation *y* indicates that there are no equations associated with these, since the two displacements are impeded by the constraint. Finally, the rotations of node 113 in relation to the *x*- and *z*-axes are found at lines 569 and 570 of the system. This table is often called the *vector ID table*, from the ID symbol associated with it in the SAP source.

[8] The renumbering of the elements is performed by frontal solvers.

NODE	OLD	Tx	Ty	Tz	Rx	Ry	Rz
113	**1280**	**567**	**0**	**568**	**569**	**0**	**570**

Table 2-1. Nodal degrees of freedom numbering

Now we can see that after renumbering it is possible to associate biunivocally a line in the coefficient matrix to a component of unknown displacement, that is, to a degree of freedom.

The vectors of known terms correspond to the nodal forces due to loads applied directly at the nodes, as well as to distributed loads. There are as many vectors of known terms as there are load cases. The part of the force due to loads directly applied to the nodes can be constructed during this phase: if more than one force is acting on the same degree of freedom in the same load case, these are summed (in obedience to the law of superposition of forces). The part of the nodal forces due to distributed loads is calculated and summed during the next phase, that is, during *assembly*.

2.2.8 ASSEMBLY

Assembly is perhaps the most difficult phase to understand: in simplified terms we might say that assembly is the procedure during which the coefficient matrix of the system (*global stiffness matrix*) is created, and which completes the vectors of known terms due to distributed loads. In effect, if this is what happens here, it is necessary to understand only *why* this is done. There are various ways to illustrate what assembly means: we will use the *static approach* and the *energy approach*. Both of these approaches will be further described in later chapters, and here we will limit the discussion to a brief overview.

In the static approach, assembly is the procedure that calculates and sums, node by node, the forces that act on the node itself due to the motion of other nodes and to applied loads. The forces arrive at the node in question by means of the elements connected to it: the forces due to the displacements of connected nodes are called *reactions*; forces due to applied loads are called *actions*. The single term K_{ij} of the global stiffness matrix is equal to the force required in the direction of the degree of freedom i due to the existence of a unitary value of displacement of the degree of freedom j.[9] If the node associated with degree of freedom "i" through the vector ID appertains to elements that are not connected

[9] For example: if I rotate the end of a beam, what forces and couples have to be applied to the ends in order for there not to be other displacements (either translations or rotations) other than those that I have applied?

in any way to the node associated with degree of freedom j, then $K_{ij} = 0$.[10] This case occurs quite frequently, and in fact matrix **K** is generally rich in null terms (a *sparse* matrix). The sum of all the forces acting on degree of freedom i, due to the movement of the nodes (that is, the sum of the reactions) is given by

$$R_i = \sum_j K_{ij} u_j ,$$

where the sum is extended to all degrees of freedom of the structure (j=1, *n*).

Let f_i be the sum of the actions on degree of freedom i. In order for equilibrium to exist, the action has to be equal and opposite to the reaction, or in other words:

$$R_i = f_i$$

and thus it is necessary to solve the system that has as unknowns the *n* displacements u_i

$$\sum_j K_{ij} u_j = f_i . \qquad [2\text{-}1]$$

♣

In the energy approach the assembly is seen essentially as the procedure that sums the contributions of each element to the elastic strain energy,[11] and to the load potentials (with the contribution of the distributed loads).

In effect, the strain energy of the entire structure is given by the sum of the strain energy of the parts of which it is composed, and this simply on the basis of the additive properties of integrals. Each of the contributions Z_e to the total strain energy is expressed as a function of the vector of unknown nodal displacements, in the guise of a quadratic form associated to a square matrix whose order is equal to the total number of the degrees of freedom, or in other words, of the type

$$Z_e = \frac{1}{2} \mathbf{u}^{\mathbf{T}} \cdot \mathbf{K_e} \cdot \mathbf{u} , \qquad [2\text{-}2]$$

where **u** is the vector of unknown nodal displacements, of *n* dimensions; *e* the generic element; and $\mathbf{K_e}$ the *expanded stiffness matrix* of the individual finite

[10] What forces must be applied to the right brake of a bicycle to keep it in balance when the left brake has been used? None, obviously.

[11] Besides the theory of linear elasticity, it is always possible to use variational principles, by which it is possible to generalise the ideas set out here.

element. The reason why the strain energy of the individual finite element depends on a matrix, as was expressed in the preceding equation, will be explained later. If we accept eq. 2-2, the strain energy for the entire structure becomes

$$Z = \sum_e \frac{1}{2}\mathbf{u}^\mathbf{T} \cdot \mathbf{K_e} \cdot \mathbf{u} \equiv \frac{1}{2}\mathbf{u}^\mathbf{T} \cdot \mathbf{K} \cdot \mathbf{u}.$$

An analogous principle holds for the load potentials. In fact, summing the contributions of each element, we can write

$$W = \sum_e \mathbf{u}^\mathbf{T} \cdot \mathbf{f_e} \equiv \mathbf{u}^\mathbf{T} \cdot \mathbf{f}.$$

Using the *principle of minimum potential energy* Π, where

$$\Pi = Z - W$$

it can be seen that this principle is satisfied when $\mathbf{u}$ satisfies the matricial equation (that is, the system of algebraic equations)

$$\mathbf{K} \cdot \mathbf{u} = \mathbf{f}, \qquad [2\text{-}3]$$

where **K** is exactly the sum of the expanded stiffness matrices of the individual elements.

Regardless of the way in which it is obtained, the system of equations 2-3 gives the vector of the unknown nodal displacements, with which, on the basis of the *astrological principle* described at the beginning of this chapter, it is possible to obtain the stresses and deformations within the elements.

Later we will return to the assembly in greater detail. Here it is sufficient to say that the system 2-3, in both forms in which it has been written, is the discretised expression of precise physical principles, and assembly constitutes that system on the basis of rules that rely on those principles.

2.2.9 SOLUTION

Once the matrix of the system has been constituted and the vectors of known terms completed with the effect of the distributed loads, a generic program for calculation is used to solve the system, calculating as many vectors of *n* unknowns as there are load conditions. Each of these vectors contains *n* values, representing the displacements of the nodes according to the *n* degrees of freedom of the structure. Therefore, by the end of the solution process, how much each node has moved in each load condition is known. The displacement is expressed in reference to a global system of coordinates, and is thus an absolute displacement: the *relative* displacement of two nodes can easily be obtained from two absolute displacements.

Later we will discuss solutions of sparse systems, symmetric systems, and those with a large number of equations. Here we shall only say that the actual solving phase can create problems of RAM and disk space that have been addressed in many ways by the solvers now available. In some cases (and frequently with some solvers) the solving phase is even faster than the successive phase of *stress recovery*, so that by now it seem excessive to underline the difficuly of this task: today there is little difference between solving a system of a thousand equations and one of ten thousand, and the larger system is only slightly more unwieldy.

If errors were made during the phase of assigning the constraints, these will become evident, where this is possible,[12] during the solving phase. This is the situation where "something's not working", "the program is giving me some strange messages", and so on. Often the non-expert will find himself outside of the solving program and not know why. Without digressing into the issue of the "user-friendliness" of the software, as interesting as that is, we will only say that when this happens, it is very often due to problems of a lack of constraints. In order to make the problem explicit, it is necessary to assemble the global stiffness matrix, because that is the only way to verify that it cannot be inverted, and thus that the system has no solution.

When the solving phase is complete, the displacements are memorised on disk, and the next phase, *stress recovery*, begins.

2.2.10 STRESS RECOVERY

Usually the expression *stress recovery* indicates the procedure that conducts the nodal displacements to the stresses within elements. In a certain sense this procedure can be seen as an assembly in reverse, a kind of *disassembly*, in that it goes from general data proper to the entire structure to that proper to the individual finite elements, which at this point are considered to be independent of each other, small microcosms in their own right.

In stress recovery a decisive role is played by the *astrological principle*: after the system has been solved, the displacements that determine what takes place within the finite element are known: it is sufficient to apply the simplified rules using these displacements in order to know, within the limits of the FEM, what

[12] Bear in mind that errors in constraints are not always made explicit by an unexpected halt in the program: in these cases only the skill of the engineer stands between erroneous results and their being used in practice.

the stresses, strains and displacements of all of the points appertaining to the element are.

Because the *astrological principle* makes it possible to know the displacements of the nodes of the element, the first thing that must be done is to extract from the vector that gives the displacements of all the nodes of the structure the smaller vector that gives the displacements of the nodes of the individual element. Frequently, the *astrological principle* uses displacements expressed according to a particular system of reference, the element's system of reference, or *local system*: therefore, once the vector that gives the displacements of the nodes of the elements has been constructed, it must be transformed to arrive at those displacements in the local system.

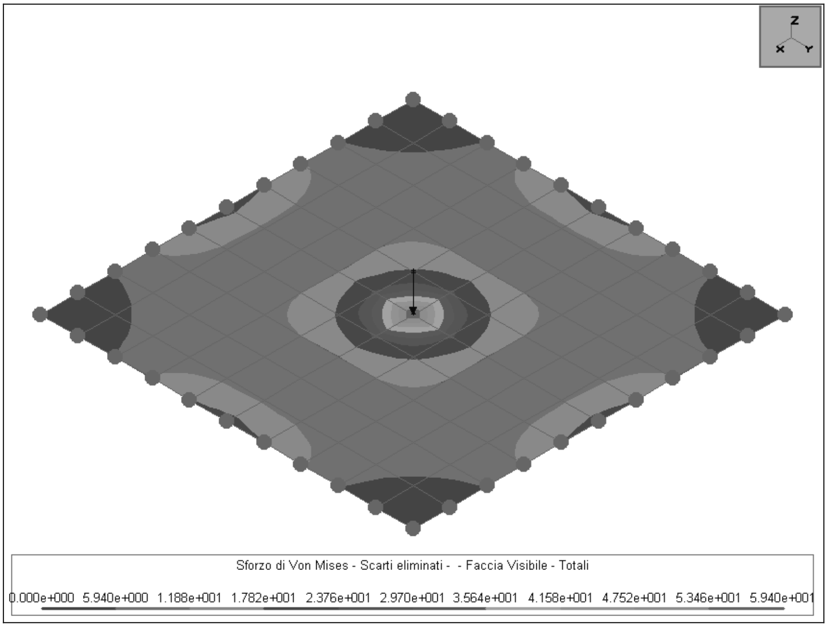


Figure 2-12: Rendering of the Von Mises stress values on a plate clamped on four sides with a central load (Sargon model, 10x10 element mesh).

Once the transformation has been accomplished, in order to know the stresses within the element itself, it is only necessary to apply to the nodal displacements of the element the simplified rules proper to them. Generally speaking, the stresses are calculated at a predetermined number of points in the element, which can more or less coincide with the nodes of the element. All of these stresses are then memorised, element by element, and constitute the result of stress recovery, as well as the large part of the final result as a whole. Obviously, it doesn't make sense to try to determine displacements or stresses outside of the element using the rules proper to the element itself and to its nodal displacements: would you ever cast a horoscope for Tom using the positions of Dick's planets?

2.2.11 OUTPUT AND INTERPRETATION OF THE RESULTS

The output phase, automatically performed by the software program, concludes the solution of *a* problem, but not the phase of analysis of *the* problem. Indeed, in output we arrive at the beginning of the most interesting part of the problem.

Output is usually given in the form of two sets of files: ASCII and binary. The results, that is, the displacements and stresses, can figure in both sets of files, but generally the results in the ASCII file are unusable. This is mainly because the results of a finite element analysis are only rarely the final results desired, but even if this were the case, they would still be unusable because it is not humanly possible to manage thousands and thousands of numbers organised in huge tables efficiently. Often the desired result can be obtained indirectly from the results of a finite element analysis, which must be in any case validated.

However, the ASCII file should be checked to make sure that there are no "warnings" or other anomalies that might signal a problem during the course of reading the input data, during assembly, or in the solution itself. Often the diagnostics are small, easily managed ASCII files.

It can thus be seen that the most interesting results are those written in the binary files, and this is because they can be further processed with ease, in order to give them a graphic interpretation, and permit immediate access to significant values. Today the phase of *graphic rendering* cannot be dispensed with.

In close relationship to the preceding is the very important step of *checking results*, which provides us with a guarantee that the solution respects the characteristics of the problem to be solved. It is thus necessary to: check the deformation in qualitative as well as quantitative terms; check the reactions of the constraints, which must be balanced according to the loads we want to impose; check the progress and intensity of measured stresses, which must not be contrary to the general rules (exceptions or special cases are always suspect), and so forth, checking as much as possible.

The danger of errors is ever present, the fact that reliable precision instruments are used does not guarantee that they are correctly used, or state of the art. Very often the outcome of these checks is a *return to step 5*, but sometimes it is necessary to go back to step 1. Of course, as more experience is gained this happens less and less frequently.

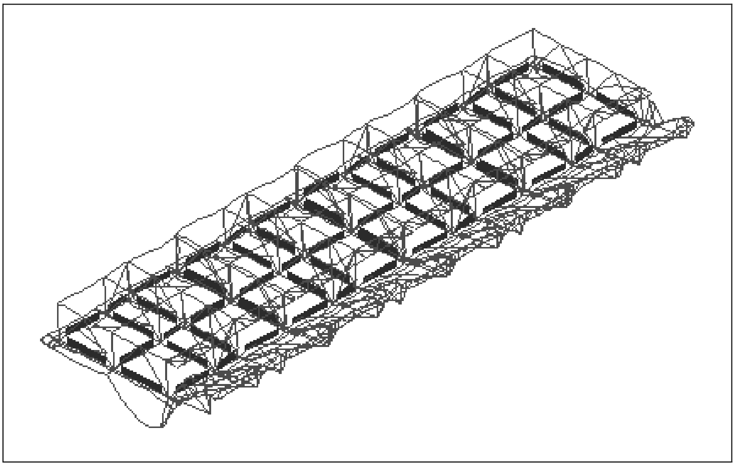


Figure 2-13: Verification of the deformation makes zones of large deformation evident.

3 ELASTICITY: SPECIAL FORMULATIONS

3.1 INTRODUCTION

The analysis of the stresses and strains of a solid can be performed in accordance with several theories that describe the behaviour of the material under the action of the forces applied. Our theory of choice is that of linear elasticity, the fruit of centuries of studies and research which described with increasing clarity what takes place within a solid as a result of the application of external actions. It continues to be the theory most widely used even today.

Within the limits of these pages it is not possible to provide a detailed explanation of all the concepts employed by the theory of elasticity, and thus we will briefly describe the concepts and conclusions, referring the reader to other texts when more in-depth explanations are desired. Appendices I and II contain a brief review and list of symbols used.

In general, the theory of elasticity refers to solids, that is, to three-dimensional bodies that are generic in terms of shape and applied actions.

In order to describe completely what takes place as a result of the application of forces either on the boundary (where the actions are known because they are directly applied, and the displacements are unknown), on the volume, or on both, it is necessary to know the displacements of each point of the solid as a consequence of the application of the forces, and the stresses and strains at each point of the solid. On the parts of the boundary where the displacements are assigned (that is, where constraints are present), it is also necessary to know the extent of the reactions exerted by the constraints of the structure.

Although this is far from a simple problem, it is one that has been solved, and for which there are closed-form solution equations: the theory of elasticity has been able to provide answers to all the questions and generate a set of coherent equations to relate all the various unknown terms to each other.

Although the general solution does exist (see Appendix II), it does not lend itself to each and every case. The presence of special conditions (which abound in practice) makes it evident that the answer sought has important regularities that a general kind of answer does not satisfy. The existence of these special cases has made it possible to arrive at theories in which the solution is simplified, losing some of its complexity.

Thus were born the theories of beams, plates, plane stress, plane strain and others, which will be described briefly in the sections that follow. However, it should be underlined (should there be any doubt) that these theories do not alter the more general theory, but are instead specific to some special cases, and introduce ulterior simplifying hypotheses. Such theories are extremely important

Structural analysis with finite elements
978-0-7277-4093-9

because, with only rare exceptions, constructive elements possess some kind of geometric property that, on the one hand, makes possible a simplification or economy of construction and, on the other – fortunately – leads to a simplification of the problem from a strictly analytical-mathematical point of view.

At least summary knowledge of these theories is indispensible for the correct application of the FEM. Also indispensible is a correct understanding of the fundamental energy aspects of the problem of elasticity. The following sections will provide a brief explanation of the theories as well as the fundamental energy aspects, summarising the principal hypotheses.

3.2 THE ENERGY ASPECTS OF THE PROBLEM OF ELASTICITY

The study of the state of stress and strain in a solid in equilibrium can be performed in a particularly clear way by means of principles that make use of the potential energy of loads and potential elastic energy (strain energy). Obviously, the other great form of energy that appears when we deal with mechanics – kinetic energy – does not appear when we deal with statics. Scientists of past centuries used the term *vis viva* for kinetic energy, to indicate that it was a form of energy *in progress*, so to speak. In contrast, potential energy is a kind of energy stored in the form of a position or of a configurational variation: a weight placed at a certain height, a spring shortened to half its length, and so on. The potential energy of loads or elastic potential energy can be transformed into kinetic energy (or other forms of energy) at any time, for example, by eliminating the constraints or loads that establish the original configuration of equilibrium.

In the theory of elasticity in statics, all energy expended by the forces enters into the body without dissipation or the appearance of other forms of energy. The elastic strain energy stored in a solid depends solely on its state of stress and strain, and since the strain and the stress can be calculated starting from the displacements, the elastic strain energy depends only on the deformed configuration assumed by the body, it is a so-called state function.

When an elastic body is subjected to forces, those forces perform a certain quantity of work, which is then stored in the body in the form of strain energy. In order to exclude the appearance of kinetic energy, it is hypothesised that the loads are applied *quasi-statically*, that is, so slowly that dynamic effects are negligible. Naturally, if final load P is to be applied, it cannot be applied all at once, because this would heighten the dynamic effects; the load has to be applied through infinite states of equilibrium, very slowly, so that it increases gradually. Thus during the course of applying a load up to P, *the force is not constant but varies.*

If instead load P is applied all at once, the system would begin to oscillate with respect to the equilibrium configuration, and these oscillations – in the absence of dissipation – would continue infinitely. In real systems the oscillations attenuate, and the equilibrium configuration that is finally reached is the same that would have been reached if the process of loading had been quasi-static.

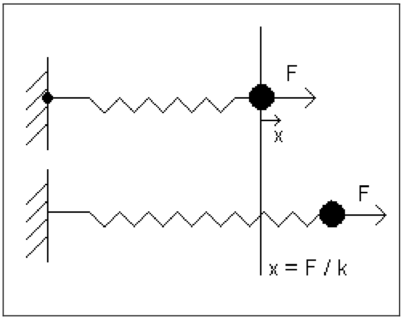


Figure 3-1: Spring of rigidity K with applied force F.

In order to fix these ideas, let's look at a simple spring of rigidity *k*, with a force *F* applied to mass m. Let *x* be the displacement of the end of the spring. The strain energy of the spring is $Z=kx^2/2$, while the potential of force *F* (presumed constant and applied all at once) is equal to –Fx, and both energies are state functions. At first glance one might think to set the work performed by force F (W=Fx) equal to the stored strain energy, setting

$$W=Z \qquad [3\text{-}1]$$

$$Fx = kx^2/2.$$

In fact, this reasoning is erroneous, because *if force F remains constant* the load process is not quasi-static and there has to be kinetic energy C. That is, there must always be

$$W=Z+C \qquad [3\text{-}2]$$

$$Fx = kx^2/2+ mv^2/2.$$

The system begins to oscillate with the centre in the future equilibrium configuration, equation 3-2 is satisfied at every instant, there is a continuous exchange between the various forms of energy, and in theory the oscillations would continue indefinitely.

In reality, the kinetic energy dissipates, and finally the spring stops in the final equilibrium configuration, from which it no longer moves. Why? Because in this configuration, given a very small displacement of quantity dx, the variation of work of the forces is exactly compensated for by the variation in strain energy *and there is no room for kinetic energy*, which must be null in order for there to be equilibrium.

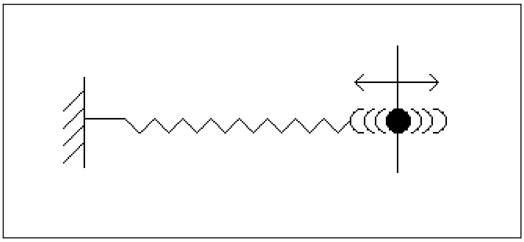


Figure 3-2: Oscillations about the future equilibrium configuration.

If we take the variation of energy[1] with the variation of the configuration (that is, by x), equation 3-2 becomes

$$dW=dZ+dC,$$

but if dC must be null, we necessarily obtain that

$$dW=dZ$$

or, denoting as Z-W the *total potential energy* Π

$$d(Z-W)=d\Pi=0. \qquad [3\text{-}3]$$

Equation 3-3 represents the principle of the stationarity of total potential energy: *in the equilibrium configuration the first variation of total potential energy that is obtained by varying the equilibrium configuration itself is null.* Note that in the case in which the configuration is the function of more than one variable (as happens in a solid), the configurational variation can take place in an infinity of different ways. Consequently, we must impose that equation 3-3 is valid *for all the possible configurational variations*, not just for some. In fact, if there were to exist some configurational variations such to permit the appearance of kinetic energy in violation of equation 3-3, the initial configuration could not be one of equilibrium. Thus, a system is in equilibrium in a given configuration if and only if the variation of total potential energy is null *for every possible configurational variation.* It should be noted that in the presence of constraints, the generic configurational variation has to respect the existing constraints.

Let's return to the example of the spring. The condition 3-3 occurs only when x assumes a particular value such that

$$dW=Fdx=dZ = kxdx$$

for every possible dx, that is,

$$(F-kx)dx = 0,$$

[1] For simplicity's sake we have used the symbol for the differential "d" and not that for variation "δ".

for every possible dx

$$x = F/k.$$

This is the only configuration with respect to which a small variation produces an identical variation of the work of the loads and of the strain energy.

It can be seen that, by using the conservation of energy, it is easy to obtain the velocity with which mass m moves through the configuration that will become that of equilibrium when all of the kinetic energy has dissipated. If v is the velocity of the mass, the kinetic energy that the mass has when it moves through the equilibrium configuration x=F/k, after having applied all at once the constant force F, is

$$C = m\,v^2 / 2 = W(F/k) - Z(F/k) = (F^2/k) - (kF^2 / 2\,k^2) = (F^2 / 2\,k)$$

and thus

$$v = \sqrt{\frac{F^2}{km}} = \frac{F}{m}\sqrt{\frac{m}{k}} = F/(m\omega)$$

where v is the velocity of the mass, and ω the angular frequency (rad/sec) of the oscillatory motion.

♣

Let's see now what happens if the loading process is quasi-static. If the loading process is quasi-static the final quantity work done by the forces is the sum of all the elementary quantities of work done by force F, which is not constant during the loading process, but varies linearly with x, according to Hooke's law. Let's suppose that F is proportional to x by the unknown constant "a":

$$F=ax.$$

The total work is the integral sum of all the elementary quantities of work Fdx:

$$W= \int F(x)dx= \int axdx= ax^2/2.$$

Setting W=Z (which we can now do because there is no kinetic energy, since W was evaluated in the hypothesis of a quasi-static loading process), we obtain

$$ax^2/2=kx^2/2$$

and thus

$$a=k$$
$$F=kx$$
$$x=F/k.$$

This is what we had found previously using the real loading process.

♣

Summarising, we have found a particularly elegant way of imposing equilibrium: writing the expression of total potential energy and specifying that this total potential energy does not change when the configuration is arbitrarily varied in ways that are infinitely small and compatible with the constraints. There exists a unique configuration for the given problem that satisfies this condition, and that is the desired equilibrium configuration.

In the general case, the elastic strain energy is obtained by summing the strain energies of all the infinitesimal volumes dV of which it is composed. At a generic point P(x, y, z) where there are present stresses $\boldsymbol{\sigma}$ and strains $\boldsymbol{\varepsilon}$, the strain energy dZ is

$$dZ=0.5\boldsymbol{\sigma}^T\boldsymbol{\varepsilon}dV$$

(see Appendix I for the notation and Appendix II for a review of the theory of elasticity).

The total strain energy is thus

$$Z=\int dZ=0.5\int\boldsymbol{\sigma}^T\boldsymbol{\varepsilon}dV.$$

3.3 Technical beam theory

3.3.1 Introduction, review of properties of plane areas

Let's assume a rectilinear right-angle prism obtained by rigidly translating a generic section Q in the direction normal to the plane of the section itself. Given an axis r', and s the distance of a generic point on the section from axis r', the *first moment of area* of section Q with respect to axis r' is defined by the quantity

$$S_{r'} = \int sdA,$$

which is essentially the sum of all the infinitesimal areas of the section times the distances of these areas from the axis in question.

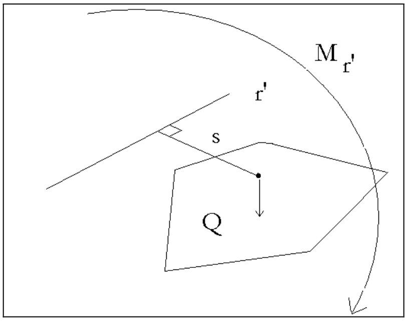


Figure 3-3: physical interpretation of the first moment of area

The first moment of area can be easily interpreted geometrically. If a heavy plate of a shape identical to that of section Q is placed on a horizontal plane, it can be seen that moment $M_{r'}$ of the weight forces of the plate with respect to axis *r'* is given by

$$M_{r'} = t\ \rho\ S_{r'}$$

where t is the thickness of the plate and ρ the weight density of the material used. Among all the possible parallel axes there exists one particular axis *r* with respect to which the static moment of section Q is null.

If a different direction is chosen (for example, the direction perpendicular to the previous one) it will be found that there exists one and only one axis "s" for which the static moment is null. The common point G of axis r and axis s is called the *centroid* of the section (when suspended by a thread that passes through this point the plate will not rotate).

When the section is placed with reference to a pair of orthogonal centroidal Cartesian axes (r, s), the following magnitudes, called *moments of inertia*, can be defined:

$$I_r \equiv \int s^2 dA$$

$$I_s \equiv \int r^2 dA$$

$$I_{rs} \equiv \int rs dA\,.$$

The first two of these are the moments of inertia of the section with respect to axis r and axis s respectively. The third is called the *mixed second moment of area* of the section with respect to axes (r, s). *Generally, the mixed second moment of area of the section is non-null.* However, if axes (r, s), are rotated by a suitable angle of rotation α, it is always possible to find a pair of centroidal Cartesian axes (y, z) for which the mixed second moment of area is null. These axes are called *principal* axes and are characterised by the property of rendering respectively minimum and maximum the moment of inertia of the section with respect to any centroidal axis: graphically representing the variation of the moments of inertia with the variation of the placement of the axes with respect to which they are calculated gives rise to an ellipse, the axes of which correspond to the principal axes of the section.

It can be easily proven that, given I_r, I_s, I_{rs}, angle α by which the pair of axes (r, s) must be rotated in order to find the pair of axes (y, z) is given by:

$$\alpha = \frac{1}{2} arctg(\frac{2I_{rs}}{I_s - I_r})\,.$$

If there is an axis of symmetry in a section, this is also a principal axis, and the other principal axis will be simply an axis perpendicular to that and passing through the barycentre of the section. *The formulas of the technical theory of beams are valid using the principal axes system of coordinates*, and are not valid (unless appropriate modifications are made) if the axes of reference are not the principal axes. This is an important point and must be borne in mind: if a section is L-shaped, the principal axes are *not* parallel to the sides.

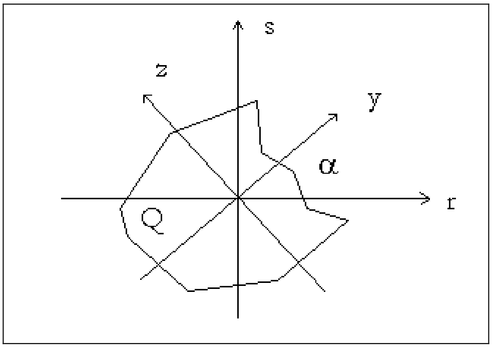


Figure 3-4: Finding the principal axes.

The locus of the points G for all the sections that make up the prism will be called the *axis line* or, more briefly, the *axis*.

Various sub-theories (also called Saint-Venant cases) converge in beam theory, each of which deals with particular conditions of stress that the prism is subject to.

3.3.2 ELONGATION/SHORTENING

One of the possible conditions of stress occurs when the prism is loaded with forces in the same direction as its axis, with resultants lying on the axis itself. In this case the prism is subject to elongation or shortening, and the prism's axis – after deformation – remains rectilinear. A classic example of this is a bar suspended by one end.

Elongation is the simplest case, and we will examine it in detail because it allows us to demonstrate various things that will be useful later. The formal steps of this simple example can be generalised to much more complicated cases, and share a kind of common structure with more complex examples (for which we will omit the steps), and therefore a thorough understanding is worth the effort.

Let's presume that the sections of the prism are translated parallel to its axis. If this is true, then all the points of the section that have abscissa x are subject to the same displacement u in the direction of the axis of the prism, *and thus all the points of a section have the same strain and the same stress.*

If u is equal for all the points of the same section of the prism, then it depends only on the x-coordinate, and not on the y- or z-coordinate. In this case for the displacement function u(x, y, z), we have:

$$u(x, y, z) = u(x).$$

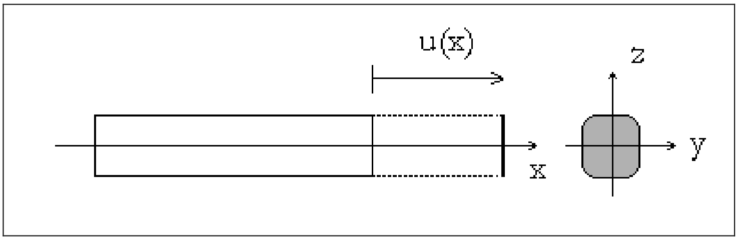


Figure 3-5: Kinematic model of elongation/shortening.

We can therefore write the fundamental equations of the theory of elasticity, modified to deal with the particular set of displacements taken into account (*displacement field*). Let's hypothesise about the the displacement field and see what happens. For the sake of rigour it is necessary to add hypotheses about the displacements v(x, y, z) and w(x, y, z). We can see that to the following hypotheses

$$v(x, y, z) = -\nu y u'$$

$$w(x, y, z) = -\nu z u',$$

where ν is Poisson's ratio, there corresponds the fact that

$$\sigma_y = \sigma_z = 0.$$

In essence, the fibres of the prism are under normal stress only in the direction of the prism's axis, and to this force correspond free transversal contractions (or dilatations) which are responsible for displacements v and w. Since these transversal contractions are, as we said, free, *the stresses in the direction normal to the axis of the prism are identically null.* Analogously, it can be seen that the shear strains are everywhere null, and thus the shear stresses are null.

$$\tau_{xy} = \tau_{yz} = \tau_{zx} = 0$$

Let's continue taking into consideration only the normal stresses in the direction of the prism.

Congruence relation (relationship between strains and displacements):

$$\varepsilon = du/dx = u'. \qquad [3\text{-}4]$$

Constitutive relation (relationship between stresses and strains):

$$\sigma = E\varepsilon. \qquad [3\text{-}5]$$

We will use the term *axial force* to refer to the quantity N(x) defined thus:

$$N(x) \equiv \int \sigma dA = \sigma A. \qquad [3\text{-}6]$$

We shouldn't be put off by this integration. We have written others and will write more, so it is worthwhile to look at it more closely. A normal tension σ is applied to the individual infinitesimal element of area of section dA. The consequent infinitesimal force dF is given by the product of the infinitesimal area dA and the normal tension acting on it: $dF=\sigma dA$. If we now want to know the total sum of all these infinitesimal forces – that is, if we want to know the value of the resultant – we must add all of the infinite terms that correspond to all of the infinite points of the section. We need only place the "S" symbol for sum, stretched into the symbol for *integral sum* "$\int$", in front of the products.

$$\int dF=\int \sigma dA$$

The sum of all these infinite infinitesimal forces is a finite value[2] called "axial force", symbolised by N. The symbol "≡" indicates that quantity $\int \sigma dA$ *defines* the term "axial force".

It should be noted that in the previous formula tension σ can been placed before the integral sign because it depends not on y or z (and thus not on the various points of the section), but only on x (as does the ε from which it derives). In the same way as when we factor out a value common to all terms in a sum,

$$(ac+bc+dc+ec) = c(a+b+d+e),$$

we can place outside the integral a term that does not change in the various points that contribute to the integral sum that the integral represents:

$$\int \sigma dA=\sigma \int dA.$$

The introduction of this new term N, axial force, is useful because it defines an aggregate of forces with a particular significance in terms of statics. This is also the case in the theories of membranes and plates. Instead of keeping the stresses in the point we keep an appropriate sum, either simple or weighted, of the stresses that have a particular significance in the context of the theory. It is like substituting the individual invoices of individual businesses with the total sum that represents gross national product. Here we evaluate the total effects on the section, rather than the effects on individual points. The quantities introduced by this procedure are called *generalised stresses* (and for the corresponding strains, *generalised strains*) indicating that the concepts that are valid for a single point have been generalised for a set of points. For example, the generalised relationship between stresses and strains in this case is:

$$N=EAu' \qquad [3\text{-}7]$$

(note that here the generalised strain coincides with the point strain).

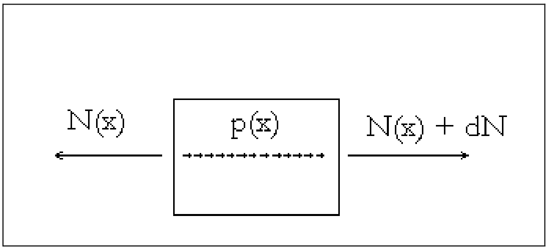
[2] As you might guess, if we add a finite number of infinitesimal terms we obtain another infinitesimal quantity.

If the generic segment of a bar is subjected to a distributed axial load (that is, a distributed load in the direction of the axis of the element) of intensity p(x), the resulting equilibrium is:

$$-N+p(x)dx+(N+dN)=0,$$

which becomes

$$dN/dx=-p(x). \qquad [3\text{-}8]$$



By combining the three equations for equilibrium, constitution, and congruence we arrive at the solution equation in which the displacement u(x) is unknown:

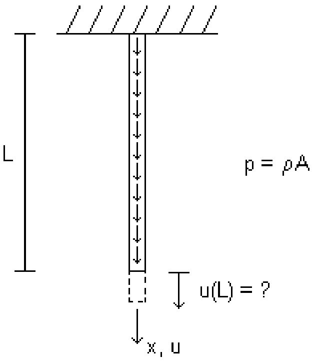
$$EAu''=-p(x). \qquad [3\text{-}9]$$

As you can see, the solution to the problem relates the second derivative of the unknown displacement u to the given distributed load. This kind of equation, in which the unknown is a *function* (rather than one or more numbers) and in which the derivatives of the function appear, is called a *differential equation*. This is a very simple differential equation, inasmuch as the derivatives appear in respect to a single variable rather than to several.

In our equation the unknown is function u(x), that is, the displacement undergone by the various sections of the element. When u(x) is known, by means of the relationships between displacements and strains (congruence), we can evaluate the strain and from that the stress, by means of the relationship between strains and stresses (an example is given below). It is very important to understand this series of logical and mathematical steps because this is the series of steps perfomed by the FEM, at least in its most widely used formulation (the *displacement* formulation, that is, the one in which the displacements are the primary unknowns). It is actually possible to solve the equations keeping the forces as unknowns, but this is not the most widely-used formulation.

Example:

Given a bar subject to its weight per unit of length p and suspended by one end, what is the total elongation? What is the tension at the joint section?



Successively integrating the differential solution equation, we get

$$EAu''=-p$$

$$EAu'= -px+c_1$$

$$EAu = -px^2/2+x\cdot c_1+c_2.$$

Setting x=0, u=0 (the displacement is null because we are at the point of suspension) and x=L, $\varepsilon=0$ (the strain is null because it is the free end), we have

$$c_1=pL$$

$$c_2=0$$

$$EAu = -px^2/2+pLx$$

$$EAu'=EA\varepsilon=-px+pL$$

$$\sigma=p(L-x)/A.$$

Therefore,

$$u(L) = pL^2/(2EA)$$

$$\sigma(0)=pL/A.$$

For the sake of completeness, let's look at the problem from the point of view of energy, applying the principle of stationarity of total potential energy discussed in section 3.2. We will take some rather difficult steps, but bear in mind that these are mathematical steps related to integral calculus. Even if the steps themselves are not completely clear, the general procedure should be.

The strain energy Z is obtained by taking the integral sum of all the contributions $(1/2)\sigma\varepsilon$ multiplied by the infinitesimal volume dV on which they act:

$$Z=\tfrac{1}{2}\int\sigma\varepsilon dV= \tfrac{1}{2}\iint E\varepsilon^2(dAdx)= \tfrac{1}{2}\int EAu'^2dx. \qquad [3\text{-}10]$$

Note that taking the sum of all the volumes is equivalent to taking the sum of the infinitesimal areas at all infinitesimal "heights" dx, and the sum of all the "dx" for which this takes place (two sums, two integral signs). The expression for strain energy is first written in the most generic form, and this used in the particular situation under examination, for which the strain ε is equal to u′. The final expression is the generalised strain energy.

It can be seen in the last step that not all quantities vary from point to point on the section, and thus one of the two sums – that with respect to dA – is equal to A, inasmuch as the other terms can be placed outside the integral sign (factored out).

With only a few substitutions, strain energy Z can also be written equivalently in a generalised sense thus:

$$Z=½\int(N^2/EA)dx. \qquad [3\text{-}11]$$

The infinitesimal quantity of work performed by the axial loads dW is obtained by multiplying the infinitesimal force pdx by the displacement u(x):

$$dW=p(x)u(x)dx.$$

The work of the loads is easily obtained by taking the sum of all the segments along dx:

$$W=\int p(x)u(x)dx.$$

Note that the solution equation can also be found by the stationarity of total potential energy. In fact, it turns out that

$$\Pi = Z - W = ½ \int \sigma\, \varepsilon\, dV - \int p\, u\, dx = ½\iint E\, \varepsilon^2\, (dA\, dx) - \int p\, u\, dx =$$

$$= ½\int E\, A\, u'^2\, dx - \int p\, u\, dx.$$

If u(x) undergoes an infinitesimal variation δu(x), the new configuration is obtained with a total displacement u(x)+δu(x). Making Π stationary, we get

$$\delta\Pi=\Pi(u+\delta u)-\Pi(u)=0.$$

If we perform the previous calculation we get

$$\int E\, A\, u'\, \delta u'\, dx - \int p\, \delta u\, dx = 0.$$

Partially integrating the first integral, and after a few steps, we get

$$[E\, A\, \delta u\, u']^L - [E\, A\, \delta u\, u']_0 - \int (E\, A\, u'' + p)\, \delta u\, dx=0.$$

The value inside the first brackets is null because in x=L, being the free end, u′=0; the value inside the second brackets is null because in order to respect the constraints in x=0, there must always be δu=0. In order for the integral for all possible values of δu to be null, the parentheses *at all points* of the rod must be null. If we imagine that δu is null everywhere except in a generic point Q, one possible choice has to be in point Q

$$EAu''+p=0$$

But Q can be any point whatsoever, and thus in all points the previous equation must be true and, moreover, equal to equation 3-9 found previously by other means.

♣

As you can see, applying the principle of stationarity of total potential energy leads directly to writing the conditions for equilibrium. In this example the mathematical steps may seem very complex. In actual practice, in the application of the FEM things are much simplified because functions u(x) and δu(x) generally become simple polynomials.

We have :

1. written the elastic strain energy Z as a function of the unknown displacement u;
2. written the work W performed by the loads as a function of the unknown displacement u;
3. annulled the first variation of the total potential energy Π=Z-W for all possible variations of the configuration u(x);
4. arrived at the equations (in this case differential, but in future cases algebraic) that are used to find the unknown displacements directly;
5. calculated the strains by means of the displacements (see example);
6. calculated the stresses by means of the strains, thanks to the constitutive relation.

We will take steps that are very similar to these when we solve problems using the FEM. The difference is that we will make further, significant simplifying hypotheses (*the astrological prinicple*).

3.3.3 BENDING AND SHEAR (EULER-BERNOULLI)

Beam theory refers to slender, right-angle, rectilinear prisms (as we shall see, with a length that is at least 10 to 15 times greater than the maximum transversal dimension of the plane figure which, when translated, gives rise to the prism). If the solid being examined for the stress and strain has these characteristics, and if the answer is examined in areas that are sufficiently far from those in which the forces are directly applied,[3] it will be found that the results predicted by the theory closely agree with those observed and verified by experiment. If the length of the prism is increased, the sections remaining the same, the agreement

[3] According to Saint-Venant's principle [1], the effect related to *distribution* of forces applied to the two ends of the prism, given equal resultants, rapidly become negligible as the distance from the extreme area where the forces are applied increases. Further, in zones that are sufficiently far from the base of the prism, the distribution is independent of the particular way in which the resultants are applied.

of theory with experimental data also increases. The theory disregards problems tied to the diffusion of the stresses in areas close to the point where the loads are applied. *The theory only takes into account stresses σ in the direction of the axis of the prism.* Strictly speaking, the theory considers – as we shall see – shear stresses to be null.

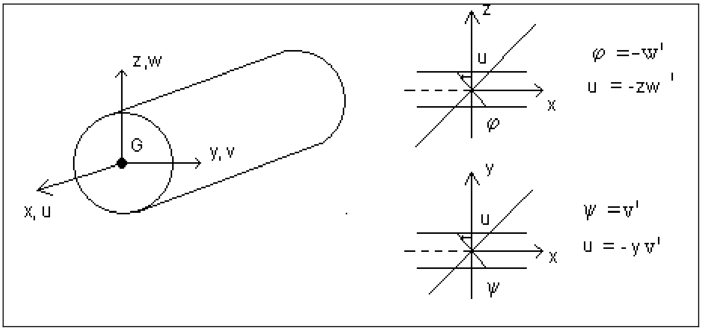


Figure 3-6: Notation and kinematic model for bending along two principle axes.

The whole theory derives from the hypothesis that *the sections undergoing deformation rotate remaining planar and perpendicular to the axis line in their deformed position.* The prism will be referred to a system of centroidal axes (that is, a system of axes whose origin coincides with the barycentre) in which the x-axis coincides with the axis of the prism, and *the other axes of the section, y and z, will be the principal ones.*

Given u(x, y, z), v(x, y, z), and w(x, y, z) as the unknown displacments of a generic point P with coordinates (x, y, z), we make the following hypotheses about the displacements (let's imagine that bending takes place only about the z-axis, and thus that the movements of the points on the axis take place in direction z):

$$u(x, y, z) = -y\psi(x)$$

$$v(x, y, z) = v(x)$$

$$w(x, y, z) = 0.$$

Imagining the displacements to be very small (and thus that angle ψ can be fused with its tangent), and recalling that the sections must remain perpendicular to the axis line in its deformed position, we have

$$\psi \approx (\mathrm{tg}\psi) = v'.$$

Thus:

$$u(x, y, z) = -yv'$$

$$v(x, y, z) = v(x) \qquad [3\text{-}12]$$

$$w(x, y, z) = 0.$$

The first hypothesis, regarding u(x, y, z), follows from that of maintaining the sections planar and perpendicular to the axis line. The third hypothesis, regarding v, means admitting that all the points of the same section are displaced along the y direction in the same way (v is only a function of x) and implies that strain ε_y is null. Strictly speaking, these hypotheses about the displacement field (including the second one) lead to normal stresses σ_z and σ_y that are not equal to zero (more precisely, they would be equal to $\nu\sigma_x /(1-\nu)$), but these stresses are not dealt with. In order for there to be null stresses the displacements w cannot be null, and the displacements v must also depend on y. In fact, the fibres in tension or compression that are left free to shrink transversally (so as not to give rise to stresses σ_z and σ_y) determine a certain quantity of displacement v(y) and w(z). These displacements are not dealt with. *Therefore, it is not possible to use the technical theory of beams to obtain normal stresses that are not in the direction of the axis of the prism.* This should not be forgotten here, or during the analysis of finite elements either.

In other words, we have the following approximations of the normal stresses:

$$\sigma_x = \sigma_x\ (x, y)=\sigma=E\varepsilon=Eu'=-Eyv''$$

$$\sigma_y = 0$$

$$\sigma_z = 0.$$

The shear stresses are subject to significant simplifying hypotheses as well. In fact, in coherence, the kinematic hypotheses made lead to the annulment of the shear and shear stresses

$$\tau_{xy} = 0$$

$$\tau_{yz} = 0$$

$$\tau_{zx} = 0.$$

In dealing with the beam the generalised stresses of *bending moment* M and *shear* T are introduced in the following way (note that we are dealing with a definition):

$$M=M_z(x)\equiv\int\sigma y dA$$

$$T=T_y\ (x)\equiv\int\tau dA.$$

If a uniformly distributed load q(x) in the direction of y acts on the beam, we have the following indefinite equations of equilibrium

$$dT/dx = -q(x)$$

$$dM/dx=T \qquad [3\text{-}13]$$

$$d^2M/dx^2=-q(x).$$

The first is equilibrium in translation, the second equilibrium in rotation. The third is a restatement of the second using the first.

The generalised strain is given by the *curvature of the axis line* C(x), which, if the displacements are small, can be approximated by the second derivation of the curve that represents the axis line in the deformed position, w(x), called the *elastic line.*

$$C=v''.$$

The relationship between stresses and strains is particularised in the following way, with I_z the moment of inertia of the section with respect to the principal axis z

$$M_z=-EI_zC=-EI_zv''. \qquad [3\text{-}14]$$

This result is obtained simply by applying the definition of bending moment, substituting the expression found for the stress σ, and recalling the definition of the moment of inertia about the y-axis. The negative sign "-" is due to the fact that a bending moment assumed to be positive generates a downwards rather than upwards concavity, given the choice made as to the axes' positive direction.

If we were to calculate the value of the moment M_y we would find an identically null value, thanks to the fact that the system of reference used is the principal one (I_{yz} =0). In fact,

$$M_y = \int \sigma z dA=-Ev''\int yzdA=-Ev''I_{yz}=0.$$

The differential equation that must be satisfied is:

$$EIv^{IV} - q(x) = 0, \qquad [3\text{-}15]$$

that is, a differential equation in which the fourth derivative of the elastic line v(x) appears. This equation can be found simply by substituting the expression 3-14 found for bending moment in the third equation 3-13.

Differential equation 3-15, called the *elastic line equation*, tells us that the function that gives the deformation of the axis line of a rectilinear beam has significant regularities, which we can briefly summarise:

- In segments that are without distributed loads, v is in the form of a cubic function;
- In the segments where distributed load is constant, v is in the form of a quartic function;
- In the segments where the distributed load is linear, v is in the form of a quintic function.

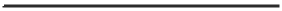
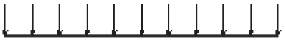
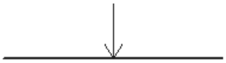
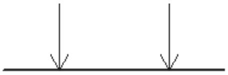
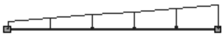
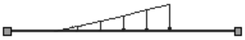
	v is a cubic function ($v=ax^3+bx^2+cx+d$)
	v is a quartic function ($v=ax^4+bx^3+cx^2+dx+e$)
	v is given by two cubic functions
	v is given by three cubic functions
	v is a quintic function ($v=ax^5+bx^4+cx^3+dx^2+ex+f$)
	v is given by a cubic function, a quintic function, and another cubic function

Table 3-1: Examples of beams and their relative elastic lines.

The strain energy of a segment of a beam is written

$$Z =(1/2) \int MCdx=(1/2) \int EI\, v''^2\, dx.$$

In fact, if we appy the usual rule that gives the strain energy and we write

$$Z= (1/2) \int \sigma\varepsilon dV=(1/2)E\int \varepsilon^2 dV,$$

and substitute the expression found for ε, we obtain

$$dZ= (1/2)E\int \varepsilon^2 dV=(1/2)E\int (v'')^2\, dx\, (\int y^2 dA)=(1/2)EI_z\int v''^2 dx.$$

It should be noted that the strain energy related to shear does not appear. The work performed by the distributed load is

$$dW=q(x)\, v(x)\, dx,$$

while the total potential energy for a finite segment of a beam, summing the various infinitesimal contributions, becomes

$$\Pi=(1/2) \int EI_z\, v''^2\, dx - \int q\, v\, dx. \qquad [3\text{-}16]$$

In what we have just discussed it should be noted that the shear has appeared only in the equilibrium equation. In effect, the intial hypotheses about the displacement field would lead to considering the shear stress on the section to be null, and thus shear to be null as well. To be precise, the hypothesis that the sections remain *perpendicular to the axis line* means that the value of strain γ, and thus of stress τ, are identically null.[4] Actually, the equilibrium equation $M'=T$ tells us that when the moment varies along the element there necessarily has to be shear, and consequently shear stresses. The error is one of the consequences of the arbitrary hypotheses made about the displacement field. In the technical theory of beams the shear is evaluated by means of the equilibrium conditions, when the distribution of the bending moment is known.

3.3.4 BENDING AND SHEAR (TIMOSHENKO)

To get around the problems inherent in the Euler-Bernoulli beam theory, while at the same time maintaining a theory that is easy to use, some corrections have been made to the intial hypotheses. The hypothesis that the sections rotate while remaining planar is kept, but the hypothesis that the section remains perpendicular to the axis line in the deformed position is discarded. This leads us to write the following displacement field:

$$\begin{aligned} u(x, y, z) &= -y\psi \\ v(x, y, z) &= v(x) \qquad [3\text{-}17] \\ \psi &= v'-\beta(x) \end{aligned}$$

where $\beta(x)$ is an additional angle that moves the section from the normal to the axis line. This angle is made only depending on x, and thus is supposed to be constant for all the points of the section (an obvious consequence of the hypothesis that the sections remain planar).

The expressions for the two strains ε and γ are found using the definitions:

$$\varepsilon=u_{,x}=-y(v''-\beta')$$

$$\gamma=u_{,y}+v_{,x}= u_{,y}+v'=\beta(x).$$

Note that the stress τ is considered to be constant on the section (inasmuch as γ does not depend on either y or z).

If we apply the definition of T and the relationship between stress and strain, we would get

$$T(x)=GA\beta(x)$$

but, in order to take into account the fact that the distribution of shear stresses is actually not constant, we add the shear factor χ, which makes it possible to evaluate a mean value in the sense of energy (that is, the value that gives rise to a strain energy equal to that which we would have taking into account the distribution of shear stresses). Essentially, if Z is the strain energy associated with the "exact" shear, which takes into account the distribution of the stresses, and

[4] It is sufficient to write $\gamma_{xy}=u_{,y}+v_{,x}= -\psi+ v_{,x}$ and to remember that if the sections are perpendicular to the axis line $\psi=+v_{,x}$: from $\gamma=0$ it turns out that $\tau=0$.

Z* that predicted by these formulas, we set the corrective factor χ, called the shear factor, such that

$$Z=\chi Z^*.$$

As a consequence, after a few steps (not shown here), we obtain:

$$T(x)=GA\beta(x)/\chi$$

or briefly, with the introduction of the *shear area* $A^*=A/\chi$

$$T(x)=GA^*\beta(x). \qquad [3\text{-}18]$$

This is the generalised relationship between stresses and strains that is added to that of bending.

There are tables available that give the values for the shear factor in function of the various section shapes (see, for instance [Zienkiewicz and Taylor 1989]). Table 3-2 summarises the values of the shear factor for some section typologies.

The relationship between bending stress and bending strain has to be modified thus:

$$M(x)= -EI\psi'=EI(\beta'-v'') \qquad [3\text{-}19]$$

(we need only apply the definition of bending moment).

In essence, there appears a new unknown, the function $\beta(x)$. Substituting the expressions for the generalised stresses found in the indefinite equilibrium conditions, we find the equations that solve the problem:

$$\begin{gathered} GA^*\beta'=-q(x) \\ EI(\beta^{III}-v^{IV})=-q(x). \end{gathered} \qquad [3\text{-}20]$$

Section	Shear factor χ
Rectangular	1.2
Solid circular	1.11
Hollow circular, thin section	2
I. Shear parallel to the web	A/Ht_w
I. Shear parallel to the flanges	$(3/5)(A/Bt_f)$

A, area of the section;
H total height;
B, flange width;
$t_w$web thickness;
t_f, flange thickness.

Table 3-2: Shear factor values.

The problem is obviously a complicated one. We can simplify it by hypothesising that there are no distributed loads q(x). In this case the first of the two previous equations tells us that β must be constant as x is varied:

$$\beta(x)=\text{constant}.$$

The second equation then becomes the known equation of the elastic line when there are no distributed loads:

$$v^{IV}=0,$$

which has a cubic equation as its solution. It would appear that nothing has changed with respect to the Euler-Bernoulli beam, but in reality when we set conditions on the rotations of the faces, *we must set them on* ψ*, not on v′ as before.* This leads to differences in the calculation of the forces that are precisely the approximate solution we are looking for.

Let's consider, for example, a simple cantilever beam with a concentrated load at the end (pointing upwards). The solution for v is a cubic equation, thus

$$v=ax^3+bx^2+cx+d$$

$$v'=3ax^2+2bx+c.$$

We set:

$$v(0) = 0 \qquad \rightarrow \qquad d=0$$

$$\psi\,(0)=0 \qquad \rightarrow \qquad c=\beta.$$

Now recall that

$$M=Px - PL = -EIv'' = -EI(6ax+2b)$$

that is,

$$PL-Px=EIv''=EI(6ax+2b).$$

We find, making similar monomials equal,

$$b=PL/2EI$$

$$a=-P/6EI.$$

Since T=P, we have further

$$T=P=GA^*\beta$$

$$\beta=P/GA^*.$$

The coefficients a, b, c, and d being by now known, we evaluate the displacement at the end f=v(L)

$$f = -PL^3/(6EI) + PL^3/(2EI) + PL/(GA^*) = [PL^3/(3EI)] + PL/(GA^*) =$$

$$= [PL^3/(3EI)]\ [1 + 3EI/(GA^*\ L^2)].$$

The term $[1+3EI/(GA^*\ L^2)]$ represents the factor by which we multiply the displacement that we have found with the theory that excludes shear, in order to arrive at the predicted displacement of the more complete theory. The term

$$3EI/[GA^* L^2]$$

is thus the correction. For the sake of brevity we will call it c. If we set $I=kAh^2$, where h is the total height of the section, A its area, and k a parametre equal to, for example, 1/12 for the rectangular section and 1/16 for the circular section, the factor in question becomes

$$c=3E\chi kh^2/(GL^2).$$

Since it turns out that $E/G= 2(1+\nu)$, where ν is Poisson's ratio, we again have

$$c=6\ (1+\nu)\ \chi\ k\ (h/L)^2.$$

Now let's look at the value of c according to different types of section and different relationships L/h, presuming that steel is used[5] ($\nu=0.3$).

From table 3-3 it is possible to extrapolate the engineering values for the percentage differences between the values of the simplified theory and those of the theory that includes shear. These appear in the table 3-4.

If we had performed exact calculations for beams supported at both ends with a distributed load we would have found differences that were 2-3 times greater.

As you can see, it is clear that if the beam is sufficiently shallow the error that results when strain energy is neglected is acceptable. In the case of hyperstatic beams, errors of a similar order of magnitude are found in the components of stress, with values varying in function of the constraints, of the point along the beam, and of the individual components of stress. *In this case we need only point out that use of the Euler-Bernoulli beam theory used for prisms that are not sufficiently shallow will lead to errors that can be quite significant.* Further, when L/h tends to approach 4, 3 or less, then Timoshenko's theory cannot be applied either, inasmuch as the kinematic hypothesis upon which it is based is no longer valid.

Section	L/h=5	L/h=10	L/h=20	L/h=30
Rectangular	0.03120	0.00780	0.00195	0.00087
Solid circular	0.02167	0.00542	0.00135	0.00060
Hollow circular(D/t=30)	0.078	0.01950	0.00487	0.00217
IPE300	0.13616	0.03404	0.00851	0.00378
HEA300	0.25680	0.06420	0.01605	0.00713
HEB300	0.26426	0.06606	0.01652	0.00734
HEM300	0.22391	0.05597	0.01399	0.00622

Table 3-3 Percentage differences between the displacement values predicted by simplified theory and those predicted by the theory including shear

[5] Given that ν cannot exceed 0.5, values generally close to these are obtained with other materials as well.

Section		L/h=5	L/h=10	L/h=20	L/h=30
Rectangular	▮	3.1%	0.8%	0.2%	<0.1%
Solid circular	●	2.2%	0.5%	0.1%	<0.1%
Hollow circular (D/t=30)	○	7.8%	1.9%	0.5%	0.2%
IPE300	I	13.6%	3.4%	0.9%	0.4%
HEA300	I	25.7%	6.4%	1.6%	0.7%
HEB300	I	26.4%	6.6%	1.7%	0.7%
HEM300	I	22.4%	5.6%	1.4%	0.6%

Table 3-4: Differences in bending between shallow beam and deep beams, end-loaded cantilever.

3.3.5 Torsion

For torsion as well it is possible to arrive at simplified results by beginning with a set of reasonable hypotheses about the way that points on the section move.

Let’s imagine that the sections rotate by an angle that varies linearly with the distance from the end of the prism (and thus linearly with x, that is, that “angle”= θ^*x with θ^* a constant), and that the points of the section move in the direction of the axis of the prism independent of the point on the section, but not independent of x (free and constant warping, uniform torsion, fig. 3-7).

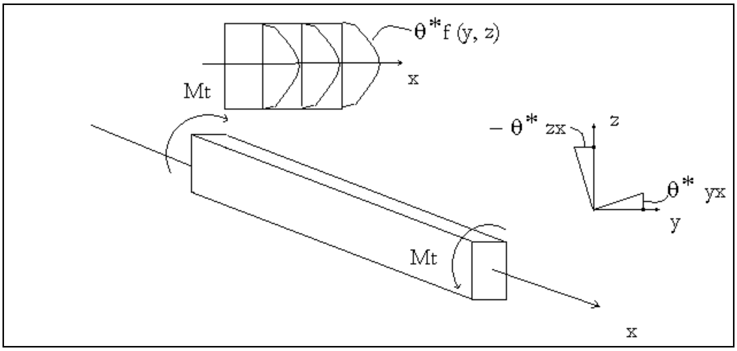


Figure 3-7: Notation and kinematic model for torsion.

These hypotheses can be written

$$u(x, y, z) = u(y, z) = \theta^{*} f(y, z)$$

$$v(x, y, z)= v(x, z) = -\theta^{*}xz \qquad [3\text{-}21]$$

$$w(x, y, z)=w(x, y) = \theta^{*}xy.$$

From these hypotheses, applying the congruency relations, we deduce the strains:

$$\varepsilon_x=\varepsilon_y=\varepsilon_z=0$$

$$\gamma_{xy}=\theta^{*}\ (f_{,y}-z)$$

$$\gamma_{xz}=\theta^{*}\ (f_{,z}+y)$$

$$\gamma_{yz}=0.$$

Thus, applying the constitutive relations, we deduce the stresses

$$\sigma_x=\sigma_y=\sigma_z=0$$

$$\tau_{xy}=G\theta^{*}\ (f_{,y}-z)$$

$$\tau_{xz}=G\theta^{*}\ (f_{,z}+y)$$

$$\tau_{yz}=0.$$

Recalling the indefinite equilibrium equation, and admitting null volume force, there must be (cfr. Appendix II)

$$\tau_{xy,y}+\tau_{xz,z}=0$$

and thus

$$f_{,yy}+f_{,zz}=0.$$

Further, for global equilibrium, the sum of all the contributions that the shear stresses make to the torsional moment has to be equal to the applied torsional moment, and thus the elementary forces $\tau_{xz}dA$ and $\tau_{yz}dA$ multiplied by the respective lever arms "y" and "z" and summed at all the points must give the applied torsional moment

$$M_t=\int(-z\tau_{xy}+y\tau_{xz})dA.$$

Substituting the expressions of τ_{xy} and τ_{xz} in the previous equation we find

$$M_t=G\theta^{*}\ (I_p+\int(-zf_{,y}+yf_{,z})dA).$$

Having *defined* the *moment of polar inertia* thus:

$$I_p\equiv\int(y^2+z^2)dA=I_z+I_y,$$

if we now *define* the *moment of torsional inertia* I_t as the quantity

$$I_t \equiv I_p+\int(-zf_{,y}+yf_{,z})dA$$

we can write

$$M_t=G\ I_t\ \theta^{*}, \quad [3\text{-}22]$$

which is the relationship that relates generalised stresses and strains. The proportionality constant θ^* is called the angle of twist per unit length and can be measured as a derivative of the angle of rotation of the section about the axis of the prism, with respect to the x coordinate:

$$\theta^*=d\theta/dx=\theta'.$$

From the condition for indefinite equilibrium of the beam segment subject to distributed torsional moment per unit length m_x

$$dM_t+m_x dx=0$$

it can be seen that when there is no distributed torsional moment we must have

$$M_t'=0$$

and thus for equation 3-22

$$\theta''=0,$$

which implies a linear variation for θ, while $\theta'=\theta^*$ is constant.

It should be noted that the moment of torsional inertia is a quantity proper to the geometry of the section, inasmuch as it can be calculated beginning with the moment of polar inertia and from the function f(y,z) that has to satisfy the conditions on the domain. In the case of a solid circular section, the moment of torsional inertia coincides with the moment of polar inertia. This is not the case with other types of section.

Table 3-5 gives the values for I_t for some of the principal sectional shapes currently used.

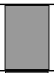
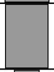
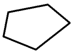
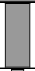
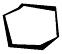
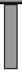
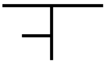
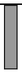
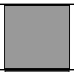
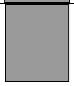
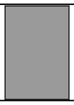
This can lead us to the known stress values as well as to the expression of the warping function f(y,z). Various theories that further simplify the problem have been developed according to the shape of the section, arriving at the formulas that are widely used in engineering practice. For hollow, closed profiles with thin walls the well-known formula of Bredt is used

$$\tau=\frac{M_t}{2\Omega s}$$

where s is the thickness of the profile (and can also vary) and Ω the area enclosed by the median line of the profile: the shear stress is understood as being tangential to the median line. For profiles created by one or more elongated rectangles, the shear stress is obtained from

$$\tau=\frac{2M_t}{I_t}z$$

with the presumption that the elongated rectangle in question is parallel to the y-axis, and that z varies between –0.5s and +0.5s. The values contained in the table are valid for the value of the moment of torsional inertia.

Section		I_t	Section		I_t
Solid circular		$\frac{\pi R^4}{2}$	Rectangular section b/a=2.0 [Timoshenko and Goodier 1970]		0.229 a^3b
Solid elliptical		$\pi\frac{a^3b^3}{a^2+b^2}$	Rectangular section b/a=2.5 [Timoshenko and Goodier 1970]		0.249 a^3b
Hollow section with thin walls of constant thickness		$\frac{4s\Omega^2}{\Gamma}$	Rectangular section b/a=3.0 [Timoshenko and Goodier 1970]		0.263 a^3b
Hollow section with thin walls of varying thickness		$\frac{4\Omega^2}{\oint\frac{d\Gamma}{s}}$	Rectangular section b/a=4.0 [Timoshenko and Goodier 1970]		0.281 a^3b
Section composed of elongated rectangles		$\frac{1}{3}\sum_i l_i s_i^3$	Rectangular section b/a=5.0 [Timoshenko and Goodier 1970]		0.291 a^3b
Square section [Timoshenko and Goodier 1970]		0.1406 L^4	Rectangular section b/a=10. [Timoshenko and Goodier 1970]		0.312 a^3b
Rectangular section b/a=1.2 [Timoshenko and Goodier 1970]		0.166 a^3b	Rectangular section b/a=∞ [Timoshenko and Goodier 1970]		0.333 a^3b
Rectangular section b/a=1.5 [Timoshenko and Goodier 1970]		0.196 a^3b			

Ω	area enclosed by the median line
Γ	perimeter, borderline
s	thickness
l	length of an elongated rectangle
a,b	side of a rectangle (short side), semiaxis of an ellipse

Table 3-5: Values for the moment of torsional inertia.

3.3.6 FINAL CONSIDERATIONS

3.3.6.1 THE EULER-BERNOULLI “BEAM”.

The behaviour of a “beam” is usually described as a combination of the axial, bending and torsional behaviour that are mutually decoupled (admitting axial forces to be applied at the centroid: where this is not the case it is necessary to add offset bending moments that are calculated by multiplying the axial forces by the eccentricity with respect to the principal axes).

For the sake of brevity and simplicity, we will adopt matrix notation, constructed in such a way as to respect the formulas and the results that we have already introduced. Here we will also generalise the results found for bending about the principal Z-axis, including bending about the principal Y-axis (a beam in space).

In the case of the Euler-Bernoulli beam (that is, without the contribution of shear strain energy), the vectors of the unknown displacement functions $\mathbf{U}$, strains $\boldsymbol{\varepsilon}$ and generalised stresses $\boldsymbol{\sigma}$ can be defined in the following way:

$$\mathbf{U} = \{u, v, w, \theta\}^T$$

$$\boldsymbol{\varepsilon} \equiv \{u', \theta', -w'', -v''\}^T$$

$$\boldsymbol{\sigma} \equiv \{N, M_t, M_y, M_z\}^T.$$

Thus we have the following fundamental relationships:

$$\boldsymbol{\varepsilon}=\mathbf{CU}$$

$$\boldsymbol{\sigma}=\mathbf{D}\boldsymbol{\varepsilon}$$

where we have set

$$\mathbf{C}=\left|\begin{array}{c:c:c:c} \frac{\partial}{\partial x} & 0 & 0 & 0 \\ \hdashline 0 & 0 & 0 & \frac{\partial}{\partial x} \\ \hdashline 0 & 0 & -\frac{\partial^2}{\partial x^2} & 0 \\ \hdashline 0 & -\frac{\partial^2}{\partial x^2} & 0 & 0 \end{array}\right|$$

$$\mathbf{D} = \left|\begin{array}{c:c:c:c} EA & 0 & 0 & 0 \\ \hdashline 0 & GI_t & 0 & 0 \\ \hdashline 0 & 0 & EI_y & 0 \\ \hdashline 0 & 0 & 0 & EI_z \end{array}\right|.$$

The strain energy of a beam segment is written

$$Z=0.5\int\boldsymbol{\sigma}^T\boldsymbol{\varepsilon}dx=0.5\int\boldsymbol{\varepsilon}^T \mathbf{D}\ \boldsymbol{\varepsilon}dx$$

(note that **D** is symmetric), and thus

$$Z=0.5\int\mathbf{U}^T\mathbf{C}^T\mathbf{DCU}dx.$$

This last formula is fundamental, because it expresses the strain energy directly as a function of the unknown displacement functions, contained in the vector **U** **.**

The work performed by the distributed loads can be expressed in a similar way with the introduction of vector **p**:

$$\mathbf{p}=\{p_x, p_y, p_z, 0\}^T$$

where the contribution of distributed torsion is presumed to be null. Thus

$$W=\int\mathbf{p}^T\mathbf{U}dx.$$

Writing in matrix form prepares us for the discussion in Chapter 4, when we will go into the FEM in detail. The organisation in rows and columns is largely arbitrary, in the sense that given equal values of Z (which is a *number*, not a matrix) the matrices can be defined in many different ways by changing the order in which the unknown displacement functions or strains appear.

3.3.6.2 THE TIMOSHENKO "BEAM"

It is possible to write the equations for the Timoshenko beam in matrix form as well, bringing together the various contributions. However, in this case *it is necessary to add the contribution of shear strain energy and to model explicitly the two new unknown displacement functions*, one for bending about the Y-axis, the other for bending about the Z-axis. In both cases the unknown displacement functions are rotations of the faces, that is, rotation φ for bending about the Y-axis, and rotation ψ for bending about the Z-axis.

These two relationships are valid

$$\varphi = w' - \alpha$$

$$\psi = v' - \beta$$

where α and β are the corrections, that is, the additional angles related to shear strain.

The vectors of unknown displacement, strains and generalised stresses are

$$\mathbf{U} \equiv \{u, v, w, \theta, \varphi, \psi\}^T$$

$$\boldsymbol{\varepsilon} \equiv \{u', \alpha, \beta, \theta', -\varphi', -\psi'\}^T = \{u', v'-\psi, w'-\varphi, \theta', -\varphi', -\psi'\}^T$$

$$\boldsymbol{\sigma} \equiv \{N, T_y, T_z, M_t, M_y, M_z\}^T.$$

The fundamental relationships we saw earlier

$$\boldsymbol{\varepsilon} = \mathbf{C}\mathbf{U}$$

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon}$$

still continue to be valid, where we set

$$\mathbf{C} = \left|\begin{array}{c:c:c:c:c:c} \frac{\partial}{\partial x} & 0 & 0 & 0 & 0 & 0 \\ \hdashline 0 & \frac{\partial}{\partial x} & 0 & 0 & 0 & -1 \\ \hdashline 0 & 0 & \frac{\partial}{\partial x} & 0 & -1 & 0 \\ \hdashline 0 & 0 & 0 & \frac{\partial}{\partial x} & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & -\frac{\partial}{\partial x} & 0 \\ \hdashline 0 & 0 & 0 & 0 & 0 & -\frac{\partial}{\partial x} \end{array}\right|$$

$$\mathbf{D} = \left|\begin{array}{c;{2pt/2pt}c;{2pt/2pt}c;{2pt/2pt}c;{2pt/2pt}c;{2pt/2pt}c} EA & 0 & 0 & 0 & 0 & 0 \\ \hdashline 0 & GA_y & 0 & 0 & 0 & 0 \\ \hdashline 0 & 0 & GA_z & 0 & 0 & 0 \\ \hdashline 0 & 0 & 0 & GI_t & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & EI_y & 0 \\ \hdashline 0 & 0 & 0 & 0 & 0 & EI_z \end{array}\right|$$

where A_y is the shear area when the shear is acting in direction y and A_z the shear area when the shear is acting in direction z. For strain energy Z on a segment of the beam, the following equation is still formally valid

$$Z=0.5\int \boldsymbol{\sigma}^T \boldsymbol{\varepsilon} dx=0.5\int \boldsymbol{\varepsilon}^T \mathbf{D}\ \boldsymbol{\varepsilon} dx$$

and thus also

$$Z=0.5\int \mathbf{U}^T \mathbf{C}^T \mathbf{D} \mathbf{C} \mathbf{U} dx.$$

3.4 PLANE STRESS

Let's consider a thin planar lamina subject to a set of forces acting in the same plane as the lamina and uniformly distributed along the thickness. If the thickness "t" of the lamina is very small, we can consider the normal tension acting perpendicular to the plane of the lamina to be null, as are the shear stresses directed normal to the lamina. In fact, there are no applied normal or tangential forces on the two surfaces of the lamina, because there are no forces that have components that are perpendicular to the lamina. Since they are null on both faces, given the modest thickness, it is hypothesised that they are everywhere null.

Let's presume that the plane of the lamina is plane x, y. Thus we can set

$$\sigma_z=\tau_{yz}=\tau_{zx}=0. \qquad [3\text{-}23]$$

In a problem of plane stresses the only stresses that are not null are parallel to the median plane of the lamina; there are no stresses normal or shear stresses perpendicular to the plane itself.

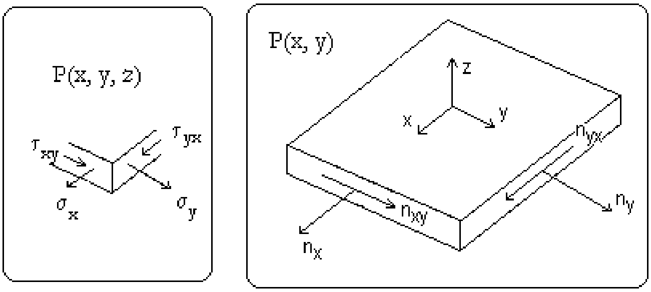


Figure 3-8: Plane stresses, notation.

Therefore, for the vector of unknown strains we can set

$$\boldsymbol{\varepsilon}=\{\varepsilon_x, \varepsilon_y, \gamma_{xy}\}^T$$

and for the stresses the vector

$$\boldsymbol{\sigma}=\{\sigma_x, \sigma_y, \tau_{xy}\}^T.$$

Neither the stress vector nor the strain vector depends on z; both depend only on x and y.

The strain ε_z is obtained by means of the most general constitutive relationship (cfr. Appendix II), while the two strains γ_{yz} and γ_{zx} are null

$$\varepsilon_z = -\nu(\sigma_x + \sigma_y)/E.$$

The vector of unknown displacements is

$$\mathbf{U}= \{u, v\}^T,$$

which means that there are only two unknown translations.

The matrix **D** that gives the relationship between strains and stresses (cfr. Appendix II) becomes:

$$\boldsymbol{\sigma}=\mathbf{D}\boldsymbol{\varepsilon}$$

$$\mathbf{D}=\frac{E}{1-\nu^2}\begin{vmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{vmatrix}$$

where ν is Poisson's ration, and E is Young's modulus. The indefinite equations of equilibrium become

$$\sigma_{x,x} + \tau_{xy,y} + b_x = 0$$

$$\tau_{xy,x} + \sigma_{y,y} + b_y = 0$$

(understanding that $\tau_{xy} = \tau_{yx}$), while the relationships between displacements and strains are reduced to

$$\boldsymbol{\varepsilon}=\mathbf{CU}$$

having set

$$\mathbf{C}\equiv\begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix}.$$

The problem is generally solved making use of Airy stress functions. In fact, given a certain unknown function ϕ(x, y), if we place (presuming volume forces b_x and b_y to be constant)

$$\sigma_x = \frac{\partial^2 \phi}{\partial y^2} - x b_x$$

$$\sigma_y = \frac{\partial^2 \phi}{\partial x^2} - y b_y$$

$$\tau_{xy} = -\frac{\partial^2 \phi}{\partial x \partial y}$$

it can be seen that the conditions of indefinite equilibrium are automatically satisfied. In order for the strains associated with the stresses to satisfy the compatibility conditions,[6] there must also be

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) \left(\sigma_x + \sigma_y \right) = 0,$$

which corresponds to the following condition

$$\frac{\partial^4 \phi}{\partial x^4} + 2 \frac{\partial^4 \phi}{\partial x^2 \partial y^2} + \frac{\partial^4 \phi}{\partial y^4} = 0.$$

Therefore, if we are able to find the function that satisfies the above differential equation on the whole domain, respecting as well the boundary conditions, the problem is solved.

As seen in the problems of plane stress, we are looking for displacements, stresses and strains that are functions of only two coordinates, x and y. *The possible displacements in direction z imposed on the lamina by external factors do not give rise to any stresses in the lamina*, which *does not have any bending stiffness.* The stresses are constant along the thickness (the z coordinate), and only change when the point on the plane (x, y) is changed. Often these laminae are called *membranes*, to distinguish them from *plates*, which on the contrary do have bending stiffness.

It is possible to define the *generalised stresses* by multiplying the tensions by the thickness of the lamina. In this case the stresses *take the physical dimensions of a force per unit of length* and can be regarded as forces transmitted by a segment of lamina of unit length. The stiffness matrix of the material is also multiplied by thickness t, giving rise to a matrix called the *extensional stiffness matrix.*

To be precise, when we set:

$$n_x \equiv \int_{-t/2}^{t/2} \sigma_x dz = t \sigma_x$$

[6] These compatibility conditions are obtained by taking the differential of the stress-strain relationships and making the consequent mixed derivatives equal.

$$n_y \equiv \int_{-t/2}^{t/2} \sigma_y dz = t\sigma_y$$

$$n_{xy} \equiv \int_{-t/2}^{t/2} \tau_{xy} dz = t\tau_{xy}$$

the vector of generalised stresses becomes

$$\boldsymbol{\sigma}=\{n_x, n_y, n_{xy}\}^T$$

and the extensional stiffness matrix becomes

$$\mathbf{D}\equiv\frac{tE}{1-\nu^2}\begin{vmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{vmatrix}$$

(for simplicity we will leave the symbols the same). For the *generalised* stresses as well the formal relationship is

$$\boldsymbol{\sigma}=\mathbf{D}\boldsymbol{\varepsilon}$$

and the successive relationship which gives the elastic strain energy, and which is formally identical to that found for beams (except for the fact that here the integration is on an area rather than a line) is

$$Z=0.5\int \mathbf{U}^T\mathbf{C}^T\mathbf{D}\mathbf{C}\mathbf{U}dA.$$

To summarise: *plane stress* is for thin laminae loaded parallel to the median plane. There is no stress component perpendicular to the median plane. There are only two unknown displacement functions, generalised stresses, called membrane stresses, having the dimensions of a force per unit of length. The domain of study is an area.

3.5 Plane strain

Consider a structure obtained by translating continuously a given plane figure normal to its plane, subject to equal forces on each of the infinite plane sections of which it is composed. Further, these forces do not have any longitudinal components.

In these conditions we can make the following simplifying hypotheses about the strain regime to which the body is subject, presuming that the axis of extrusion is the z-axis.

$$\varepsilon_z=\gamma_{yz}=\gamma_{zx}=0. \qquad [3\text{-}24]$$

The strain vector is then

$$\boldsymbol{\varepsilon}=\{\varepsilon_x, \varepsilon_y, \gamma_{xy}\}^T$$

and for the stress vector we have

$$\boldsymbol{\sigma}=\{\sigma_x, \sigma_y, \tau_{xy}\}^T.$$

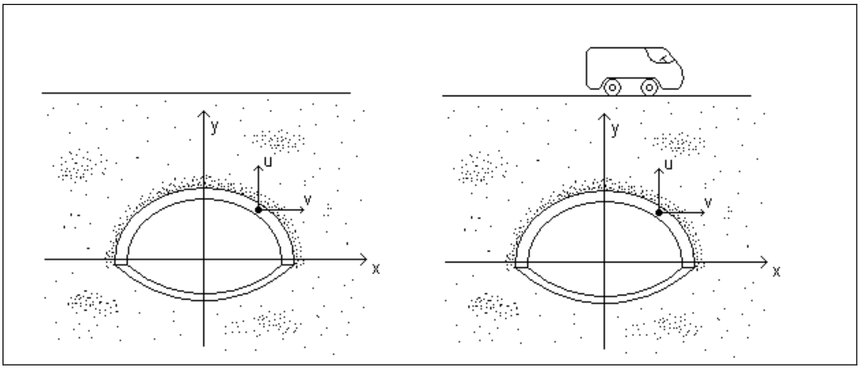


Figure 3-9: (left) A typical problem of plane strain; (right) a problem that cannot be studied in plane strain because the loads are not equal at every section.

The stress σ_z is obtained by means of the most general constitutive relationship (cfr. Appendix II), while the two stresses τ_{yz} and τ_{zx} are null

$$\sigma_z = \nu(\sigma_x + \sigma_y).$$

The formulation of plane strain problems closely resembles that already seen for plane stress problems, with the only difference found in the stiffness matrix of the material, which becomes

$$\mathbf{D}=\frac{E}{(1+\nu)(1-2\nu)}\begin{vmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & \dfrac{1-2\nu}{2} \end{vmatrix}$$

Like plane stress problems, in the case of plane strain problems the only two unknown displacement functions are translation u and translation v. The plane strain formulation is quite useful in the treatment of tunnels, conduits and dams, where the notable development in one direction can be made by comparing it to a development in infinite length. Naturally, the results found can be applied to areas that are sufficiently far from the ends (of the tunnel, gallery or dam).

In plane strain problems the domain is a plane. In theory, the thickness is infinite. Even though in reality it is finite, it does not enter into the formulation, inasmuch as all of the magnitudes are calculated by unit of thickness.

The strain energy per unit of thickness Z* can be written using the usual rules, inasmuch as a null strain ε_z is associated with the non-null component of stress σ_z: it therefore does not contribute to strain energy Z*, which can be written

$$Z^*=0.5\int \mathbf{U}^T\mathbf{C}^T\mathbf{D}\mathbf{C}\mathbf{U}dA,$$

that is, in a way that is formally identical to what we have already seen for plane stress.

3.6 Thin plate (Germain-Kirchhoff)

The problems of thin plates and thick plates have given rise to numerous studies, both because they are intrinsically rather complex, and also because we encounter these problems very frequently in actual practice. Volume 2 of Corradi dell'Acqua's excellent book [1992] contains a panoramic, modern treatment of the subject, with a formulation that is very close to matrix calculation and therefore to the FEM. The classic text of Timoshenko-Krieger [1959] offers a broad overview, including the various special solutions that have been found.

Here we will limit ourselves to a summary and discussion of the hypotheses and the fundamental results.

Let's consider a solid that is a plane figure with a given thickness t, subject to forces perpendicular to its median plane. A solid with these characteristics is called a *plate*, and it has various characteristics that serve to simplify the general problem of the theory of elasticity. Essentially, the median plane deforms, generating a bending in the plate; this bending will give rise to stresses in bending as well as in shear. The domain to be studied, originally a solid, is reduced to a plane. Of the three unknown displacement functions we started with, there remains only one.

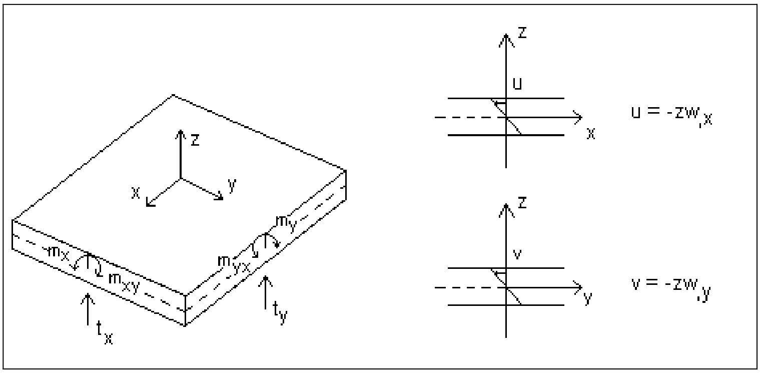


Figure 3-10: Plate stresses, kinematic hypothesis for a thin plate.

For the sake of simplicity, let's suppose that the median plane of the solid is plane xy. If we presume that the segments perpendicular to the median plane before deformation rotate while still remaining perpendicular to the median plane after deformation, we can set

$$u=-zw_{,x}$$

$$v=-zw_{,y} \qquad [3\text{-}25]$$

$$w=w(x, y)$$

where w is the translation of the points of the median plane in a direction normal to the median plane itself.

This hypothesis about the displacements defines the Germain-Lagrange-Kirchhoff thin plate,[7] in which, as we will see, the strain energy due to the effects of shear is not taken into consideration.

We calculate the strains by applying the definitions (cfr. Appendix II):

$$\varepsilon_x=-zw_{,xx}$$

$$\varepsilon_y=-zw_{,yy}$$

$$\varepsilon_z=0$$

$$[3\text{-}26]$$

$$\gamma_{xy}=-2zw_{,xy}$$

$$\gamma_{yz}=0$$

$$\gamma_{zx}=0.$$

The last two equations tell us that the shear stresses related to the forces perpendicular to the median plane are null, and thus the formulation of the Kirchhoff plate *excludes strain energy associated with shear*, a circumstance which is acceptable only if the plate is sufficiently thin.

The following generalised stresses are *defined*:

$$m_x \equiv \int_{-t/2}^{t/2} \sigma_x z dz$$

$$m_y \equiv \int_{-t/2}^{t/2} \sigma_y z dz$$

$$m_{xy} \equiv \int_{-t/2}^{t/2} \tau_{xy} z dz \qquad [3\text{-}27]$$

[7] Marie-Sophie Germain (1776-1831) presented a mémoire to the Paris Académe des Sciences during the years in which Napoleon was wreaking havoc in Europe. In 1811 Joseph-Louis Lagrange (1736-1813) reexamined the mémoire and introduced some modifications. Later, Gustav Robert Kirchhoff (1824-1887) broadened and clarified the significance of the various hypotheses.

$$t_x \equiv \int_{-t/2}^{t/2} \tau_{zx} z dz$$

$$t_y \equiv \int_{-t/2}^{t/2} \tau_{yz} z dz .$$

These stresses are moments and shears *per unit of length* (thus the use of lower-case letters), and bending stresses, that is, they are related to the curvature of the median plane of the plate. We repeat, to obtain the moments and shears it is necessary to multiply these generalised forces by a width (usually of one unit). The membrane stresses, as can be easily seen, are everywhere null, inasmuch as they depend on integrals of the type

$$\text{membrane stress} = \text{constant} \int_{-t/2}^{t/2} z dz = 0 .$$

In a flat plate subject to loads normal to the median plane there is no membrane behaviour.

As far as shears t_x and t_y are concerned, from the equations 3-26 it follows that the values of shear behaviour are everywhere null, inasmuch as the shear strains associated with them are null because of the initial hypotheses made about the displacement field. What we obtain recalls what we had already obtained for Euler-Bernoulli beams.

Given that the shears have been estimated to be everywhere null, we can state the *generalised stress vector* as follows:

$$\boldsymbol{\sigma} \equiv \begin{vmatrix} m_x \\ m_y \\ m_{xy} \end{vmatrix} .$$

If we apply the relationships that allow us to go from point strains to point stresses (constitutive relations), we can express all the point stresses as functions of the coordinates and the displacement w derived in an appropriate way. To be precise, we find

$$\sigma_x = -\frac{E}{1-\nu^2}(zw_{,xx} + z\nu\, w_{,yy})$$

$$\sigma_y = -\frac{E}{1-\nu^2}(zw_{,yy} + z\nu\, w_{,xx})$$

$$\tau_{xy} = -2Gzw_{,xy} .$$

By substituting the expressions thus found for the point stresses in the equations that define the generalised stresses, we find the following final relation:

$$\begin{vmatrix} m_x \\ m_y \\ m_{xy} \end{vmatrix} = D \begin{vmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \dfrac{1-\nu}{2} \end{vmatrix} \begin{vmatrix} -w_{,xx} \\ -w_{,yy} \\ -2w_{,xy} \end{vmatrix} \qquad [3\text{-}28]$$

where “D” is the bending stiffness of the plate (note the analogy with the EI of the rectangular beam, EI=Ebh^3/12):

$$\mathrm{D} \equiv \frac{Et^3}{12(1-\nu^2)}.$$

If we *define* the vector

$$\boldsymbol{\varepsilon} \equiv \begin{vmatrix} -w_{,xx} \\ -w_{,yy} \\ -2w_{,xy} \end{vmatrix} \qquad [3\text{-}29]$$

as the vector of generalised strains, these definitions relate the *generalised* stresses to the *generalised* strains, that is, as usual

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon}$$

where

$$\mathbf{D} \equiv \mathrm{D} \begin{vmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \dfrac{1-\nu}{2} \end{vmatrix}. \qquad [3\text{-}30]$$

The only unknown displacement function is the function w(x, y). We can write the relation between the displacement and the strain in a generalised sense in the usual way

$$\boldsymbol{\varepsilon} = \mathbf{C}\mathbf{U}$$

as long as we set

$$\mathbf{U} \equiv \mathrm{w}$$

$$\mathbf{C}\equiv\begin{vmatrix} -\dfrac{\partial^2}{\partial x^2} \\ -\dfrac{\partial^2}{\partial y^2} \\ -2\dfrac{\partial^2}{\partial x\partial y} \end{vmatrix}.$$

The generalised stresses have to satisfy the appropriate indefinite equilibrium conditions (a small segment of a generic plate has to be in equilibrium), a relationship that is written in the following way, imagining that there exists a load distributed per surface unit “q” acting in the direction normal to the median plane

$$-m_{xy,x}-m_{y,y}+t_y=0$$

$$t_{x,x}+t_{y,y}+q=0$$

$$m_{xy,y}+m_{x,x}-t_x=0.$$

If we substitute the relations between generalised stresses and generalised strains in the indefinite equilibrium equations, we arrive at the following differential equation that formally solves the problem (the Germain-Lagrange equation):

$$\frac{\partial^4 w}{\partial x^4}+2\frac{\partial^4 w}{\partial x^2\partial y^2}+\frac{\partial^4 w}{\partial y^4}=\frac{q}{D}.$$

The strain energy on a given part of the domain is written thus

$$Z=0.5\int\boldsymbol{\sigma}^{\mathbf{T}}\boldsymbol{\varepsilon}dA=0.5\int\boldsymbol{\varepsilon}^{\mathbf{T}}\mathbf{D}\boldsymbol{\varepsilon}dA=0.5\int\mathbf{U}^{\mathbf{T}}\mathbf{C}^{\mathbf{T}}\mathbf{DCU}dA, \qquad [3\text{-}31]$$

that is,

$$Z=0.5\int|\,w\,|\begin{vmatrix} -\dfrac{\partial^2}{\partial x^2} & -\dfrac{\partial^2}{\partial y^2} & -2\dfrac{\partial^2}{\partial x\partial y}\end{vmatrix}\begin{vmatrix} D & D\nu & 0 \\ D\nu & D & 0 \\ 0 & 0 & \dfrac{D(1-\nu)}{2}\end{vmatrix}\begin{vmatrix} -\dfrac{\partial^2}{\partial x^2} \\ -\dfrac{\partial^2}{\partial y^2} \\ -2\dfrac{\partial^2}{\partial x\partial y}\end{vmatrix}|\,w\,|dA\cdot \qquad [3\text{-}31a]$$

It should be noted that the integral has to be taken on a surface, not a volume.

To summarise, in a thin (or Germain-Kirchhoff) plate the membrane stresses are null, as are the normal and shear stresses acting perpendicular to the median plane of the plate. In contrast, the moments of the plate, *which have the physical dimensions of moments per unit of length*, are non-null. From the moments of the plate it is possible to calculate the stresses by applying the following formulas:

$$\sigma_x = 12\frac{m_x z}{t^3}$$

$$\sigma_y = 12\frac{m_y z}{t^3}$$

$$\tau_{xy} = 12\frac{m_{xy} z}{t^3}.$$

On the median plane the point stresses are null (z=0), while on the planes at the ends they are maximal (z=± t/2). There is only one unknown displacement function, but it is a function of two variables, and no longer of only one as in beam theory.

3.7 Thick plate (Mindlin-Reissner)

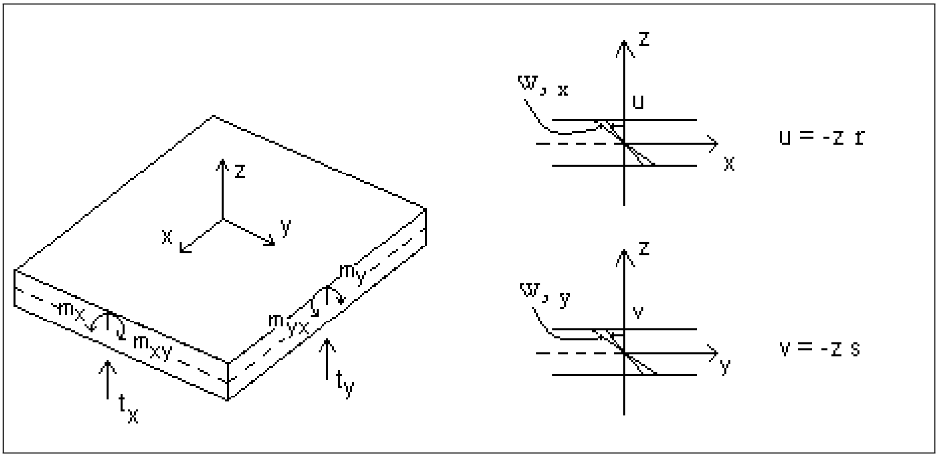


Figure 3-11: Notations and cinematic hypotheses for the Mindlin-Reissner plate.

To overcome the difficulties related to the use of a formulation in which the cross-sectional strain energy is neglected, a theory was proposed that does away with the hypotheses that require that segments perpendicular to the median plane *before* deformation remain perpendicular to the median plane *after* deformation as well. *The hypothesis that such segments remain rectilinear when rotating* remains (fig. 3-11). It is quite clear that we are dealing with a generalisation of plates similar to what we have already seen for beams. The theory is tied to the names of Raymond D. Mindlin (1906-1987) and Eric Reissner (1913-1996).

The hypotheses about the displacement field become

$$u=-zr\ (x,\ y)$$

$$v=-zs\ (x,\ y) \qquad [3\text{-}32]$$

$$w=w(x,\ y),$$

that is, the rotation of the segment is no longer identified with one of the partial derivatives of the unknown function w. There are thus three unknown fuctions – w(x, y), r(x, y), and s(x, y) – and "r" and "s" represent the rotations of the segment about the y-axis and x- axis.

The point strains are written

$$\begin{aligned} \varepsilon_x &= -zr_{,x} \\ \varepsilon_y &= -zs_{,y} \\ \varepsilon_z &= 0 \qquad [3\text{-}33] \\ \gamma_{xy} &= -z(r_{,y}+s_{,x}) \\ \gamma_{yz} &= w_{,y}-s \\ \gamma_{zx} &= w_{,x}-r. \end{aligned}$$

For the vector of generalised stresses we have to maintain the vector complete with unitary shears, and thus

$$\boldsymbol{\sigma} \equiv \left| \begin{matrix} m_x \\ m_y \\ m_{xy} \\ t_x \\ t_y \end{matrix} \right|$$

applying the theoretical point-level stress-strain relation and using the definitions of generalised stresses, we can find the generalised stress-displacement relations

$$\left| \begin{matrix} m_x \\ m_y \\ m_{xy} \\ t_x \\ t_y \end{matrix} \right| = \left| \begin{array}{ccc:cc} D & D\nu & 0 & 0 & 0 \\ D\nu & D & 0 & 0 & 0 \\ 0 & 0 & D\frac{1-\nu}{2} & 0 & 0 \\ \hdashline 0 & 0 & 0 & \frac{Gt}{1.2} & 0 \\ 0 & 0 & 0 & 0 & \frac{Gt}{1.2} \end{array} \right| \left| \begin{matrix} -r_{,x} \\ -s_{,y} \\ -r,_y - s,_x \\ w_{,x}-r \\ w_{,y}-s \end{matrix} \right|. \qquad [3\text{-}34]$$

If we now define the vector of unknown displacements as

$$\mathbf{U} \equiv \{w,\ r,\ s\}^T$$

and define the generalised strains in the following way

$$\boldsymbol{\varepsilon} \equiv \begin{vmatrix} -r_{,x} \\ -s_{,y} \\ -r_{,y}-s_{,x} \\ w_{,x}-r \\ w_{,y}-s \end{vmatrix} \qquad [3\text{-}35]$$

we can write the usual congruency relation **ε**=**CU** by using the matrix **C** defined as

$$\mathbf{C} \equiv \begin{vmatrix} 0 & -\dfrac{\partial}{\partial x} & 0 \\ 0 & 0 & -\dfrac{\partial}{\partial y} \\ 0 & -\dfrac{\partial}{\partial y} & -\dfrac{\partial}{\partial x} \\ \dfrac{\partial}{\partial x} & -1 & 0 \\ \dfrac{\partial}{\partial y} & 0 & -1 \end{vmatrix}.$$

Thus the relations between strains and generalised stresses becomes the usual

$$\boldsymbol{\sigma}=\mathbf{D}\boldsymbol{\varepsilon}=\mathbf{DCU},$$

as long as we set

$$\mathbf{D}=\left|\begin{array}{ccc:cc} D & D\nu & 0 & 0 & 0 \\ D\nu & D & 0 & 0 & 0 \\ 0 & 0 & D\dfrac{1-\nu}{2} & 0 & 0 \\ \hdashline 0 & 0 & 0 & \dfrac{Gt}{1.2} & 0 \\ 0 & 0 & 0 & 0 & \dfrac{Gt}{1.2} \end{array}\right|$$

(note that the shear factor for 1.2 is applied in a way that is analogous to what we saw earlier for rectangular sections in the beam problems).

Elastic strain energy on a thick beam of domain A is written (dA=dxdy)

$$Z=0.5\int\boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\varepsilon}\,\mathrm{dA}=0.5\int\boldsymbol{\varepsilon}^{\mathrm{T}}\mathbf{D}\boldsymbol{\varepsilon}\,\mathrm{dA}.$$

This expression is obtained by applying the definition that is valid for continua, that is, by writing the integral on the volume as a function of the point stresses and strains, and then recalling the definitions introduced for generalised stresses and strains.

By substituting in place of the vector of generalised strains its expression as a function of unknown displacements, we obtain the by now familiar formula

$$Z=0.5\int \mathbf{U}^{\mathrm{T}}\mathbf{C}^{\mathrm{T}}\mathbf{D}\mathbf{C}\mathbf{U}\,\mathrm{d}A. \qquad [3\text{-}36]$$

That is, explicitly,

$$Z=0.5\int\left(\begin{vmatrix} w & r & s \end{vmatrix}\begin{vmatrix} 0 & 0 & 0 & \frac{\partial}{\partial x} & \frac{\partial}{\partial y} \\ -\frac{\partial}{\partial x} & 0 & -\frac{\partial}{\partial y} & -1 & 0 \\ 0 & \frac{\partial}{\partial y} & -\frac{\partial}{\partial x} & 0 & -1 \end{vmatrix}\begin{vmatrix} D & D\nu & 0 & 0 & 0 \\ D\nu & D & 0 & 0 & 0 \\ 0 & 0 & D\frac{1-\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & \frac{Gt}{1.2} & 0 \\ 0 & 0 & 0 & 0 & \frac{Gt}{1.2} \end{vmatrix}\begin{vmatrix} 0 & -\frac{\partial}{\partial x} & 0 \\ 0 & 0 & -\frac{\partial}{\partial y} \\ 0 & -\frac{\partial}{\partial y} & -\frac{\partial}{\partial x} \\ \frac{\partial}{\partial x} & -1 & 0 \\ \frac{\partial}{\partial y} & 0 & -1 \end{vmatrix}\begin{vmatrix} w \\ r \\ s \end{vmatrix}\right)dA.$$

[3-36a]

The above expression simply tells us in matrix form what the strain energy Z is equal to.

As for thin plates, it is possible in the case of thick plates to arrive at the differential equations that solve the problem (cfr. for example Timoshenko-Krieger [1959]). Here we omit the description of these differential equations since their solution is not necessary if the properties of the FEM are used (we need only know equation 3-36).

Summarising, the formulation for thick plates often includes the addition of strain energy contributed by shear, using as an unknown function *three functions* of x and y (w, r, s) and excludes the membrane stresses. To the functions for calculating the stresses already examined for thin plates are added the following two formulas

$$\tau_{xz}=\frac{6t_x}{t^3}\left(\frac{t^2}{4}-z^2\right)$$

$$\tau_{yz}=\frac{6t_y}{t^3}\left(\frac{t^2}{4}-z^2\right),$$

which make it possible to calculate the shear stresses normal to the median plane if the generalised shear stresses are known.

The shear stresses are distributed parabolically with a maximum value on the median plane and null values at the end planes. The maximum value of shear stress is

$$\tau_{xz}=1.5t_x/t,$$

obtained by dividing the generalised shear by the thickness and multiplying by 1.5.

3.8 AXIAL SYMMETRY

Consider a solid generated by a 360° rotation about the axis of a plane figure of some generic form (a solid of revolution). If this solid is subjected to identical forces in each of its infinite sections obtained by a plane passing through the axis of revolution, and none of these forces has a component of circumference, but there do exist components that are radial or parallel to the axis of rotation (axially symmetric forces), then it is possible to write the equations of elasticity theory in a particularly useful way.

First, the solid is placed in reference to a cylindrical coordinate system, (r, θ, z), fig. 3-12. In this reference system we will have at each point P the three displacements (u, v, w), which are respectively in the radial, circumferential and vertical directions.

Given the axial symmetry of the problem, we can set

$$\tau_{r\theta}=\tau_{\theta z}=0.$$

The components of strain in the cylindrical reference system are defined as

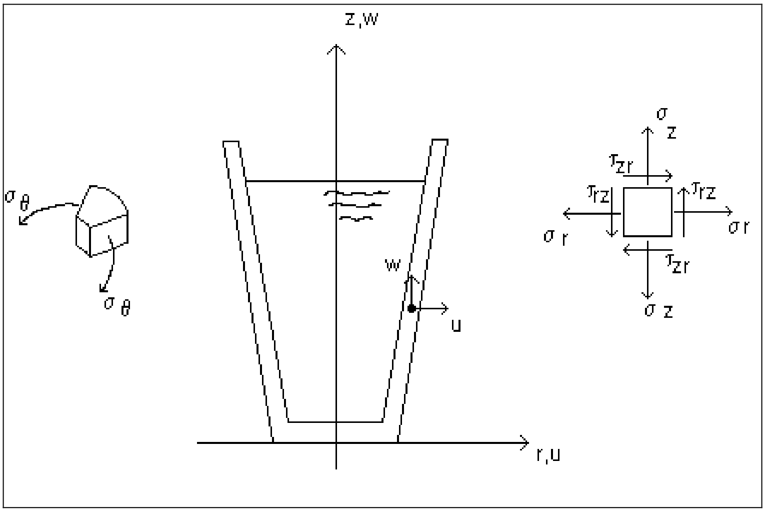


Figure 3-12: Notation for axially symmetrical structures.

$$\varepsilon_r = u,_r$$

$$\varepsilon_\theta = u/r$$

[3-37]

$$\varepsilon_z = w,_z$$

$$\gamma_{rz} = u,_z + w,_r$$

The conditions for axial symmetry are such that

$$v=0,$$

that is, that there is no displacement of the points in the circumferential direction. Thus result the following two vectors of strains and stresses:

$$\boldsymbol{\varepsilon} = \{\varepsilon_r, \varepsilon_\theta, \varepsilon_z, \gamma_{rz}\}^T$$

$$\boldsymbol{\sigma} = \{\sigma_r, \sigma_\theta, \sigma_z, \tau_{rz}\}^T,$$

and each involves functions that depend solely on (r, z).

The only components of unknown displacement are the translation u(r, z) and the translation w(r, z), neither of which depend on θ. We can create the vector of unknown displacements $\mathbf{U} = \{u, w\}^T$ and introduce the congruence matrix $\mathbf{C}$, defined so that

$$\boldsymbol{\varepsilon} = \mathbf{C}\mathbf{U}$$

$$\mathbf{C} = \begin{vmatrix} \frac{\partial}{\partial r} & 0 \\ \frac{1}{r} & 0 \\ 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial z} & \frac{\partial}{\partial r} \end{vmatrix}.$$

This matrix simply repeats what has already been explained about the formulas that relate the displacements to the strains (in the cylindrical reference system). The indefinite equilibrium equations can be written:

$$\frac{\partial \sigma_r}{\partial r} + \frac{\partial \tau_{rz}}{\partial z} + \frac{\sigma_r - \sigma_\theta}{r} = 0$$

[3-38]

$$\frac{\partial \tau_{rz}}{\partial r}+\frac{\partial \sigma_z}{\partial z}+\frac{\tau_{rz}}{r}=0\,.$$

The relationship between stresses and strains can be written in the usual way:

$$\boldsymbol{\sigma}=\mathbf{D}\boldsymbol{\varepsilon}$$

with matrix **D** defined as

$$\mathbf{D}=\frac{(1-\nu)E}{(1+\nu)(1-2\nu)}\begin{vmatrix} 1 & \frac{\nu}{1-\nu} & \frac{\nu}{1-\nu} & 0 \\ \frac{\nu}{1-\nu} & 1 & \frac{\nu}{1-\nu} & 0 \\ \frac{\nu}{1-\nu} & \frac{\nu}{1-\nu} & 1 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2(1-\nu)} \end{vmatrix}$$

(note that the matrix is symmetrical).

The strain energy Z accumulated in the solid in question can be written as

$$Z=0.5\int \boldsymbol{\sigma}^{\mathbf{T}}\boldsymbol{\varepsilon}\ dV,$$

and bearing in mind that

$$dV=\ r\ d\theta\ dA=\ r\ d\theta\ dr\ dz$$

$$Z\ =0.5\int \boldsymbol{\sigma}^{\mathbf{T}}\boldsymbol{\varepsilon}\ r\ d\theta\ dr\ dz.$$

That same strain energy can be written as a function of the displacements using the formulas that relate the displacements to the strains, and the strains to the stresses, and observing that nothing depends on the variable θ, which must be integrated between 0 e 2 π:

$$Z=\pi\int \mathbf{U}^{\mathrm{T}}\ \mathbf{C}^{\mathrm{T}}\ \mathbf{D}\ \mathbf{C}\ \mathbf{U}\ r\ dr\ dz. \qquad [3\text{-}39]$$

As in the cases of plane stresses and plane strains, it is possible to introduce a suitable stress function $\phi(r,\ z)$, called *Love's function*,[8] that makes it possible to express the unknown stresses

$$\sigma_r=\frac{\partial}{\partial z}\left(\nu\nabla^2\phi-\frac{\partial^2\phi}{\partial r^2}\right)$$

[8] A. E. H. Love (1863-1940).

$$\sigma_\theta = \frac{\partial}{\partial z}\left(\nu\nabla^2\phi - \frac{1}{r}\frac{\partial\phi}{\partial r}\right)$$

$$\sigma_z = \frac{\partial}{\partial z}\left((2-\nu)\nabla^2\phi - \frac{\partial^2\phi}{\partial z^2}\right)$$

$$\tau_{rz} = \frac{\partial}{\partial r}\left((1-\nu)\nabla^2\phi - \frac{\partial^2\phi}{\partial z^2}\right).$$

Love's stress function has to satisfy the conditions

$$\nabla^2\nabla^2\phi = 0,$$

where the operator ∇^2 corresponds to the Laplace operator in Cartesian coordinates, and is written

$$\nabla^2 = \frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} + \frac{\partial^2}{\partial z^2},$$

bearing in mind that our functions do not depend on θ.

The complicated equations that we have given must not cause us to lose track of the significant results that we have obtained:

1. In an axially symmetrical solid loaded axially symmetrically only two translations are held as unknowns, u and w, neither of which depend on the θ coordinate;

2. Only four of the six components of stress and strain are unknown, and these depend solely on (r, z);

3. The domain of the problem is no longer a volume, but a plane.

3.9 GENERIC SOLIDS

For generic solids the only thing to do is to use the general equations of elasticity theory, which lead to holding as unknowns the three translations u(P), v(P), w(P) at every point of the structure.

Given that the solution involves six different components of stress and strain at every single point of the solid, it is easy to understand that solid modelling is generally the analyst's last resort, to be used when none of the simplified theories is capable of dealing with all of the most important aspects of the problem at hand.

For a review of the main results in elasticity theory in the most general case of solids, see Appendix II.

3.10 SUMMARY

In examining the various special formulations of elasticity theory we have seen that it is possible to follow a scheme which is common to all the formulations and which we will briefly summarise here.

First, the vector of unknown displacements $\mathbf{U}$ is introduced. This vector can contain one or more unknown functions of one or more variables. Next, the generalised strains $\boldsymbol{\varepsilon}$ and the generalised stresses $\boldsymbol{\sigma}$ are defined. The generalised strains are related to the displacements of the congruence relations, generally differential (matrix $\mathbf{C}$):

$$\boldsymbol{\varepsilon}=\mathbf{C}\mathbf{U}.$$

The generalised stresses are related to the generalised strains by the constitutive relation (the generalisation of Hooke's law):

$$\boldsymbol{\sigma}=\mathbf{D}\boldsymbol{\varepsilon}.$$

Finally, by expressing the strain energy Z as a function of the stresses and strains we arrive at the general expression

$$Z=0.5\int\mathbf{U}^{T}\mathbf{C}^{T}\mathbf{D}\mathbf{C}\mathbf{U}\,d\Xi,$$

in which $d\Xi$ is an infinitesimal portion of the domain typical of the problem at hand (a line, plane, etc.).

As you can see, the strain energy is made to depend solely on the unknown displacements $\mathbf{U}$. This expression, when suitably used in relation to the principle of the stationarity of total potential energy, leads directly to the differential solution equations. If displacements $\mathbf{U}$ are expressed approximately as the weighted sum of known functions (and unknown weights), the differential solution equations become algebraic.

This will be seen in chapter 4.

4 THE FINITE ELEMENT

4.1 INTRODUCTION

We have reviewed the various formulations of elasticity theory that are used in different cases as a framework within which to search for the solutions to a given structural problem. As we have said, the choice of formulation is a necessary first step in performing any kind of modelling. The modelling of the different parts of a structure are made by applying, for the various parts, different simplified theories to which correspond different ideal geometric point loci.

In fact, the choice of a given formulation automatically results in the choice of the type of domain, since, although it is true that nature always furnishes us with three-dimensional objects, this is not the case with theory, which makes ample use of schematisations and simplifications. The domains that we deal with are essentially the following:

Problem	**Domain**	**Displacement functions**
Euler-Bernoulli "beam" (in space)	Straight line	u, v, w, θ
Timoshenko "beam" (in space)	Straight line	u, v, w, θ, φ, ψ
Rod	Straight line	u
Thin plate	Plane surface	w
Thick plate	Plane surface	w, r, s
Axial symmetry	Plane surface	u, v
Plane stresses	Plane surface	u, v
Plane strains	Plane surface	u, v
Solid	Volume	u, v, w

Table 4-1: Problems, corresponding domains and unknown displacement functions.

In a qualitative sense we can say that the simpler the domain is to study (a plane is simpler than a volume, a line simpler than a plane), the more complex the unknown displacement field, since this has to satisfy an increasing number of regularity conditions. For example, the curve that gives the strain of a beam cannot contain cusps, and thus its first derivative must be continuous, while the function that gives the displacement field of a solid can contain discontinuities in

Structural analysis with finite elements
978-0-7277-4093-9

the derivative while still respecting the congruity of the displacements, that is, the fundamental characteristic that there be no gaps or overlaps.

Having established what kind of domain is suitable for the part of structure in question, we can begin to investigate the system used by the FEM to find an approximate solution to the problem. It bears repeating that the solution is approximate not only because it solves the system of solution equations approximately, but also because the solution equations themselves introduce significant approximations with respect to the actual structure. If the hypotheses that lead to the choice of a given formulation are erroneous, or not sufficiently verified, then even though the solution is in keeping with the given problem, the solution is not "exact".

Although there are cases in which the structure in question can be modelled entirely using only one of the formulations of elasticity theory discussed earlier, it often happens that in a given structure there are some parts that can be modelled with one theory, and others that can be modelled through the use of a different theory. For example, if we want to study a building with a steel frame whose service core is in reinforced concrete, we have to model all the beams with one-dimensional elements by using beam theory, where the core in reinforced concrete has to be modelled with spatial membranes or plate-shell elements (depending on whether membrane behaviour is considered sufficient or if it also requires the bending component). In another example, if we want to study the deck of a ship, and we want to limit the degrees of freedom, we cannot model the transversal and longitudinal beams with plate-shell elements: it is necessary to model the deck with plate-shell elements, and the beams and girders with appropriately calibrated one-dimensional elements.

In principle, all structures can be modelled as a solid, but this approach, unitary and logical though it may be, would be extremely unwieldy in many cases, and essentially impossible in some.

For the sake of simplicity, in this chapter we will presume that the structure can be modelled by means of a single formulation, and thus a single domain. Later this hypothesis will be omitted so that we can study the most general cases.

4.2 Inside the finite element

4.2.1 The problem

Let us consider a domain (line, plane or solid) divided into many elementary parts, each of which has a given position and a given shape. Let us now consider a generic part of those into which we have divided the domain, and ask ourselves these questions: How is it possible to describe what takes place within the "piece" of that domain in terms of stresses and strains in a simple way? How can we ensure that each piece respects what happens in adjacent pieces in order that there is no tearing or interpenetration? Is it possible to make it so that the displacements of the boundaries of the piece in question respect the displacements of the boundaries of adjacent pieces, the condition that in fact prevents tearing and laceration? Can we make it so that in addition to displacement the stresses and strains as well maintain the necessary requisites of continuity between pieces, while still remaining an approximate solution? What kind of analogy does there have to be between all the pieces that we have divided the domain into? Under what conditions can we expect that, while still described in an approximate way, the simplified problem comes close to the original problem?

The finite elements method[1] provides clear answers to all these questions.

We will begin with the observation that if the function that describes the displacements were known, then all of the stresses and strains would also be calculated; in fact, we can calculate the strains from the displacements (thanks to congruity), and we can calculate the stresses from the strains (thanks to the constitutive relation, which is essentially a generalisation of Hooke's law).

The fact is that the function that gives the displacements is enormously complex and therefore totally unknown.

Let us see how, by successive steps, we can arrive at the answers to these questions.

4.2.2 ELEMENT TYPE, ELEMENT SHAPE

A first way to simplify the problem lies in simplifying the domain where the solution is sought. Given that it is up to us to establish the "pieces", that is, the *finite elements*, it is certainly simpler to have to deal with elements whose shapes are regular and simplified rather than irregular and complex.

Intuition tells us that we can expect that as the *shape* changes so does the *type* of finite element: in fact, we have to look for an approximate solution within the simplified, differently shaped domains. On the other hand, there is no reason why there can't be different formulations and different kinds of simplifications for a given domain, even assuming elements of an identical shape. While it is possible to find cases where elements with the *same shapes* have *different formulations*, there are no cases where elements have *different shapes* but *identical formulations*, that is to say, identical sets of governing equations, unless we assume that one or more points *collapse* onto others (which can generally be excluded since they lead to very particular consequences). As we shall see, the *shape* that the finite element has to have is strictly related to its formulation, that is, to the system of hypotheses that identify the element itself.

[1] O. C. Zienkiewicz [2] retraces the earliest use of the term "finite elements" to a 1960 work by Clough: "The finite element in plane stress analysis", *Proc. 2nd ASCE conf. On Electronic Computation,* Pittsburgh, PA, Sept 1960.

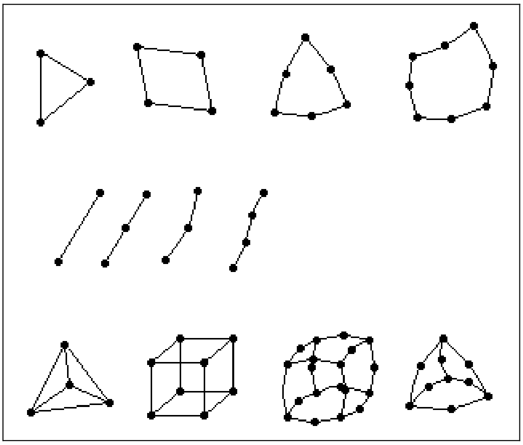


Figure 4-1: finite element shapes, examples

Now, let's suppose that we have chosen a given shape (for example, triangular for a plane domain, or triangular with curved sides, etc.). Given equal shapes, the elements can have different positions (that is, they may lie in different areas of the structure), different dimensions, and even different appearances: they can be more or less regular, or more or less distorted. If the elements were generated haphazardly and there were no precise rules for generating them, they would be quite difficult to describe. But because the shapes of the elements obey very precise, pre-established rules, it turns out that *the geometry of an element can be reconstructed as long as a certain number of its points are known*; these points are called *nodes.*

For example, if the domain being studied is a line element, it can be straight (two nodes), parabolic (three nodes) or cubic (four nodes). In the case of curved beams the use of curved elements necessarily leads to a better performance, for obvious reasons of geometry. If the domain is a surface, the element can be a triangle (with three nodes required for its identification), a quadrilateral (with four nodes required), or by assuming that the sides can be, for example, parabolic, it can be a triangle with curved sides (with six nodes required) or a quadrilateral with curved sides (with eight nodes required).

Once the shape that the element has to have has been established, it is necessary that the element itself pass through a certain number N nodes of the domain of study, such that when the coordinates of these N nodes are known, the *aspect* and the *position* of the finite elements can be completely determined. Choosing the *type* of finite element also immediately determines the number of nodes required to identify it. In contrast, the position of the nodes on the domain, is largely a matter of choice, in the sense that with the exception of a few significant points on the structure which must correspond to nodes, the majority of the points that identify the element, and thus the choice of its exact aspect, is up to the engineer.

What criteria are used to determine the geometry of an element? We will learn more about this later, but we can begin right away by saying that there are three essential criteria.

The first is that of the formulation, which is related to the type of element. It should be noted that even if in principle it is convenient to be able to decide, given a particular problem, what kind of element is most suitable, in practice the choice of the type of finite element is quite limited in many of the most widely available software programs, which actually make use of only one or two types of element for every possible formulation. This situation changes radically when one of the more advanced software programs is used: these allow access to many different types of elements for a given problem, which implies a thorough knowledge of the differences between the various types of elements, a knowledge that usually only specialists have. Non-specialists are better off working with a few dependable elements whose behaviour they know well rather than attempting to work with very complicated systems without having the requisite background.

The second criterion is that of *dimension*. In general we can say that the greater the rapidity with which the stresses change (the greater the *stress gradient*), the smaller the finite element has to be, and thus, the greater the number of elements in the area under examination.

The third criterion is that of *regularity*. As we will see, the ideal shape for finite elements is the most regular one, with sides that do not vary too much in dimension or angles that deviate too much from 60° (for triangular elements) or 90° (for quadrangular elements).

4.2.3 THE COORDINATE AND CONNECTIVITY TABLES

Once the formulation and the nodes through which the finite element has to pass are known, then its geometry is also completely known.

If we leave for a moment our individual finite element and go back to examine the situation of the structure, we can easily compile a list of the structure's nodes, that is, a list of points identified by a unique marker, and by a position.

The position of each node is identified by its coordinates (x, y, z), so that if "n" is the number of nodes of a structure, in general "3n" coordinates are necessary to identify the nodes. The table that lists each node numbered progressively along with its the coordinates is called the *coordinate table* and is certainly one of the fundamental data used by every finite element analyst (e.g. Table 4-1).

NODE	X	Y	Z	TX	TY	TZ	RX	RY	RZ
7	0.000E+00	0.000E+00	0.000E+00	FIXED	FIXED	FIXED	FIXED	FIXED	FIXED
8	3.000E+03	0.000E+00	0.000E+00	FIXED	FIXED	FIXED	FIXED	FIXED	FIXED
9	3.000E+03	3.000E+03	0.000E+00	FIXED	FIXED	FIXED	FIXED	FIXED	FIXED

Table 4-1: Nodes 7-9 of a coordinate table in SARGON; in addition to the coordinates the constraints have also been listed.

With this table, when the progressive number of a node is known, then its position is established. *The number of each node acts as a unique identifier that distinguishes it from all others.*

We said that three coordinates are always necessary to identify the position of a point, even if in certain cases the problems being studied are planar. Some finite element software programs exploit the fact that the structural problem being studied is two-dimensional[2] (either because the frame or truss is planar, or because the problem is planar in its stresses or strains, or because the problem being studied is a thin plane or is axially symmetric) requiring only two of the three coordinates. This approach was justified at the dawn of computers when the difficulties related to the input of data on one hand and a shortage of RAM on the other made it necessary to limit the information furnished. Today, however, breaking up the work of a program into many submodes depending on the planarity of the problem at hand creates more disadvantages than advantages: given the amounts of RAM available, CPU capacity and instruments for graphic interface, we might as well always describe a problem by inserting it into a three-dimensional space, eliminating explicitly, if anything, the non-necessary degrees of freedom. Further, in the important area of frames and metal trusses, the hypothesis about plane stress does not limit the three-dimensionality of problems of stability normally addressed: in this context the use of a planar schematisation can even be misleading.

We find an important special case when in the coordinate table there exist two nodes that have different numbers but the same coordinates (that is, two nodes that occupy the same position in space). Two such nodes are called *double nodes*. Later on we will see that double nodes can sometimes be desirable, but they can also greatly hinder the problem solving.

Let's return now to the situation as seen from the point of view of the individual element.

It should be noted that there is a unique correspondence between the nodes in the domain and the nodes of a given element. In fact, the elements used to

[2] By a two-dimensional problem we mean one in which the structure is planar.

model the structure are numbered, and the nodes of the element are purposely marked with node IDs (generally by using letters – i, j, k, l, m, n…, – but also referring to them *locally* as nodes N1, N2, N3, etc.). When we say that the ID identifies a node *locally*, we mean that it does not take into consideration the fact that the element is positioned in a given structure with a given nodal numeration. It is very important to understand this thoroughly. When we speak of node "j" of a certain element, we mean the second node of that element, independent of the particular node on the particular structure to which node "j" corresponds. In evaluating what takes place within a finite element, we have to "forget" what takes place all around it, and thus we need a local system of reference and a local numeration in order to be able to formulate the problem for that kind of element definitively and generally.

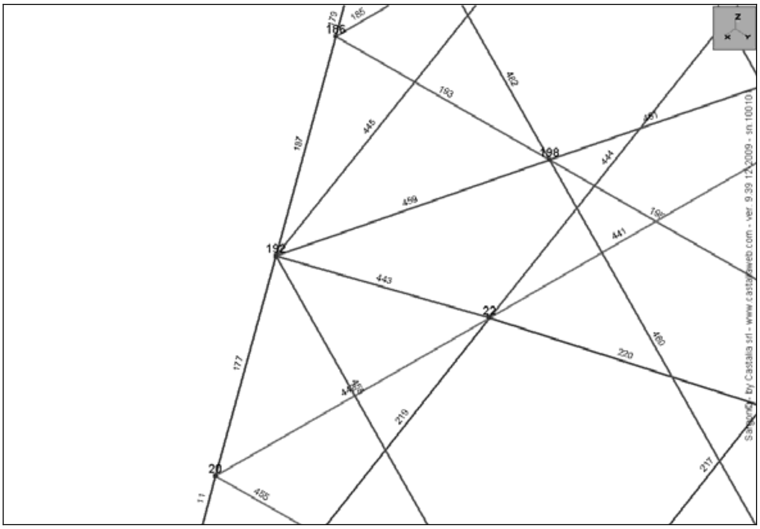

179
185
186
193
462
187
445
198
461
444
459
199
441
192
443
22
460
220
177
458
219
20
217
11
455
Z
X
Y
Sargon® - by Castalia srl - www.castaliaweb.com - ver. 9.39 12/2009 - sn.10010

Figure 4-2: Example of connection in a truss.

The nodes *of the structure* that serve to describe the elements are also identified by means of a progressive numeration (1, 2, 3, 4, …) which is unique and relates to the structure as a whole. Thus (see figure 4-2) node "j" of element 177 is uniquely identified as node 192 in the overall numeration. Note that the correspondence is not two-way; there can be various nodes of various elements that correspond to node 192 of the structure; all that is needed for this to happen is for the various elements all to have that node in common. For example, in fig. 4-2 we see that in addition to element 177, elements 458, 443, 459, 445 and 187 all have node 192 in common (some elements as node "i", others as node "j").

Element 177, node j	→	Node 192
Element 458, node j	→	Node 192
Element 443, node i	→	Node 192
Element 459, node i	→	Node 192
Element 445, node j	→	Node 192
Element 187, node i	→	Node 192

Up to now we have said that the shape of a given element is determined by its type and by the position of its nodes. Thus, in order to know the geometry of an element it suffices to know the global number attributed to each of the nodes that describe the element. If a certain element requires nodes (i, j, k, ...), it is possible to substitute for these local nodes (i, j, k, ...) the unique global numeration of the corresponding nodes, thus arriving at the definition of the concept of *element connectivity*: the element connectivities are defined by the ordered set of global numbers associated to local nodes (i, j, k...) of the element itself. The set of connectivities of all elements is a table called the *connectivity table*, which constitutes a datum of fundamental importance for all users of the FEM.

<table>
<tr><th colspan="5">----------CONNECTIVITY SECTION----------</th></tr>
<tr><th>PLATE</th><th>N1</th><th>N2</th><th>N3</th><th>N4</th></tr>
<tr><td>1</td><td>7</td><td>11</td><td>17</td><td>16</td></tr>
<tr><td>2</td><td>11</td><td>12</td><td>18</td><td>17</td></tr>
<tr><td>3</td><td>12</td><td>13</td><td>19</td><td>18</td></tr>
<tr><td>4</td><td>13</td><td>14</td><td>20</td><td>19</td></tr>
<tr><td>5</td><td>14</td><td>15</td><td>21</td><td>20</td></tr>
</table>

Figure 4-3: Connectivity table in Sargon, in reference to plate elements. The local nodes are called "N1", "N2", etc.

Thanks to the incidence table and the coordinate table we can construct the *mesh*, that is, the set of all the finite elements that describe the structure, along with the areas where they are located, dimensions, and exact positions. The creation of a good structural mesh is the first intermediate result of an analysis, and in large part the final outcome depends upon it. In order for an analysis to be meaningful, the mesh has to be constructed with care.

4.2.4 CONNECTION

We have seen that in order to define the shape of an element whose type is known, we must know what its nodes are, and then, once the coordinate table is known, find its connectivities. The nodes of each element not only serve to define the shape, but also serve to establish its *connections*. In simplified terms, we can say that the node is *connected* to the element since it is attached to the element such that, if the node moves, *it transmits at least a part of its movement to the element itself.*

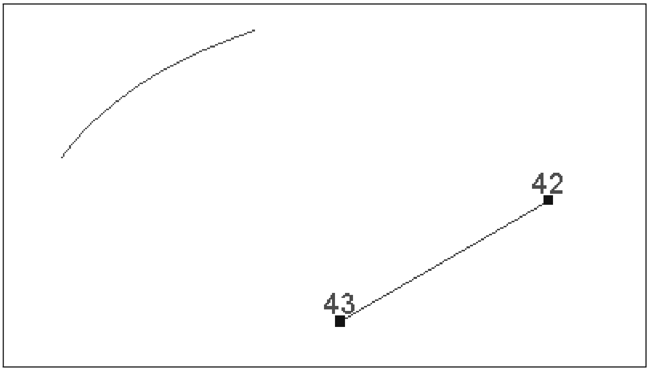


Figure 4-4: Connection. The rotations and translations of nodes i (43) and node j (42) of the elements are transmitted to points of the element (see the deformed configuration).

What do we mean when we say that the point of the element identified by the node has to transmit *at least a part* of its displacement? Might there be cases in which one of the movements of a node connected to an element does not transmit anything to that element?

Yes, this can occur, and not having understood this fact is one of the most common errors made by those lacking in expertise in finite elements.

For example, the rotation of a node transmits nothing at all to a rod element, since by definition this has no stiffness in bending. *The node rotates, but nothing at all happens to the element.* In contrast, if this same node were connected to an unreleased beam element, the same rotation of the node would give rise to effects in the beam element, whose axis line would be subject to the same rotation.

The rotation of a node about an axis normal to the plane of a membrane doesn't provoke anything at all in the membrane[3] (see fig. 4-6). The translation of a node to which a certain number of rods in the same plane are connected, in a direction normal to that plane, does not provoke anything in the rods. The rods, on the other hand, do not offer any resistance to this movement (see fig. 4-5).

[3] The membrane elements described in §11.5.3. (cf. Taylor and Simo [28]), which also use this degree of freedom in their formulation, are exceptions to this rule.

How much of the displacement is shared by the node and the element is determined by the type of finite element and by the given problem. Later, when we examine the most frequently used types of finite elements in detail, we will see clearly what the significant displacements are.

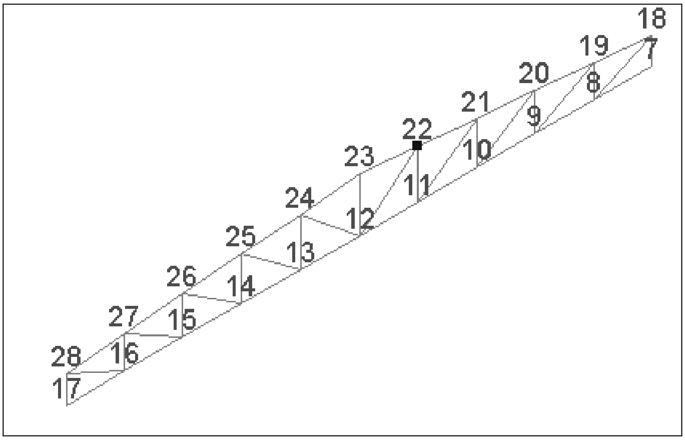


Figure 4-5: In this truss modelled with rod elements all the rotations of the nodes are free. All of the translations in a direction normal to the plane of the truss are free.

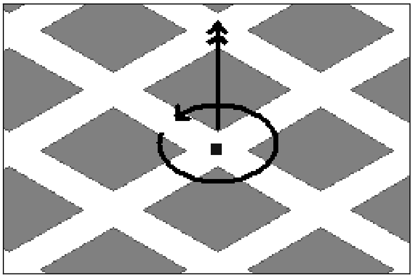


Figure 4-6. The rotation indicated by the given node does not communicate anything to the membrane connected to that node, since in the formulation of the element, rotation does not play a role (but see foot note 3). Note that the elements shown are "shrunken" for the sake of clarity.

By definition, each node of an element is connected to it, that is, it allows at least part of its displacement to be transmitted unaltered to the corresponding point of the element.

More generally, we can say that there are rules that relate the displacements of the nodes to the displacements of points of the elements: these rules concern not only the points of the elements that correspond to the nodes, but also *the points within the element itself that do not correspond to the nodes*. This will bring us to an examination of the essential core of the FEM.

4.2.5 The basic hypotheses: The "Astrological principle"

What happens in the points in a finite element? Strictly speaking we cannot say for sure unless we completely solve the system of differential equations that solve the problem at hand (of beams, of plates, of plane stress, etc.). Answering this question means facing insurmountable difficulties of calculation, since when the form of the domain is complex it is practically impossible to arrive at a closed-form solution for the system of solution equations.[4]

In general, each component of unknown displacement is a complex function of the point's position within the element, of the constraints, and of the applied loads. We can simplify the problem by making it so that the functions being sought are obtained by combining *known* functions by means of a certain number of coefficients whose combination is *unknown*. The procedure is qualitatively similar to that of series: a "strange" function is expressed as the weighted sum of certain simple functions.

It needs to be understood (we will come back to this in more detail further along) that the set of displacement functions that have to be found changes according to the kind of problem being studied. In some cases, precisely because of the type of problem, certain displacements are not relevant inasmuch as they are not part of the unknowns (see Table 4-1). For example, while it is true that in a generic problem of three-dimensional elasticity the unknown functions are the three translations of point P of the solid u(P), v(P), w(P) and the rotations are irrelevant, for a Euler-Bernoulli beam located in space the unknown functions are the three translations u, v, w (associated with the elongation of the beam and with the bending in the two principal planes) and a rotation θ (associated with torsion) of the points *on the axis of the beam*. In a rod the only unknown function is the displacement u(x) of the points *on the axis* in the direction of the axis of the rod itself. In a flat plate, in order to know the bending behaviour it is sufficient to know the displacements w(x, y) *of the points of the median plane* in the direction normal to that plane. In a problem of a planar membrane we need only know the displacements u(x, y) and v(x, y) *of the points of the membrane's median plane* in the plane of that membrane. And so on.

It is thus clear that the number and the type of unknown functions (translations or rotations) depend on the type of elasticity problem being studied and by the kind of domain that has to be discretised (whether one-, two- or three-dimensional).

Since we have the list of the unknown displacement functions, we can imagine expressing the unknown functions by means of the weighted sum of suitable known functions. *Practically speaking, we limit ourselves to leaving the numbers*

[4] On the other hand, there are various solutions to classic problems of elasticity when the form of the domain is simple or regular. Scientists worked for quite a long time to find these solutions.

as unknowns while supposing that we know the functions. Let's consider that the displacement function of z(P) in a given element related to a certain problem of elasticity is unknown (we are deliberately using a symbol that does not appear in some of the formulations in order to underline the fact that we are speaking in general terms). The idea that the unknown function can be expressed as the weighted sum of known functions can be expressed by writing

$$z(P) \approx a_1\Pi_1(P) + a_2\Pi_2(P) + a_3\Pi_3(P) + \dots a_n\Pi_n(P). \qquad [4\text{-}1]$$

Equation 4-1 expresses the fact that the unknown function z(P) is *arbitrarily* presumed to correspond to the sum of a certain number of *known functions* Π multiplied by the appropriate number of *unknown coefficients* "a". Note that while the Π functions are just that – functions –, the weights "a" are numbers. It is well known that when one or more numbers are unknown, we are dealing equations, while if one or more functions are unknown, we are dealing with differential equations. Equation 4-1 makes it possible for us to transform differential equations (unknown functions) into algebraic equations (unknown numbers). In fact, in all of the equations in which z(P) appears we can write its approximation from equation 4-1, thus arriving at a set of equations in which the only unknowns are the weights "a".

We see that in equation 4-1 we have only made use of the fact that z(P) has to be expressed by means of the sum of a certain number of known functions weighted by unknown coefficients. Let's now take the next step: *let's imagine that if we know the displacements of its nodes, then we know everything that takes place in terms of displacements within a finite element.* In the introduction we have referred to this idea as the "astrological principle" to underline the fact that at the heart of the FEM there is really a presumption: we presume to know what takes place within an element because we know the movements of a few of its points (the nodes). From a formal point of view this second presumption leads to the establishment of these conditions by the Π functions and the "a" weights.

$$\begin{aligned} z(P_1) &= a_1\Pi_1(P_1) + a_2\Pi_2(P_1) + a_3\Pi_3(P_1) + \dots a_n\Pi_n(P_1) \\ z(P_2) &= a_1\Pi_1(P_2) + a_2\Pi_2(P_2) + a_3\Pi_3(P_2) + \dots a_n\Pi_n(P_2) \qquad [4\text{-}2] \\ &\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ z(P_n) &= a_1\Pi_1(P_n) + a_2\Pi_2(P_n) + a_3\Pi_3(P_n) + \dots a_n\Pi_n(P_n). \end{aligned}$$

If we make it so that points P coincide with the nodes, then the displacements of the first member of equation 4-2 are in fact the displacements of the nodes, or briefly, *the nodal displacements.* For the sake of brevity we write $z(P_i) = z_i$, and we call z_i *nodal displacement i.* These conditions express the element's nodal displacements in function of the original weights "a". If we accept equation 4-1, these tell us what the value of the unknown displacement is in the various nodes of the element.

In equation 4-2, since the quantities $\Pi(P_i)$ are known functions calculated at known points, they are known numbers.

Note that among the "nodal displacements" we can, if necessary, list the derivatives of the displacements as well. To do so, we need only write conditions such as

$$z'(P_1)= a_1\Pi'_1(P_1) + a_2\Pi'_2(P_1) + a_3\Pi'_3(P_1) + \dots a_n\Pi'_n(P_1)$$

or

$$z''(P_1)= a_1\Pi''_1(P_1) + a_2\Pi''_2(P_1) + a_3\Pi''_3(P_1) + \dots a_n\Pi''_n(P_1)$$

in which the sum weighted by the *derivatives* of the interpolation function is identified with the required derivative of the displacement of the given point.[5] In this case, certain *displacement derivatives* also figure among the "nodal displacements", derivatives which, for example, can have the physical meaning of *rotations* in certain cases.

We see that if *n* is the number of elementary functions used, and thus the number of unknown coefficients, then *n* is also the number of the relations that have to be written to eliminate *all* of the original parameters in order to substitute them with the nodal displacements. *Thus the unknown parameters become the nodal displacements.*

As we can see, we obtain a system of equations – 4-2 – that link the weights a_i to the nodal displacements z_i . Thanks to these relations – which we can solve in function of the weights a_i – we can express the weights a_i as functions of the nodal displacements z_i. In general we obtain relations such as

$$\begin{aligned} a_1 &= z_1c_{11} + z_2c_{12} + z_3c_{13} + \dots z_nc_{1n} \\ a_2 &= z_1c_{21} + z_2c_{22} + z_3c_{23} + \dots z_nc_{2n} \qquad [4\text{-}3] \\ &\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ a_n &= z_1c_{n1} + z_2c_{n2} + z_3c_{n3} + \dots z_nc_{nn} \end{aligned}$$

where the coefficients c are the known numbers (such as $\Pi(P_i)$ in equations 4-2).

If we want to write equations 4-3 more concisely, we can organise the parameters a_i in a (*column*) vector of parameters **a**, the nodal displacements in a (*column*) vector of nodal displacements **z**, and the coefficients c in a square matrix **A**. Thus equation 4-3 becomes

$$\mathbf{a} = \mathbf{Az}. \qquad [4\text{-}3a]$$

By substituting the equations 4-3 in equations 4-1, we obtain the canonical form of the *astrological principle:*

[5] For the sake of simplicity we have presumed total derivatives, but they can also be partial derivatives.

$$z(P) \approx z_1\Phi_1(P) + z_2\Phi_2(P) + z_3\Phi_3(P) + \dots z_n\Phi_n(P), \qquad [4\text{-}4]$$

or, by arranging the functions Φ_i in a (*row*) vector $\boldsymbol{\phi}^{\mathbf{T}}$,

$$z(P) = \boldsymbol{\phi}^{\mathbf{T}}\,\mathbf{z}. \qquad [4\text{-}5]$$

We can see that since the coefficients "a" are a linear combination of the nodal displacements u (by equations 4-3), the functions Φ are in their turn linear combinations of the original functions Π.

Let's look closely at formula 4-4: we find there the Φ functions, which are called *shape functions*, and the *nodal displacements* z. Equation 4-4 tells us that *the displacements of a generic point P can be calculated if we know the displacements of the nodes of the finite element.* In fact, if the nodal displacements are known, then by applying equation 4-4 it is possible to calculate the displacements in any point of the element whatsoever. It is worthwhile to underline that equation 4-4 *is only an approximation, which can be reasonably verified if certain conditions exist.*

Looking at equation 4-4, we can also see that there one important condition has to be present. The shape functions have to be such that it must be true that the *i*th shape function has to equal 1 in the *i*th node and 0 in any other node that is not the *i*th:

$$\Phi_i(P_i) = 1$$

$$[4\text{-}6]$$

$$\Phi_i(P_j) = 0.$$

If this were not the case, then it would be impossible that $z(P_i)=z_i$, a condition that we accepted by definition when we introduced nodal displacements. The conditions just written are actually the conditions of congruity: they guarantee that the displacement field respects the nodal displacements.

Example

Let us consider a rod with two nodes, with coordinates x=0 and x=L. For the unknown displacement u we set

$$u(x) = a_1+a_2x \qquad [4\text{-}7]$$

that is, recalling equation 4-1

$$\Pi_1=1,\ \Pi_2=x.$$

Now let us recall the definition of nodal displacement, and applying equation 4-7

$$u_1=a_1$$

$$u_2= a_1+a_2L$$

and thus, expressing this in terms of the parameters a:

$$a_1=u_1$$

$$a_2= (-u_1+u_2)/L.$$

Substituting this in equation 4-7 we obtain

$$u(x) = u_1+(u_2-u_1)x/L= (1-x/L)u_1+ (x/L)u_2$$

which implies that

$$\Phi_1 = (1-x/L)$$

$$\Phi_2 = x/L.$$

As we can see, the shape functions automatically satisfy the conditions in 4-6.

♣

In what has just been explained we have seen that the number of the elementary functions Π that are combined is equal to the number of nodes of the element. In fact, the number of weights in 4-1 is equal to the number of elementary functions, and the number of nodal displacements necessary to express all the weights in equation 4-2 is equal to the number of those weights. *Thus the number of available nodes determines the number of elementary functions that can be combined:* the greater the number of available nodes, the greater the number of elementary functions. On the other hand, the greater the number of elementary functions, the more articulated and complex the prediction that can be made by equation 4-4, and thus the better – generally speaking – the performance of the finite element.

For example, in the case of a rod with three nodes, for a *linear* approximation for the displacement field as in equation 4-7 for the rod *with two nodes*, we can substitute a *parabolic* approximation that allows us, for instance, to comprehend correctly the behaviour of a rod subject to a uniformly distributed axial force (suspended rod). In the case in which the rod itself is modelled with elements with two nodes, the unknown (parabolic) displacement field must be approximated by means of as many straight line segments as there are finite elements with two nodes used to model the rod.

♣

Up to now we have talked about an unknown displacement function "z(P)" and we have tacitly admitted that this is the only unknown. In reality, depending on the type of problem, we can have various unknowns, and thus need to model more than one function. This doesn't necessarily represent a problem. In fact, we need only associate to each unknown function the nodal displacements proper to it in order to arrive at as many "astrological principles" (equation 4-4) as we need. For example, in a problem of plane stress in plane (x, y) the displacement functions (u, v) are unknown. At each node "i" we can consider the displacement in direction x, u_i, and the displacement in direction y, v_i, and write:

$$u(P) \approx u_1\Phi_1(P) + u_2\Phi_2(P) + u_3\Phi_3(P) + \dots u_n\Phi_n(P)$$

$$v(P) \approx v_1\Phi_1(P) + v_2\Phi_2(P) + v_3\Phi_3(P) + \dots v_n\Phi_n(P)$$

or, in matrix form, having made the vector of unknown displacement functions $\mathbf{U}=\{u(P), v(P)\}^T$

$$\mathbf{U} = \boldsymbol{\Phi}\mathbf{u_e} \qquad [4\text{-}8]$$

where this time

$$\boldsymbol{\Phi}=\begin{vmatrix} \Phi_1 & 0 & \Phi_2 & 0 & \ldots. & \ldots. & \Phi_n & 0 \\ 0 & \Phi_1 & 0 & \Phi_2 & \ldots & \ldots & 0 & \Phi_n \end{vmatrix}$$

$$\mathbf{u_e}=\{u_1, v_1, u_2, v_2, \ldots, \ldots, u_n, v_n\}^T.$$

This reorganisation of the equations in matrix form 4-8 doesn't tell us anything different from what we have already established. It simply allows us to rewrite concisely the relations which in substance we have already understood but which, essentially because of the number of different functions to look for, and of the different number of nodes to take into account, can become quite numerous.

Equation 4-8 represents the fundamental result of the approximations introduced. The subscript "e" indicates that we are referring to the nodal displacements of a given finite element.

♣

The system of hypotheses outlined here is that used historically when the FEM was in its beginnings, and is still used in many of the most frequent finite elements belonging to the various formulations (rod, beam, three- or four-node membrane, three- or four-node plane strain, solid problems using elements with four nodes, etc.). However, it is not the only one. In some cases only some of the parameters "a" can be associated with nodal displacements "u", while others have no immediate geometric interpretation. In general, these parameters, which are in addition to those of nodal displacements, do not cause any particular problems, and are dealt with from the point of view of analysis in the same manner as normal unknown nodal displacements.

There are other cases when, even if the number of nodes used to express the approximation of the displacement field is equal, for example, to *m*, only *n* of these are effectively used in connecting the element to the rest of the structure: in this case we speak of *condensation*, indicating that fewer parameters for calculation are used to describe the external behaviour of the element than are used in its formulation.

There are still other cases which use a rather high number of parameters, and which establish ulterior conditions for regularity in some points of the element in such a way that it is possible to keep only part of the original parameters as unknowns, that is, only the "final" nodal displacements of the element.

Thus we can see that there are numerous techniques, just as there are numerous and diverse formulations of the various finite elements, even when the number of nodes remains the same.

Here it sufficient to observe that each finite element is identified by a set of arbitrary hypotheses concerning the displacement field and/or the way that appropriate conditions for regularity are established. Such hypotheses make it possible to express all the relevant mechanical quantities such as strain energy, strains or stresses in function of a limited number of unknown numeric parameters, which generally coincide with the nodal displacements.

4.3 Formal developments: finite element analytics

4.3.1 Congruity

Now that we have introduced the hypotheses that make it possible for us to describe what takes place within a finite element solely as a function of nodal displacements, we can take advantage of these simplifying hypotheses to arrive at a set of important results.

The passage from displacements to strains is made, for each possible formulation, thanks to congruity relations. From the point of view of mathematics, the congruity relations tell us that the strains can be obtained starting from the displacements by taking a certain number of derivatives. For example, in the elongation of a beam, when the displacement u(x) is known, the strain is obtained from the first derivative, u′(x). In Euler-Bernoulli bending, when the translation of axis w (the elastic line) is known, we can obtain the strain from the second derivative w″(x). In plane stress theory, when the displacements u(x, y) and v(x, y) are known, the strains are obtained from suitable partial derivatives of u and v, such as $\gamma_{xy}=0.5(u_{,y}+v_{,x})$. And so on.

If $\mathbf{U}$ is the vector of unknown displacements for the formulation being studied, vector $\boldsymbol{\varepsilon}$[6] of the generalised strains associated with that formulation can thus be obtained by means of the fundamental relation

$$\boldsymbol{\varepsilon} = \mathbf{CU} \qquad [4\text{-}9]$$

where the rectangular matrix $\mathbf{C}$ is a matrix of operators, that is, each of its elements is a certain derivative applied to a suitable component of displacement. If *nf* is the number of functions of unknown displacements and *ns* the number of

[6] In order to avoid notation that is too elaborate we have used symbol ε for generalised strains as well. It should be understood the strains "ε" can be defined in different ways according to the kind of formulation.

unknown generalised stresses or strains, then matrix **C** has *ns* rows and *nf* columns, and thus is of the type (*ns*, *nf*).

For example, for plane stresses in plane (x, y), *ns*=3, *nf*=2:

$$\{\varepsilon_x, \varepsilon_y, \gamma_{xy}\}^T=\boldsymbol{\varepsilon}=\left|\begin{array}{c:c} \dfrac{\partial(\cdot)}{\partial x} & 0 \\ \hdashline 0 & \dfrac{\partial(\cdot)}{\partial y} \\ \hdashline \dfrac{\partial(\cdot)}{\partial y} & \dfrac{\partial(\cdot)}{\partial x} \end{array}\right| \left|\begin{array}{c} u(x,y) \\ \hdashline v(x,y) \end{array}\right| = \mathbf{CU}.$$

These relations simply express in matrix form, and therefore concisely, what the various formulations say about unknown displacement fields, congruity relations, and generalised strains.

Now, recalling the result obtained in the previous section, we can write

$$\boldsymbol{\varepsilon} = \mathbf{CU} = \mathbf{C\Phi u}_e$$

and defining rectangular matrix **B** as

$$\mathbf{B}=\mathbf{C\Phi} \qquad [4\text{-}10]$$

we finally obtain

$$\boldsymbol{\varepsilon} = \mathbf{B}\ \mathbf{u}_e. \qquad [4\text{-}11]$$

*Rectangular matrix **B** is a matrix of known functions.* It is obtained by applying the derivatives required by the congruity relations to the (known) shape functions Φ. Thanks to these passages, analogous to those we have already seen for the displacements, we are able to *express the strains by means of known functions (appropriately derived from the shape functions) and unknown numeric coefficients, the same nodal displacements* $\mathbf{u}_e$ *that we use to express the displacement functions.*

4.3.2 Constitution

Knowing the strains immediately opens the way to obtaining the stresses. In fact, there is a quite precise relationship between the vector of generalised stresses **σ** and the vector of generalised strains **ε.** This relationship expresses the generalisation of Hooke's law by means of a certain number of suitable numeric coefficients depending on the material and the geometry. This relationship is linear, and can be written concisely as:

$$\boldsymbol{\sigma}=\mathbf{D}\boldsymbol{\varepsilon}. \qquad [4\text{-}12]$$

Matrix **D**, *square and symmetrical*, expresses this relationship between stresses and strains. This matrix is made of numbers that we are perfectly capable of calculating when the characteristics of the material and the geometry of the structure are known. But then, for equations 4-11 and 4-12 we can write

$$\boldsymbol{\sigma}=\mathbf{DBu}_e, \qquad [4\text{-}13]$$

relations that also make it possible to express the stresses as a function only of the unknown nodal displacements $\mathbf{u}_e$*.*

We have thus arrived at the conclusion that the displacements, strains and stresses within the finite element can all be expressed by means of known functions and coefficients, and by means of the unknown nodal displacements. The *Astrological Principle* has been taken to its extreme: *if we know the nodal displacements, then we are able to calculate approximately what takes place within the finite element.*

4.3.3 ENERGY ASPECTS

Now let's take what we have found up to this point and apply it to define the situation of a finite element in equilibrium after application of the loads.

There are two ways of studying the situation of equilibrium of a finite element: by means of the energy method, that is, writing the stationariness of total potential energy, or by means of the static method, that is, setting in equilibrium all the forces acting on the element itself.

The strain energy stored in a finite element following the strains to which it has been subjected can be written as

$$Z= 0.5\int \boldsymbol{\varepsilon}^T \boldsymbol{\sigma} dV.$$

This means taking the integral sum of all the "volumes" dV of the finite element (the "volumes" can be a segment of volume, or a segment of surface or of line, depending on the kind of problem being studied) of the elementary strain energies. In this expression, $\boldsymbol{\varepsilon}$ and $\boldsymbol{\sigma}$ are *generalised* strains and stresses.

Applying what we have found up to this point, the strain energy can be written[7]

$$Z= 0.5\ \mathbf{u}_e^T\ (\int \mathbf{B}^T\ \mathbf{D}\ \mathbf{B}\ dV)\mathbf{u}_e.$$

This important relation was found by simply substituting the simplified expressions found for stresses and strains in the expression for strain energy, and placing the values that don't depend on the point in the element outside the integral sign. In this case as well we have obtained the fundamental result of making the value of a complex quantity depend essentially on two contributions: the contribution of the integral ($\int \mathbf{B}^T\ \mathbf{D}\ \mathbf{B}\ dV$), which is the *integration of a set of known functions taken inside the finite element*, and the contribution of the nodal displacements $\mathbf{u}_e$, *which are unknown numbers.*

Integral $\int \mathbf{B}^T\ \mathbf{D}\ \mathbf{B}\ dV$ is the integral of a square matrix of the same order as vector $\mathbf{u}_e$. The integral is simply a matrix whose components are the integrals of the individual terms. Matrix $\int \mathbf{B}^T\ \mathbf{D}\ \mathbf{B}\ dV$, *square and symmetrical*, is called *stiffness matrix* $\mathbf{K}_e$ of the finite element. Thus we can write

[7] Recalling that $(\mathbf{Bu})^T=\mathbf{u}^T\ \mathbf{B}^T$.

$$\mathbf{K}_e \equiv \int \mathbf{B}^T \mathbf{D}\, \mathbf{B}\, dV \qquad [4\text{-}14]$$

$$Z = 0.5\, \mathbf{u_e}^T\, \mathbf{K}_e\, \mathbf{u_e}. \qquad [4\text{-}15]$$

To obtain the strain energy in the deformed configuration of a finite element, starting from its nodal displacements, the element's stiffness matrix is pre- and post-multiplied by the vector of nodal displacements, and the number found is divided by 2. Even though it is obtained by multiplying a matrix by vectors, the strain energy Z is a number, a scalar quantity, as required by definition. $\mathbf{K}_e\, \mathbf{u_e}$ is a column vector and $\mathbf{u_e}^T\, (\mathbf{K}_e\, \mathbf{u_e})$ is a number.

It is easy to see that the matrix has to be symmetrical:

$$\mathbf{K}_e = \mathbf{K}_e^{\mathbf{T}}.$$

We need only transpose equation 4-14 and remember that **D** itself is symmetrical.

Let's look now at the applied loads. Loads can be divided into two categories: on the one hand there are the known "volume" forces **b** and the known "surface" forces **p**; on the other hand there are the unknown actions **s** that the finite element exchanges with those adjacent to it. Both of these cases concern functions that we have ordered in a vector that has the same characteristics as the vector of displacement **U**. Forces **b** and **p** are *known* applied actions; forces **r** are unknown reactions that obey the principle of action/reaction: they act in a certain direction on our finite element, and in the opposite direction on the adjacent finite element.

Let's look first at the first set's contribution to the work. If we use the symbol V for "volume" and S for "external surface" of the finite element, meaning the element's domain (be it a line, surface or volume) and its boundaries (be they points, lines or surfaces), we can write the first of the two contributions as

$$W_p = \int \mathbf{U}^T \mathbf{b} dV + \int \mathbf{U}^T \mathbf{p} dS$$

that is, by first multiplying the elementary volume forces **b**dV by the corresponding displacements **U**, and then taking the integral sum, and next multiplying the elementary surface forces by the corresponding displacements **U** and then taking the integral sum.

Taking into account that the displacement functions can be expressed by means of equation 4-8, the work can be written

$$W_p = \mathbf{u_e}^T \left(\int \boldsymbol{\Phi}^T \mathbf{b} dV\right) + \mathbf{u_e}^T \left(\int \boldsymbol{\Phi}^T \mathbf{p} dS\right).$$

Once again everything is made to depend on the displacements $\mathbf{u_e}$. The quantities defined here

$$\mathbf{f}_b \equiv \left(\int \boldsymbol{\Phi}^T \mathbf{b} dV\right)$$

$$\mathbf{f}_p \equiv \left(\int \boldsymbol{\Phi}^T \mathbf{p} dS\right)$$

are two vectors of numbers, perfectly capable of being calculated beginning with the known functions **p**, **b**, and **Φ**. If there are *n* unknown displacement functions and *m* nodal displacements, then matrices **Φ** have *n* rows and *m* columns, and thus the product $\mathbf{\Phi}^T\mathbf{p}$ has the following dimensions

$$(n, m)^T(n, 1) = (m, n)\ (n, 1) = (m, 1)$$

that is, both $\mathbf{f}_b$ and $\mathbf{f}_p$ have the dimensions of a vector with *m* rows (a column vector).

We can then write

$$W_p = \mathbf{u_e}^T(\mathbf{f}_b + \mathbf{f}_p)= \mathbf{u_e}^T\mathbf{f_e}.$$

Since we set

$$\mathbf{f_e}\equiv \mathbf{f}_b + \mathbf{f}_p$$

the vector is called *the vector of equivalent nodal forces*. If $W_p= \mathbf{u_e}^T\mathbf{f_e}$, then the work performed by the volume and surface forces is obtained by multiplying the *nodal* displacements $\mathbf{u_e}$ by the *nodal* forces $\mathbf{f_e}$ so that this product of the forces and displacements of the nodes gives rise to the work performed by the loads applied on the element as a whole. In reality, the $\mathbf{f_e}$ do not exist; they *are not applied on the nodes of the actual structure*. They are fictitious forces which have, however, as far as work is concerned, the same effects that volume and surface forces weighted by the corresponding displacements would have. They are *equivalent* nodal forces.

Applying this same logic to the reaction forces **s** as well, it is easy to intuit that we can say that the contribution W_r of the reaction forces to work W as a whole is obtained by

$$W_r = \mathbf{u_e}^T\mathbf{r_e}$$

since we established

$$\mathbf{r}_e\equiv(\int\mathbf{\Phi}^T\mathbf{s}dS).$$

In contrast to the forces $\mathbf{f_e}$, the nodal forces $\mathbf{r_e}$ are completely unknown. The total potential energy therefore becomes (recalling that Π=Z-W)

$$\Pi= 0.5\ \mathbf{u_e}^T\ \mathbf{K}_e\ \mathbf{u_e} - \mathbf{u_e}^T\mathbf{f_e} -\mathbf{u_e}^T\mathbf{r_e}. \qquad [4\text{-}16]$$

4.3.4 Equilibrium

If we want to establish equilibrium we have make Π invariable for a generic variation of configuration (the principle of stationariness of total potential energy). Now, the configuration – that is, the set of displacements – is completely contained in the vector of nodal displacements, since we presume that the *shape* of the displacement field does not vary, because it is always described by the same shape functions. *Therefore, in going from one configuration to another the only thing that changes are the nodal displacements, which are solely responsible for* *the quantity of displacement within our finite element.* The variation in total potential energy is thus equal to

$$\delta\Pi = 0.5\, \boldsymbol{\delta u_e}^T \mathbf{K}_e \mathbf{u_e} + 0.5\, \mathbf{u_e}^T \mathbf{K}_e \boldsymbol{\delta u_e} - \boldsymbol{\delta u_e}^T\mathbf{f_e} - \boldsymbol{\delta u_e}^T\mathbf{r_e}.$$

Given that $\mathbf{K}_e$ is symmetric, and that the transpose of a number is still the same number, then the second term,

$$0.5\, \mathbf{u_e}^T \mathbf{K}_e \boldsymbol{\delta u_e},$$

can be written equivalently as

$$0.5\boldsymbol{\delta u_e}^T \mathbf{K}_e \mathbf{u_e}$$

and thus

$$\delta\Pi = \boldsymbol{\delta u_e}^T \mathbf{K}_e \mathbf{u_e} - \boldsymbol{\delta u_e}^T\mathbf{f_e} - \boldsymbol{\delta u_e}^T\mathbf{r_e}.$$

Factoring out the vector $\boldsymbol{\delta u_e}^T$, we get

$$\delta\Pi = \boldsymbol{\delta u_e}^T (\mathbf{K}_e \mathbf{u_e} - \mathbf{f_e} - \mathbf{r_e}).$$

Since the variation of Π has to be null *for all possible* $\boldsymbol{\delta u_e}$, it necessarily follows that in order for there to be equilibrium, the following fundamental relation has to exist:

$$\mathbf{K}_e \mathbf{u_e} - \mathbf{f_e} - \mathbf{r_e} = \mathbf{0}.$$

That is, in terms of $\mathbf{r_e}$

$$\mathbf{r_e} = \mathbf{K}_e \mathbf{u_e} - \mathbf{f_e}. \qquad [4\text{-}17]$$

This tells us that *the necessary condition for a finite element to be in equilibrium is that the vector of the equivalent nodal reactions is equal to the difference between the elastic recall forces* $\mathbf{K}_e \mathbf{u_e}$ *and the equivalent nodal forces* $\mathbf{f_e}$. This condition corresponds to having established equilibrium in a certain number of points: the nodes. This node equilibrium *averages out in terms of energy to the equilibrium of the finite element as a whole.*

Each row of the vector equation corresponds to writing the equilibrium along one of a node's components of displacement (translation or rotation) and, in scalar terms rather than vector terms, if *i* is the index of the *i*th row, we can write

$$r_i = \Sigma K_{ij} u_j - f_i \qquad [4\text{-}18]$$

(here for the sake of simplicity we have omitted the subscript "e").

Each component of displacement of a generic node of the finite element under study therefore has to be impeded by the equilibrium of three forces:

- The resultant r_i on the node, in the given direction, of the reactions that arrive to the node from adjacent elements attached to it; this force is unknown;
- The resultant f_i of the action directly or indirectly applied to the node via known external loads; this force is known;

- The resultant $\Sigma K_{ij}u_j$ of the elastic recall forces of the given finite element, acting on the given node in the given direction in relation to the displacements to which the element is subject; this is unknown *a priori*, since the displacements are unknown. If the displacements were known, then the force would be known.

If we look at equation 4-17, we can see that it concerns a vector relation. That is to say, there are vectors whose number of components is given by the *N* number of nodes of the finite element by the *n* number of nodal displacements taken as parameters for calculating at each node the unknown displacement functions of the given formulation: *m*=*Nn*.

For example, in a plane problem of *n* = 2 (unknown functions u(x, y) and v(x, y), unknown nodal displacements u_i and v_i, that is, each node's two translations in the plane), if the element has three nodes (*N*=3), then *m*=3x2=6, and equation 4-17 corresponds to 6 distinct equations.

For a solid problem of *n* = 3 (unknown functions u(x, y, z), v(x, y, z), w(x, y, z), unknown nodal displacements u_i , v_i and w_i , that is, each node's three translations in space) and if there are eight nodes (*N*=8), then *m* = 3x8=24, and equation 4-17 corresponds to 24 distinct equations.

For a beam in space with two nodes (*N*=2), with *n*=6 (unknown functions u, v, w, θ and unknown displacements u_i, v_i, w_i, θ_i, φ_i, η_i,[8] that is, each node's three translations and three rotations), then m=12, and there are thus 12 distinct equations.

4.3.5 A SIMPLIFIED MODEL FOR THE STRUCTURE

If we look for a moment at what we have found up to this point, we can see that writing the simplified relations 4-1 meant that we were no longer dealing with the equilibrium of a segment of the domain under study with all of its infinite internal points, but rather with the situation in which a certain limited number of points are considered to be subjected to the action of a given number of forces, some of which are known and some of which are not: *the nodes*.

We can consider the nodes as miniscule rigid cubes, each of which is generally subject to six distinct components of movement, six degrees of freedom. In fact, as far as equilibrium is concerned, these miniscule cubes behave exactly like miniscule rigid bodies in space, and are thus subject to three possible translations and three possible rotations, each of which is independent of the others: *the three nodal translations and the three nodal rotations.*

[8] It should be noted that in the case of beams the number of unknown functions (4) does not coincide with the number of unknown nodal displacements (6) in each node. We will discuss the reasons for this later.

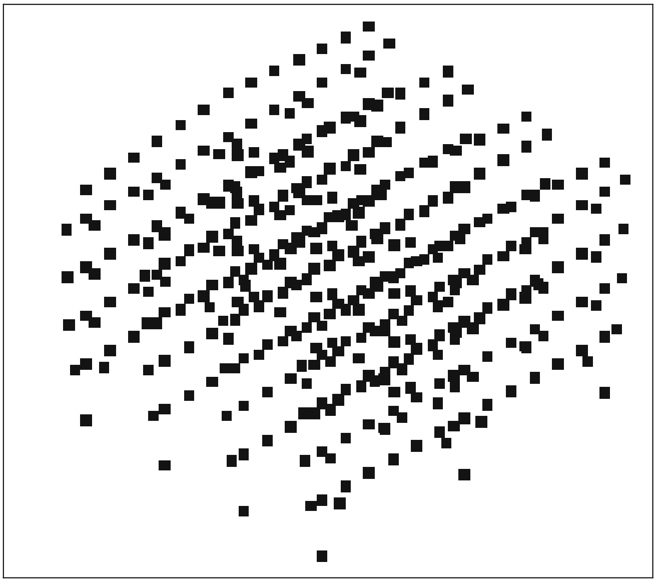


Figure 4-7: The nodes of the model of a building, without a representation of the elements. Each node has six degrees of freedom, and is subject to forces that tend to make it move, and stiffnesses that restrain it. The finite element method imposes equilibrium at each node by calculating the displacements that give forces of elastic recall that are equal to the applied forces.

The nodes of each finite element are induced to move from their original position either by volume or surface forces applied directly to the element, or communicated to the nodes by it, or by actions communicated to the element by adjacent elements, which are in their turn distorted.

The forces that make the nodes move are above all caused by actions applied directly (when the forces are actually applied to the nodes) or indirectly (when these forces derive from the volume and surface forces applied to the element). But they are also caused by the fact that *other elements* are attached to that node, and these other elements in their turn are subject to distortions which they counter with elastic reaction forces: an example of this mechanism is when we place a book on a glass that rests on a table: the table will be pressed down because the glass, compressed by the weight of the book, communicates the actions to the table.

The movement of the nodes of the element in question (caused by the forces listed earlier) is countered by a set of reaction forces due to the fact that, since they are attached to the element, the nodes are held to it by means of the elastic recall forces. These forces act on the nodes in ways that depend on the formulation of the element (that is, depending on the chain of simplifying hypotheses made first about the formulation and then about the element), *and these should not be confused with the forces that are applied on a part of a real structure.*

The elastic recall forces increase as the movement of the nodes increase, so that, up to a certain amount of movement, they exactly counterbalance the forces that make the nodes want to move; the nodes thus remain in equilibrium. In this situation any ulterior variation of the configuration would annul the first variation of total potential energy.

So, we can imagine the nodes of an element to be like little cubes that are solicited by forces that want to make them move, and are recalled by "springs" that tend to put them back in their places: part of these springs are owing to the finite element in question, and part to adjacent finite elements.

If there is no "spring" attached to a certain component of translation or rotation of a given node, then there are two possible cases:

1. There are forces that want to make the node move in that direction (that is, in the direction of the translation or rotation);
2. There are no forces that want to make the node move in that direction.

In the first case, equilibrium is obviously impossible, in that there are no elastic recall forces that can oppose the node's movement.

In the second case equilibrium doesn't matter, since nothing makes the node move and, at the same time, nothing opposes any movement.

In its turn, the lack of a "spring" attached to the degree of freedom in question can be due to two distinct causes:

1. No elements of any kind are attached to the node in question;
2. None of the elements attached to the given node possess a stiffness for the degree of freedom, due to the formulation of the elements themselves.

In the first case, the node is said to be *free* or *disconnected.* In the second case we say that the degree of freedom is inactive, or that the stiffness associated to is *null stiffness.*

4.4 THE PROBLEM OF CONGRUENCE AND COMPATIBILITY

The displacement field predicted by equation 4-1 does not contain gaps or overlaps since it is obtained by combining continuous functions linearly. We can thus expect that congruity is satisfied within the finite element. But what happens on the edge?

Up to this point we have looked at what happens to a certain finite element by imagining it to be extracted from its context. We have not worried about any coherences between what takes place within "our" finite element and what takes place in others, especially in adjacent ones.

This is not acceptable, because it is evident that there has to be some kind of coherence between what happens in one finite element and what happens in elements adjacent to it.

For the sake of simplicity, let's consider two adjacent finite elements of the same type, and look at what takes place at the points that lie along the shared edges. The edge in question can be a point (as is the case when the two elements in contact have linear domains, for example, beam or rod elements), a line (as happens when the domains of the elements are segments of surfaces, for example in plates or membranes), or a surface (as happens when the domains are solids, such as for tetrahedral or brick elements). The points that lie on the shared edge appertain simultaneously to both elements.

In order to avoid contradictions in our hypothesis, the displacement of the points on the shared edge, conceived as belonging to one of the finite elements, should be equal to the displacements of the points of the shared edge, conceived as belonging to the other finite element. This circumstance is called *compatibility*. Compatibility is simply a particular aspect of congruity, that is, of the requirement that there are no gaps or overlaps at the boundary between two finite elements. In the earlier definition we spoke of displacement, but we must also add all the derivatives of displacement required by the formulation. For example, if two plate elements are adjacent, then the deflection must be continuous, but so is the derivative of the displacement (that is, there must be no cusps in the deformed surface).

An element is compatible if it is paired with another that is identical with respect to congruity. If instead two identical and adjacent finite elements predict different displacements (or derivatives of the displacements), then for identical points in the domain in question it is clear that congruity will be violated, and the element is incompatible.

For example, figure 4-8 shows two adjacent triangular elements, **934** and **933**, in a problem of plane stress. Let's suppose that the shared side is the side of nodes i-j for element **934** and j-k for element **933**. If a linear displacement function is used, of the type

$$u(x, y)=a_1+a_2x+a_3y$$

for the displacement, and if λ is a parameter comprised between 0 and 1 that allows us to move along the shared side, then for element **934** we must have

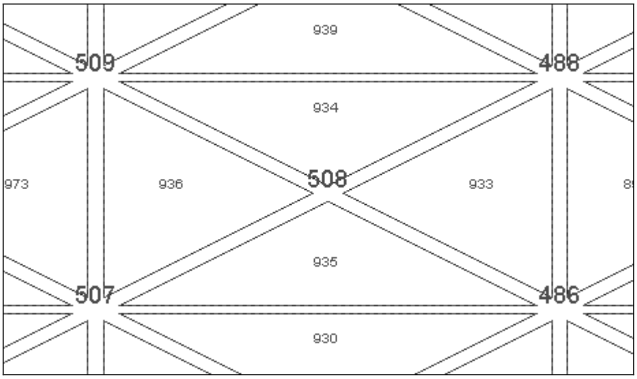


Figure 4-8: Part of the mesh of a membrane (plane stress).

$$u_{\mathbf{934}}\ (\lambda)=u_i+\lambda(u_j-u_i)$$

and for element **933**

$$u_{\mathbf{933}}\ (\lambda)=u_j+\lambda(u_k-u_j).$$

In fact, this is the only way to satisfy the linearity that was hypothesised and at the same time satisfy the equations 4-6. Substituting λ =0 and λ =1 results in the desired nodal displacements, and furthermore, the variation is linear in the abscissa λ.

But according to the table of global node numbering, we have

element **933** node j	global numbering 508
element **933** node k	global numbering 488
element **934** node i	global numbering 508
element **934** node j	global numbering 488

since the two elements are attached at the same nodes, and therefore, according to global numbering, for displacement u along the shared edge, for element **934** we have

$$u_{\mathbf{934}}(\lambda)=u_{508}+\lambda(u_{488}-u_{508})$$

and for element **933**

$$u_{\mathbf{933}}\ (\lambda)=u_{508}+\lambda(u_{488}-u_{508})$$

(the number in bold is the element number; the non-bold number is a node number). For each λ,

$$u_{\mathbf{934}}(\lambda)=u_{\mathbf{933}}\ (\lambda)$$

a relation that indicates the complete compatibility of the two elements along the shared edge. Since similar justifications can be applied to the other sides as well, the element for the stress plane whose displacement functions are linear is compatible.

The lack of compatibility is a price we must sometimes pay, given that the hypotheses about the displacement field are only approximate. In certain cases the lack of compatibility is therefore a known and accepted characteristic of the finite elements used. Obviously the lack of compatibility has significant consequences, since when the elements' dimension tends to zero, the violation of congruity does not tend to zero, and thus it may be impossible for the approximate solution to converge to the exact one. The presence of non-compatible elements also leads to the loss of important properties such as the monotonicity of the convergence, that is, the fact that successive meshes which are increasingly refined, lead to necessarily increasing degrees of precision.

In the case of non-compatible elements it is not possible to satisfy congruity completely since a part of the inter-element displacement field is discontinuous. There is no remedy for this.

However, we do find a series of errors in meshing that lead to violations of congruity even though this isn't necessary. For example, take the mesh shown in figure 4-9, which is made as before with triangular elements whose displacement field functions are linear (for clarity, the elements are *shrunken*).

It is clear that congruity is violated here even though the elements are compatible. *It is the way the mesh was constructed that prevents congruity from being respected.* In fact, along the side defined by nodes 7-9, the displacement of the points predicted by element 1 is linear, while along the same segment the displacement will be given by two linear segments when these points are thought of as appertaining to elements 2 and 3.

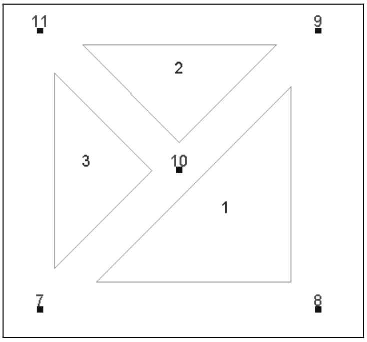


Figure 4-9: An example of a violation of congruity (for clarity, elements are shrunken).

The displacement of the points between node 7 and node 10 depends on node 7 and node 9 if the points are thought of as belonging to element 1, while it depends on node 7 and node 10 if the points are thought of as belonging to element 3. The displacements can never be identical. The same discrepancy exists for segment 10-9. The position of node 10 will not be that predicted by the shape functions of element 1, and thus there will be an overlap or gap along nodes 7-10-9.

The correct mesh is shown in figure 4-10 (the elements have been renumbered).

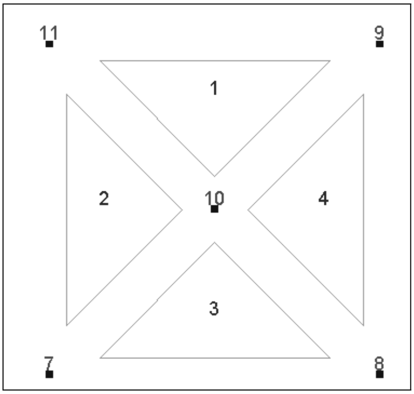


Figure 4-10: Congruity restored.

In order to treat this material systematically, we need to go into greater detail.

4.4.1 CLASSIFICATION OF CONNECTIONS

An element is said to be connected to a node when that node communicates at least a part of its displacements to the corresponding point of the element. The fact that a node is located in correspondence to a point of an element does not imply connection; in fact, the node can simply be located in that point without transferring its movements to the point itself. To be more specific, we can say that *a node is connected to an element if it is part of the connectivity table of that element.*

Let's consider the kinds of relationships that can exist between two elements.

Two elements are said to be connected if they have at least one node in common, that is, if there is at least one node that is found in the connectivity table of both elements.

Elements are divided into three categories based on their domains: one-dimensional elements (the domain is part of a line); two-dimensional elements (the domain is part of a surface); three-dimensional elements (the domain is part of a solid). The edge of a one-dimensional element is constituted of two points (the nodes at the end). The edge of a two-dimensional element is constituted of a certain number of sides, be they straight lines or curves, each defined by a certain number of nodes (in general there are two nodes per straight line sides and three nodes for curved sides). The edge of a solid element is constituted of a certain number of faces (planes or curves). The edge of each face is defined by a certain number of sides, be they straight lines or curves. The faces usually have three nodes (triangles), four nodes (quadrilaterals), six nodes (curved triangles), or eight nodes (curved quadrilaterals).

Let's now look at the relationships that exist between finite elements of the same type.

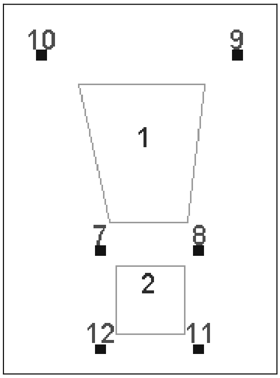


Figure 4-11: Connected sides. Side N1-N2 of the first element is connected to side N2-N3 of the second element (cfr. table 4-2).

Element	N1	N2	N3	N4
1	7	8	9	10
2	11	8	7	12

Table 4-2: Elements connected on a side.

A side of a two- or three-dimensional element is said to be connected with the side of another two- or three-dimensional element (or *connected on a side*) *if all and only the nodes of the side of the first element are the same nodes of the side of the second element*. If this is true, and if the element is compatible, then the displacements (and possibly the derivatives of the displacements) of the point on the side predicted by one of the two finite elements will be identical to those predicted by the other one.

If only one node of the edge is common to two two-dimensional elements, without any geometric superimposition of the sides connected at the node, then the elements are said to be have a *cuspidal connection*. A cuspidal connection between two identical finite elements indicates the identity of the displacements of the common point.

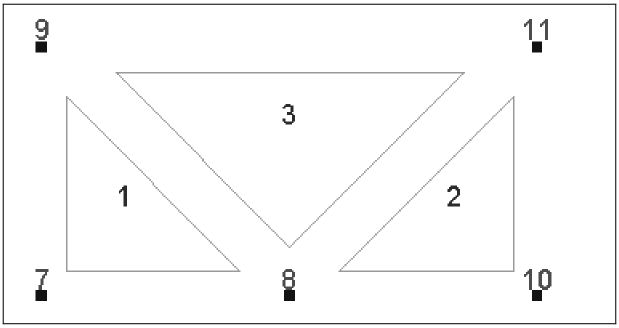


Figure 4-12: Elements 1 and 2 are cuspally connected, 1 and 3 are connected on a side, and 2 and 3 and connected on a side (cfr. table 4-3).

Element	N1	N2	N3	N4
1	7	8	9	-
2	8	10	11	-
3	8	11	9	-

Table 4-3: Node 8 is common to element 1 and element 2. Side N2-N3 of element 1 is connected to side N3-N1 of element 3. Side N3-N1 of element 2 is connected to side N1-N2 of element 3.

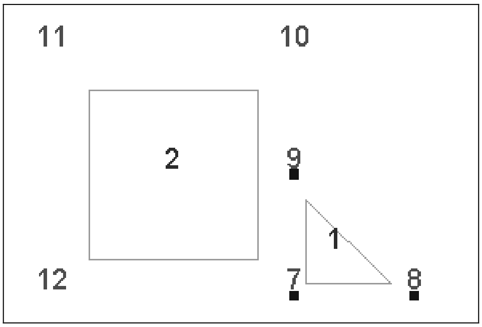


Figure 4-13: Partially connected elements: node 7 is common to side 7-10 and side 7-9, but one node is not sufficient (cfr. the incidence table, table 4-4).

Element	N1	N2	N3	N4
1	7	8	9	-
2	12	7	10	11

Table 4-4: Incidence table for the structure in fig. 4-13.

If only part of the nodes are identical, and the two sides are geometrically superimposed, then the sides are said to be *partially connected.* Two sides with a different number of nodes can only be partially connected. For example, a side with two nodes cannot be connected to a side with three nodes.

If two sides do not have any nodes in common, they are said to be not connected, or *disconnected.*

When two sides are (totally or partially) geometrically coincident but do not have any nodes in common, the two elements are simply said to be *facing.* Elements that are simply facing are not connected, thus there is no structural continuity between them.

For example, figure 4-14 shows two elements with the following incidence table:

Element	N1	N2	N3	N4
1	7	8	9	10
2	11	14	13	12

Table 4-5: Facing elements (incidence table for the structure in fig. 4-14).

Even though nodes 11 and 12 lie on the side defined by nodes 7 and 8, nodes 11 and 12 do not appear in the connectivity table for element 1, and thus they are not connected to this element. Consequently, elements 1 and 2 are not connected, even though they face each other. This could immediately be seen from the mesh in the figure on the left.

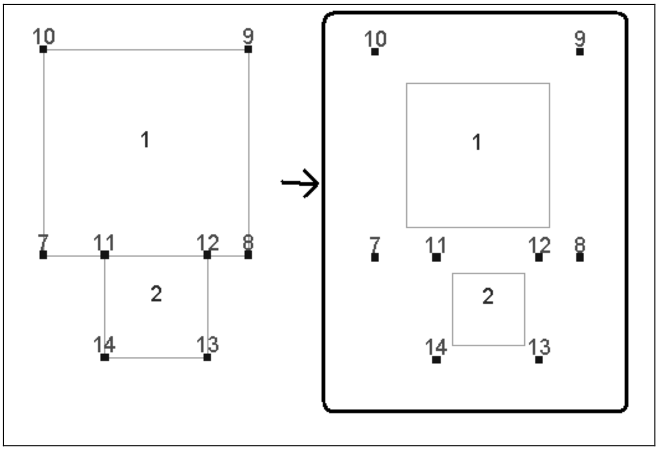


Figure 4-14: Facing elements shown without shrink (at left) and with shrink (at right).

Figure 4-15 again shows two facing elements (which have the same connectivity table as before, shown in table 4-5), but here the figures do not make it immediately apparent that the mesh is not well made: the two elements appear to be connected, but in order to see the double nodes that effectively make them disconnected, we have to see the node numbering (which shows that the numbers are superimposed).

The concepts we have seen for two-dimensional elements can be generalised to the cases of three-dimensional elements as well. A face of a three-dimensional element is said to be connected to the face of another three-dimensional element of the same type if all the nodes of the face of the first element are all and only the nodes of the face of the second element. If the faces have a node in common but they are not geometrically coincident, they are said to be cuspidally connected. If two faces don’t have any nodes in common they are said to be disconnected. Two faces with different numbers of nodes can only be partially connected.

If two elements are partially connected but have a face or a side in common geometrically, they generally violate congruity. Thus partial connections should be avoided, unless we want to simulate cracking, construction joints, or more generally, situations which present discontinuities.

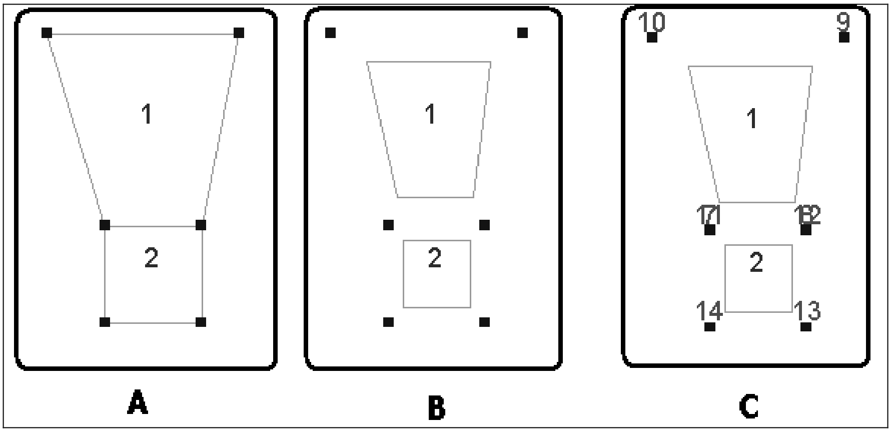


Figure 4-15: Facing elements, double nodes. A) without shrink; B) with shrink; C) with nodes numbered.

Sometimes it is not at all easy to recognise a violation of congruity within an existing mesh. Figure 4-16 shows elements that are partially connected: we can thus expect to find violations of congruity and the mesh must be corrected.

To summarise, we can subdivide the connections as follows:

Disconnected Elements	**Connected Elements**
Disconnections	Cuspidal connections
Facing, without double nodes	Partial connections
Facing, with double nodes	Effective connections

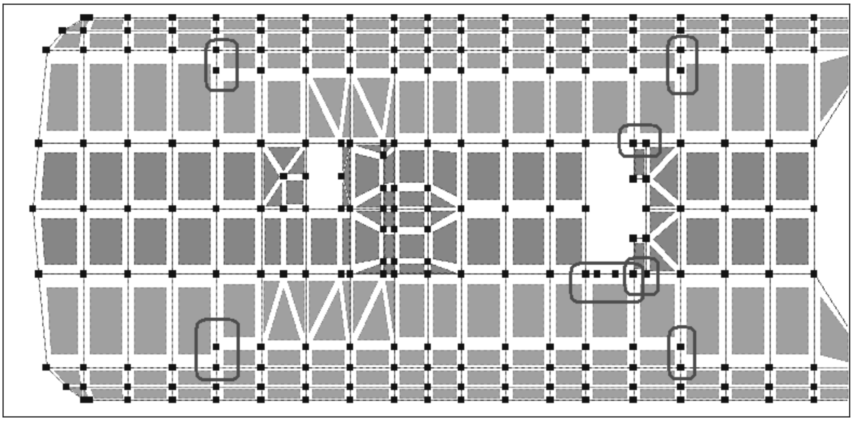


Figure 4-16: An example of a mesh with congruities violated (partially connected elements). The plate elements are shown with shrink, the beam elements are shown with dashed lines.

4.5 Local and global systems

Up to this point we have implicitly assumed that the element was oriented such that its own intrinsic reference systems coincided with the global reference system of the structure as a whole. By *intrinsic reference systems* we mean the reference system used to describe everything that takes place within the element: in the intrinsic reference system for a rod element, the first of three axes of reference coincides with the axis of the rod and the other two axes are irrelevant; in the intrinsic reference system for a beam element, the first axis coincides with the axis of the beam, and the other two axes coincide with the principal axes of the section; in the intrinsic reference system for a plate or membrane, plane (x, y) coincides with the plane of the element, and so forth.

In practice this condition of coincidence is very seldom if ever present. The elements can have different orientations with respect to each other and with respect to the global reference system adopted for the structure as a whole (see figure 4-17).

Because of this, the *local* description of what takes place within a finite element made when it is observed from a point of view coincident with the element's own intrinsic reference system is not the same as when it is observed from a point of view that coincides with the reference system of the structure as a whole, that is, the *global* reference. The "*x*-axis" or "axis 1" of a finite element might be a directrix that is *strange* from the global point of view, and vice versa, the global *x*-axis might be a directrix that is *strange* from the point of view of the finite element in question.

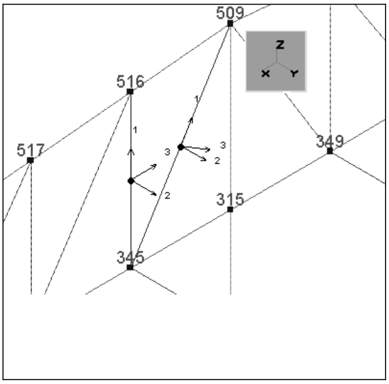


Figure 4-17: Local reference systems of two finite elements (indicated by axes 1, 2 and 3) and the global reference system. The local reference systems of the other elements are not shown.

Of course, we don't want to lose the advantages that come with using the reference system that is most convenient for the case at hand, since this means that the data as well as the results can be presented in a much simpler way.

However, we have to understand, in general terms, how we go from a description in one system to a description in another, and vice versa.

The point of departure is constituted of relations that allow us to go from an (x,y,z) coordinate system, called **l**, to an (X, Y, Z) coordinate system, called **g**. Let's suppose that (x,y,z)=**l** is the local coordinate system and (X, Y, Z)=**g** is the global coordinate system. It results that

$$\mathbf{g} = \mathbf{Tl}$$

where **T** is a matrix that is *square and orthogonal.*[9] Matrix **T** is simply created with the cosine directors of the versors of the three local axes. This can be made clear by an example.

[9] Recall that an orthogonal matrix is such that its inverse coincides with its transpose: $\mathbf{TT}^{\mathrm{T}}=\mathbf{I}=\mathbf{T}^{\mathrm{T}}\mathbf{T}$.

Let's suppose that the three local axes are directed as follows with respect to the global axes (see figure 4-18):

local x-axis	(0, 1, 1)
local y-axis	(0, -1, 1)
local z-axis	(1, 0, 0).

First we find the versors of the three local axes:

local x-axis	(0, 1/√2, 1/√2)
local y-axis	(0, -1/√2, 1/√2)
local z-axis	(1, 0, 0).

Next we form square matrix **T** with the cosine directors we have found, ordered by rows

$$\mathbf{T}=\begin{vmatrix} 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 1 & 0 & 0 \end{vmatrix}.$$

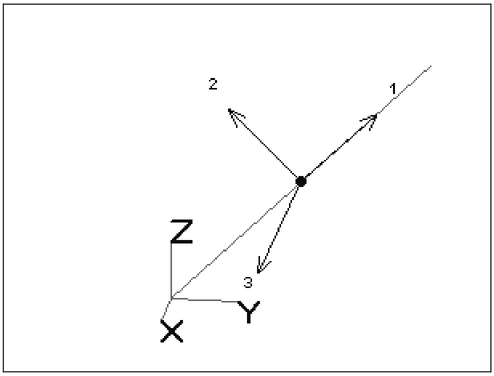


Figure 4-18: An example of going from local to global reference systems, and vice versa. The local axes (x, y, z) are indicated respectively by the numbers "1", "2", "3".

The following direct relations hold:

$$\begin{vmatrix} x \\ y \\ z \end{vmatrix} = \begin{vmatrix} 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 1 & 0 & 0 \end{vmatrix} \begin{vmatrix} X \\ Y \\ Z \end{vmatrix},$$

which can be written in concise form as

$$\mathbf{l} = \mathbf{Tg}.$$

The inverse relations also hold

$$\begin{vmatrix} X \\ Y \\ Z \end{vmatrix} = \begin{vmatrix} 0 & 0 & 1 \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \end{vmatrix} \begin{vmatrix} x \\ y \\ z \end{vmatrix}$$

and can be written in concise form as

$$\mathbf{g}=\mathbf{T}^{\mathrm{T}}\mathbf{l}.$$

Let us now look at the nodal displacements, that is, the displacements of a given node *i* of a certain element. These displacements $\mathbf{u}_{\mathrm{i}}$ are defined in the local reference system, while the displacements $\mathbf{u}_{\mathrm{ig}}$ are defined in the global reference system: we have added the subscript "g" to indicate "with respect to the global reference system". The number of nodal displacements varies according to the formulation (two for planes, three for solids, six for beams, etc.) and according to displacement type (translations or rotations). *In each case we can write a suitable square and orthogonal matrix* **T*** *that makes it possible to go from the global reference system to the local one according to the matrix formula*

$$\mathbf{u}_{\mathrm{i}}=\mathbf{T^*u}_{\mathrm{ig}}.$$

If we now look at the overall nodal displacements of the element, and we form the vector $\mathbf{u}_{\mathbf{e}}=\{\mathbf{u}_{1}, \mathbf{u}_{2}, \ldots., \mathbf{u}_{\mathrm{N}}\}^{\mathrm{T}}$, with N the number of nodes of the element, we can also form the larger matrix **T** such that

$$\mathbf{u}_{\mathbf{e}}=\mathbf{Tu}_{\mathbf{eg}}. \qquad [4\text{-}19]$$

To do this we need only set

$$\mathbf{T}=\begin{vmatrix} \mathbf{T^*} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \mathbf{T^*} & \ldots & \mathbf{0} \\ \ldots & \ldots & \ldots & \ldots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{T^*} \end{vmatrix}.$$

We can now use equation 4-19 to update what we had found in the previous section. Thus we no longer express all of the formulas as a function of nodal displacement *in reference to the local coordinate system*, but as a function of the nodal displacements *in reference to the global coordinates*. This is an indispensable step towards ultimately combining all of the data relating to the various finite elements.

First of all, we can see that since each element has its own collocation, which is different from all other elements, each thus has its own transformation matrix: thus we have to write "$\mathbf{T}_e$" rather than "$\mathbf{T}$", that is, we have to introduce subscript "e" into the transformation matrix as well.

The relationship between global nodal displacements $\mathbf{u}_{\mathbf{eg}}$ and local nodal displacements $\mathbf{u}_{\mathbf{e}}$ is:

$$\mathbf{u}_{\mathbf{e}}=\mathbf{T}_{\mathbf{e}}\,\mathbf{u}_{\mathbf{eg}}. \qquad [4\text{-}20]$$

Now let's *define* the following new quantities, which represent the extension to the global reference system of what we have seen earlier for the local reference system:

Stiffness matrix $\mathbf{K}_{eg}$ of the global system

$$\mathbf{K}_{eg} \equiv \mathbf{T}_{\mathbf{e}}^{T}\,\mathbf{K}_{e}\,\mathbf{T}_{\mathbf{e}}. \qquad [4\text{-}21]$$

Equivalent nodal forces $\mathbf{f}_{\mathbf{eg}}$ in the global reference system

$$\mathbf{f}_{\mathbf{eg}} \equiv \mathbf{T}_{\mathbf{e}}^{T}\mathbf{f}_{\mathbf{e}}. \qquad [4\text{-}22]$$

Equivalent nodal reactions $\mathbf{r}_{\mathbf{eg}}$ in the global reference system

$$\mathbf{r}_{\mathbf{eg}} \equiv \mathbf{T}_{\mathbf{e}}^{T}\mathbf{r}_{\mathbf{e}}. \qquad [4\text{-}23]$$

With these definitions the results that we found in section 4-3 can easily be expressed in the following way (we will omit the steps, but these can be reconstructed easily enough from the definitions just introduced and substituting for each instance of $\mathbf{u}_{\mathbf{e}}$ its equivalent expression $\mathbf{T}_{\mathbf{e}}\mathbf{u}_{\mathbf{eg}}$, and for each instance of $\mathbf{u}_{\mathbf{e}}^{T}$ its equivalent expression $\mathbf{u}_{\mathbf{eg}}^{T}\mathbf{T}_{e}^{T}$):

Strain energy (see equation 4-15)

$$Z = 0.5\,\mathbf{u}_{\mathbf{eg}}^{T}\,\mathbf{K}_{eg}\,\mathbf{u}_{\mathbf{eg}}. \qquad [4\text{-}24]$$

Work performed by the applied loads

$$W_p = \mathbf{u}_{\mathbf{eg}}^{\;T}\,\mathbf{f}_{\mathbf{eg}}. \qquad [4\text{-}25]$$

Work performed by the reaction forces

$$W_r = \mathbf{u}_{\mathbf{eg}}^{T}\mathbf{r}_{\mathbf{eg}}. \qquad [4\text{-}26]$$

Finally, the element's equilibrium equation (see 4-17)

$$\mathbf{r}_{\mathbf{eg}}=\mathbf{K}_{eg}\,\mathbf{u}_{\mathbf{eg}} - \mathbf{f}_{\mathbf{eg}}. \qquad [4\text{-}27]$$

In essence, we have expressed already known results in function of quantities that refer to the global rather than to the local reference system, and in particular, *in function of the nodal displacements referred to the global rather than to the local reference system*. We are thus taking steps towards describing in a unified way what takes place in the various finite elements. In order to do this we need to describe everything in accordance with the same reference system. We are dealing with equations that may appear difficult, but actually, if we keep in mind that the subscripts "eg" simply mean "of element 'e' with respect to the global reference system 'g'", the equations are much simpler to understand.

4.6 CALCULATING THE INTEGRALS

4.6.1 INTRODUCTION

We have seen that the situation of a finite element can be described by means of a set of simplifications that lead us to write the equilibrium in discrete form. In order to write the equilibrium equations materially we have to be able to evaluate both the stiffness matrix and the equivalent nodal force vectors.

Let's look again at the definition of the stiffness matrix of an element, in relation to its local reference system

$$\mathbf{K}_e \equiv \int \mathbf{B}^T \, \mathbf{D} \, \mathbf{B} \, dV. \qquad [4\text{-}28]$$

This matrix is composed of a certain number of integrals. Matrix **D** is composed of material or geometric constants, and thus its calculation does not present any particular difficulties. On the other hand, matrix **B** is composed of functions, and since the product $\mathbf{B}^T\mathbf{D}\mathbf{B}$ appears in the matrix, these functions have to be multiplied by each other, giving rise in their turn to functions that are much more complex (as well as being of a higher order).

Two distinct problems arise in evaluating these integrals numerically: the first is that the integral is generally taken over a domain (that of the element) whose form can be complex, a problem related to the element's geometry; the second is that it is not always possible to integrate in closed form the functions that result from the product $\mathbf{B}^T\mathbf{D}\mathbf{B}$, and thus there arises a genuine problem of integration.

4.6.2 PROBLEMS OF GEOMETRY

To overcome the problems related to the complexity of the finite element's shape (which, remember, can also have curved sides), a computation trick is often used that makes use of the concept of *transformation*. Without going into the formulas in great detail (which is beyond the scope of this book, although some examples will be seen in chapter 11), we can say that to the real domain of the element there is associated a fictitious domain of regular shape, defined in a space in which the coordinates assume non-dimensional values that are generally comprised between –1 and 1.

The fictitious domain has a very simple shape. For example, if the original domain was a quadrilateral with sides whose slopes are all different, the fictitious domain is a square whose centre is at the origin, with sides that go from –1 to 1. In the literature, the coordinate system of the fictitious element is often indicated by the tern (ξ,η,ζ). In essence, the actual finite element with its nodes is transformed into a fictitious finite element with an equal number of nodes but whose shape is regular.

In substance, we use *transformation formulas* such as

$$x=x(\xi,\eta,\zeta)$$
$$y=y(\xi,\eta,\zeta)$$
$$z=z(\xi,\eta,\zeta).$$

These formulas tell us to which point P(x, y, z) of the real finite element corresponds a certain point $P(\xi,\eta,\zeta)$ of the fictitious finite element. They also model the finite element's edge as a function of the non-dimensional coordinates (ξ,η,ζ) .

For example, if we are dealing with two-dimensional elements, the equation

$$\xi=-1$$

corresponds to a side of the fictitious element, while the curve

$$x=x(-1,\eta)$$
$$y=y(-1,\eta)$$
$$z=z(-1,\eta)$$

corresponds to the respective side of the real finite element.

As the values assumed by η (comprised between –1 and 1) vary, there corresponds a series of points P on the real element which describe one of its "strangely shaped" sides.

Let $\mathbf{x}$ be vector $\{x, y, z\}^T$. If $\mathbf{x}_e$ is the vector obtained with the coordinates of the element's nodes (in a three-dimensional problem $\mathbf{x}_e = \{x_1, y_1, z_1, x_2, y_2, z_2, \ldots x_n, y_n, z_n\}^T$) and $\mathbf{\Phi}$ is the matrix of the shape functions used to model the displacements, the equations that relate the (ξ,η,ζ)coordinates to the (x, y, z) coordinates can be written

$$\mathbf{x}= \mathbf{\Phi}\, \mathbf{x}_e.$$

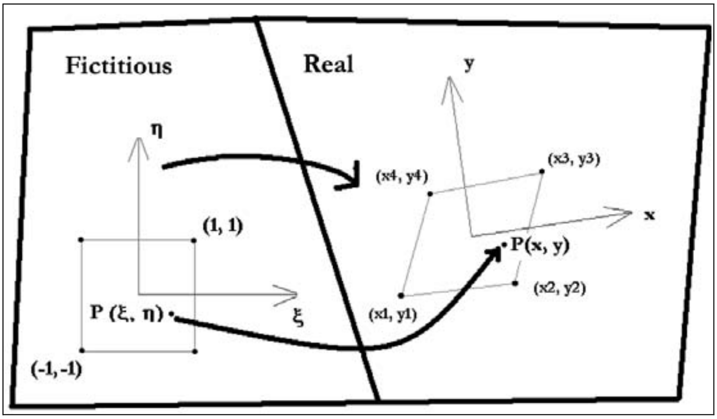


Figure 4-19: Correspondences between the fictitious finite element and the real one.

This is a particular choice for the transformation. The shape functions **Φ** are now defined in the space (ξ,η,ζ), that is, **Φ**=**Φ**(ξ,η,ζ), and from this it turns out that *to model the shape of the element's edge the same functions have been used that were used to model the unknown displacements*, an important circumstance. In fact, in equation 4-8 we had written

$$\mathbf{U} = \mathbf{\Phi u}_e.$$

The elements created from this particular assumption are called *isoparametric elements*, and they constitute an important and widely used family.

Thanks to these formulas, with a few mathematical steps we can see that the integral that must be calculated in domain (x,y,z) with dV as an infinitesimal portion of the domain, is equal to the integral taken in domain (ξ,η,ζ) with dV* as an infinitesimal portion of the domain, as long as JdV* is substituted in place of dV, where J is the *Jacobian of transformation.* In short,

$$\mathbf{K}_e = \int \mathbf{B}(x,y,z)^T \, \mathbf{D} \, \mathbf{B}(x,y,z) \, dV = \int \mathbf{B}(\xi,\eta,\zeta)^T \, \mathbf{D} \, \mathbf{B}(\xi,\eta,\zeta) \, J dV^*. \qquad [4\text{-}29]$$

If the domain is a surface in the plane (ξ,η) we will have

$$dV^* = d\xi d\eta.$$

If instead the domain is a volume, we will have

$$dV^* = d\xi d\eta \, d\zeta.$$

Formula 4-29 tells us that rather than calculate the integral over a "strangely shaped" domain (for example, one whose edges are curves) we can calculate the integration for a domain whose shape is simple (with straight line sides), as long as the formula is modified as shown.

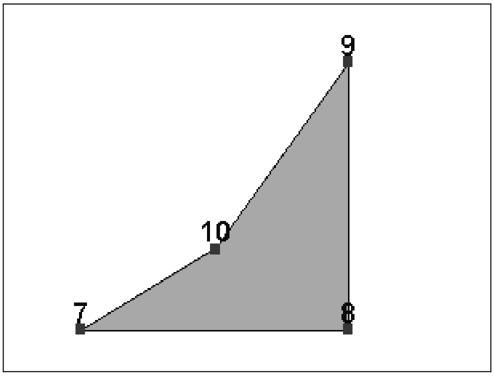


Figure 4-20: A distorted quadrilateral element: the transformation is not univocal, and the element is unacceptable.

The Jacobian of the transformation is a complicated function of the fictitious coordinates (ξ,η,ζ). It is also an indicator of the validity of the transformation. It is easy to see that if the real element is too distorted or folded onto itself, then *the transformation is not biunique*, that is, for a point in the fictitious domain there is more than one corresponding point in the real domain; it is also possible that for a point in the fictitious domain there is no corresponding point in the real domain.

We therefore have to create elements that have interior angles no greater than 180° and, if possible, with interior angles that are not too different from each other.

4.6.3 Problems of integration

Even though the domain of integration has been simplified by a suitable transformation, the fact remains that the integral to be evaluated is generally too complicated to be calculated in closed form.

To overcome this difficulty, a numeric integration is used, that is, rather than evaluate the integrals in closed form, the integral is usually calculated in a simplified way, making use of techniques of integration that do not require finding the primitive of the function to be integrated.

The technique known as *Gaussian integration* is particularly important and widely used. In Gaussian integration the integral

$$\int f(\xi) d\,\xi$$

(presumed to be only of variable ξ) is evaluated by means of a weighted sum, that is, by setting

$$\int f(\xi)\, d\,\xi = w_1 f(\xi_1)+w_2 f(\xi_2)+w_3 f(\xi_3)+\ldots. w_n f(\xi_n).$$

Function f is evaluated at a certain number of points (called Gauss points), and to each point is associated a given weight, an appropriate number w. Transforming an integral into a sum is very convenient, but does it work?

In effect we can see that the precision of a Gaussian integration depends on the number of Gauss points used to evaluate the integral: as the number of Gauss points increases, so does the precision with which the integral is evaluated.

Note that the position of the Gauss points does not remain the same as the number of points is varied, nor when the weights change. In table 4-6 we see the first Gauss weights and the first abscissas, presuming that the domain of integration varies between –1 and 1.

In order to understand how it works, let's take a simple integral such as

$$\int x^4 dx.$$

The exact result is (1/5 -(-1/5))= 2/5= 0.4.

If we adopt the Gauss formula using various numbers of points, we obtain the following results:

1 point	0
2 points	$(0.5773502692)^4+(-0.5773502692)^4= 0.22222\ 22222$
3 points	$2x0.5555555556(0.77459\ 66692)^4+0=0.3999999999$
4 points	$2x0.347...\ (0.8611363116)^4+2x0.652\ldots(0.3399810436)^4=0.399\underline{9}$.

As you can see, from three points and up the formula gives rise to an "exact" result (the weights and the non-dimensional abscissas are written to the tenth significant digit).

It can be proven that the formulas for Gaussian integration with *n* points can exactly integrate polynomials of order (2n-1). Thus, a polynomial of order m=4 requires at least three Gauss points (2x3-1=5 > 4).

The use of Gaussian integrations significantly simplifies the calculation, because a complex integral is transformed into a simple weighted sum.

The formula for Gaussian integration is also valid when integrating functions of several variables: in this case the sum is not simple but double, in that it is necessary to integrate in one or more directions.

Thus the integral

$$\int \mathbf{B}(\xi,\eta,\zeta)^{T}\ \mathbf{D}\ \mathbf{B}(\xi,\eta,\zeta)\ \mathrm{J}dV^{*}$$

which gives the stiffness matrix is transformed into a weighted sum of the values of

$$\mathbf{B}(\xi,\eta,\zeta)^{T}\ \mathbf{D}\ \mathbf{B}(\xi,\eta,\zeta)\ \mathrm{J}$$

of a suitable number of points in the domain of the fictitious element, called Gauss points.

Number of points	**Position**	**Weights**
1	0	2
2	±0.5773502692	1
3	±0.7745966692 0	0.5555555556 0.8888888889
4	±0.8611363116 ±0.3399810436	0.3478548451 0.6521451549

Table 4-6: Non-dimensional abscissas and weights for Gauss points (integrated from –1 to 1).

How to choose the correct number of Gauss points to be used in numerically evaluating the stiffness matrix has been the subject of many studies. While it is true that as the number of Gauss points increases, so does the precision, it is also true that the computational burden of evaluating the functions by integrating in several points also grows correspondingly. Further, since it can be demonstrated that the FEM tends to overestimate stiffness, some authors have proposed the deliberate use of techniques of *reduced integration* (that is, integration with fewer Gauss points than necessary) in order to come closer to the correct values for stiffness.

Regarding this, we quote K. J. Bathe:

> “We recommend that *full numerical integration* always be used for a displacement-based or mixed finite element formulation …. The reason for recommending [full] numerical integration ... is that the reliability of the finite element procedures is of utmost concern, and if an integration order lower than the “full” order is used [for a displacement-based formulation], the analysis is in general unreliable” [1996: 469].

This will be discussed further in chapter 11.

5 ASSEMBLY

5.1 ARRIVING AT A COHERENT DESCRIPTION

Up to this point we have described what takes place within a finite element without worrying about the relationship between what takes place in the individual element and what takes place in the structure as a whole. It is evident that we can't really do this, since what happens in one element is largely determined by what happens in the others, and we cannot understand the structure unless we reconstruct the overall situation.

The forces acting on a finite element are in part communicated to adjacent elements. These forces are determined by the resistance that each element makes to its distortion, and this distortion depends on the quantity of relative displacement to which its nodes are subjected.

The search for the values of displacement that simultaneously establish equilibrium among the finite elements and the set of applied loads can only be undertaken in a global way, that is, by considering the set of all possible displacements in all nodes of the structure.

There are at least two different ways to arrive at the same set of final relations. One is the *static method*, in which equilibrium equations are directly written in discrete form (that is, not everywhere, but only in some points). The other is the *energy method*, in which the total potential energy is written as the sum of the potential energies of the various elements that make up the structure.

Before describing these two methods, we first need to clarify how we go from the element's nodal displacement vector to the nodal displacement vector of the structure as a whole.

5.2 NODAL DISPLACEMENT VECTORS: FROM ELEMENT TO STRUCTURE (AND VICE VERSA)

Let's consider the vector of nodal displacements of the nodes of a given element "e" with reference to the global system "g", that is, $\mathbf{u}_{eg}$. This vector is obtained by ordering the nodes of the element from first to last – i, j, k ... (or N1, N2, N3...) – and then forming a vector constructed of as many blocks as there are nodes. That is,

$$\mathbf{u}_{\mathbf{eg}}=\{\mathbf{u}_{ig}, \mathbf{u}_{jg}, \ldots, \mathbf{u}_{Ng}\}^{T}.$$

Each block is a small vector that groups the displacements of a node; the number of these displacements depends on the type of formulation and on the finite element in question (for example, there are 6 displacements for a beam element in space, 2 for an ordinary element of plane stress, 3 for a plate element without a membrane component, and so forth).

To be precise, we will have

$$\mathbf{u}_{\mathbf{eg}} = \left|\begin{array}{c} \mathbf{u}_{ig} \\ \hdashline \mathbf{u}_{jg} \\ \hdashline \ldots \\ \hdashline \mathbf{u}_{Ng} \end{array}\right|.$$

Nodes i, j, k, (or N1, N2, N3,…) of the element have a certain global numeration that is different from the local one, and the correspondence between the two numerations, as we have seen, is given in the connectivity table. Node “j” (or “N2”, if you prefer) will be, for example, node 345 in global numeration. The nodal displacement block 2 in the nodal displacement vector of that element will be thus block 345 (that is, the three hundred and forty-fifth block) in the vector of *global nodal displacement*. Each individual block of the nodal displacement vector *of the element* corresponds to a precisely determined position in the nodal displacement vector *of the structure*.

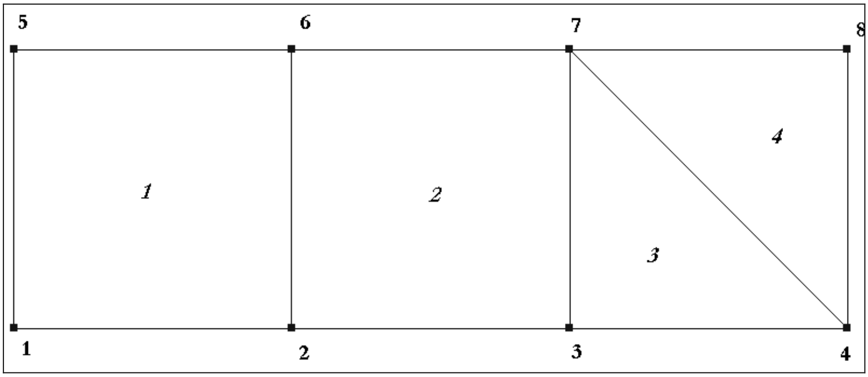


Figure 5-1: A simple structure made up of 8 nodes and 4 elements.

Let’s consider, for example, the simple structure described in fig. 5-1. Fig. 5-2 shows the correspondences between the blocks of the nodal displacement vectors of element 1 and element 2 (which are both elements with 4 nodes). Fig. 5-3 shows the analogous correspondences for elements 3 and 4 (which are elements of 3 nodes).

We can say that there exists a system that acts to transform the nodal displacement vector of element “e”, $\mathbf{u}_{\mathbf{eg}}$ (with reference to the global reference system), into the nodal displacement vector of the entire structure, $\mathbf{u}$. *In practice*, this system corresponds to a redistribution of the individual blocks that make up $\mathbf{u}_{\mathbf{eg}}$. *Formally*, the same result can be obtained by multiplying vector $\mathbf{u}_{\mathbf{eg}}$ by an appropriate rectangular matrix, $\mathbf{A}_e$.

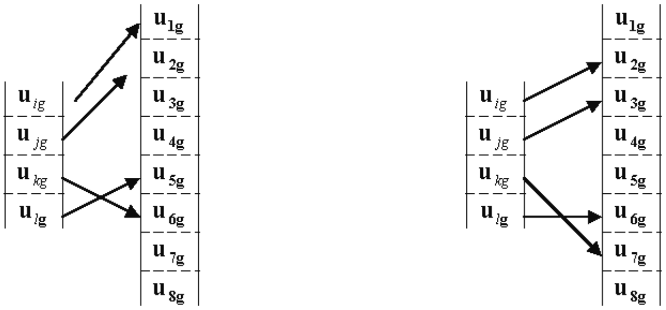


Figure 5-2: Element 1 and element 2.

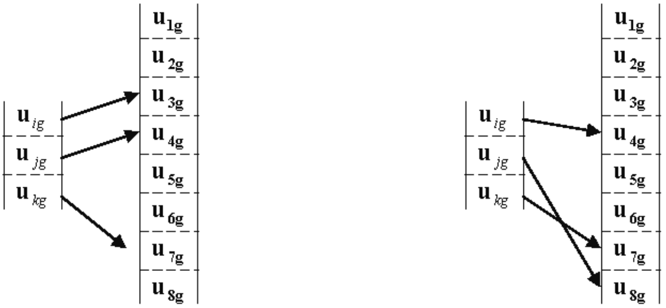


Figure 5-3: Element 3 and element 4.

In fact, it is possible to find matrix $\mathbf{A}_{\mathrm{e}}$ (the subscript "e" indicates that the matrix varies from element to element) such that

$$\mathbf{u}=\mathbf{A}_{\mathrm{e}}\,\mathbf{u}_{\mathbf{eg}}. \qquad [5\text{-}1]$$

For example, for element 1 of the structure shown in fig. 5-1, we can write,

$$\begin{vmatrix} \mathbf{u}_{1g} \\ \mathbf{u}_{2g} \\ \mathbf{u}_{3g} \\ \mathbf{u}_{4g} \\ \mathbf{u}_{5g} \\ \mathbf{u}_{6g} \\ \mathbf{u}_{7g} \\ \mathbf{u}_{8g} \end{vmatrix} = \begin{vmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{vmatrix} \begin{vmatrix} \mathbf{u}_{ig} \\ \mathbf{u}_{jg} \\ \mathbf{u}_{kg} \\ \mathbf{u}_{\mathrm{lg}} \end{vmatrix}$$

where **I** is the unit matrix on the individual block.

Formally, equation 5-1 tells us how to obtain the global nodal displacement vector starting from the nodal displacement vector of the element, and tells us what global nodal displacements correspond to given local nodal displacements. It tells us that *the displacements of the nodes of the elements are not arbitrary, but are instead quite precise displacements of the nodes of the structure to which the elements belong.*

Matrix $\mathbf{A}_e$ applied to a vector that is meaningful in the context of "e", *transforms that vector into the corresponding vector that is meaningful in the context of the overall structure.* Naturally, just as it is possible to define a matrix that allows us to go from the element to the structure, it is likewise possible to define a matrix that allows us to go from the structure to the element. This is a rectangular $\mathbf{D}_e$ such that

$$\mathbf{u}_{eg}=\mathbf{D}_e\,\mathbf{u}. \qquad [5\text{-}2]$$

For example, for element 1 shown in fig. 5-1, this matrix is the result of the following relation

$$\begin{vmatrix} \mathbf{u}_{ig} \\ \mathbf{u}_{jg} \\ \mathbf{u}_{kg} \\ \mathbf{u}_{lg} \end{vmatrix} = \begin{vmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{vmatrix} \begin{vmatrix} \mathbf{u}_{1g} \\ \mathbf{u}_{2g} \\ \mathbf{u}_{3g} \\ \mathbf{u}_{4g} \\ \mathbf{u}_{5g} \\ \mathbf{u}_{6g} \\ \mathbf{u}_{7g} \\ \mathbf{u}_{8g} \end{vmatrix}$$

It is easy to see that there must be

$$\mathbf{D}_e\mathbf{A}_e = \mathbf{I}$$

$$\mathbf{A}_e\mathbf{D}_e = \mathbf{I}.$$

We need only substitute equation 5-1 in equation 5-2, or vice versa. In the first two of these equations, $\mathbf{I}$ is of the same order as $\mathbf{u}_{eg}$; in the second, $\mathbf{I}$ is of the same order as $\mathbf{u}$. Additionally, taking into account that there necessarily has to be,

$$\mathbf{u}_{eg}^{\mathbf{T}}\,\mathbf{u}_{eg} = \mathbf{u}^{\mathbf{T}}\,\mathbf{u},$$

(given that operations 5-1 and 5-2 imply only a repositioning of the same numbers, leaving the quantities unchanged), it must also be that

$$\mathbf{D}_e=\mathbf{A}_e^{\mathbf{T}}$$

$$[5\text{-}3]$$

$$\mathbf{A}_e=\mathbf{D}_e^{\mathbf{T}}.$$

5.3 STATIC METHOD

In examining the situation of an individual finite element we have arrived at a matrix equilibrium equation that summarises the various forces in play. *On every node* of an individual finite element there are reactions exerted by adjacent elements connected to that node (vector $\mathbf{r}_{\mathbf{eg}}$), the elastic actions exerted by the finite element in question due to its various distortions (vector $\mathbf{K}_{eg}\ \mathbf{u}_{\mathbf{eg}}$), and the actions due to external loads (vector $\mathbf{f}_{\mathbf{eg}}$). It must result that

$$\mathbf{r}_{\mathbf{eg}}=\mathbf{K}_{eg}\ \mathbf{u}_{\mathbf{eg}}-\mathbf{f}_{\mathbf{eg}} \qquad [5\text{-}4]$$

(see equations 4-17, 4-27).

Looking at this equation, we can see that each of these vectors is expressed using the nodes of the element, ordered in a way that is coherent with the numbering of the element itself. If we want to obtain the same relation *exploding* these vectors in order to give them their dimension as global vectors, that is, *if we want to see what takes place in this element using global vectors instead of local vectors*, then we have to multiply by the rectangular matrix $\mathbf{A}_e$ of the element in question, as indicated by equation 5-1.

$$\mathbf{A}_{\mathbf{e}}\ \mathbf{r}_{\mathbf{eg}}=\mathbf{A}_{\mathbf{e}}\ \mathbf{K}_{\mathbf{eg}}\ \mathbf{u}_{\mathbf{eg}}-\ \mathbf{A}_{\mathbf{e}}\ \mathbf{f}_{\mathbf{eg}}.$$

If we now substitute the nodal displacement vector of the element with its expression 5-2, *in place of the nodal displacement vector of the element* $\mathbf{u}_{\mathbf{eg}}$ *we find the nodal displacement vector of the structure* $\mathbf{u}$:

$$\mathbf{A}_e\ \mathbf{r}_{\mathbf{eg}}=\mathbf{A}_e\ \mathbf{K}_{eg}\ \mathbf{D}_e\ \mathbf{u}\ -\mathbf{A}_e\ \mathbf{f}_{\mathbf{eg}}. \qquad [5\text{-}5]$$

Equilibrium equation 5-5 doesn't tell us anything substantially different from what equation 5-4 told us, it only *says it in a different way*. Rather than writing the equilibrium of the element without regard to the rest of the structure, equation 5-5 writes the equilibrium of the element *in function of vectors that have the dimension of the vectors of the structure, not the dimensions of the vector of the element.* The nodal displacement vector $\mathbf{u}$ is the same for all of the elements of the structure, and no longer changes from element to element as did $\mathbf{u}_{\mathbf{eg}}$.

At this point we can impose a new condition. If it is true that $\mathbf{A}_e\ \mathbf{r}_{\mathbf{eg}}$ is the vector that represents the reaction forces exerted on the nodes of element "e" by adjacent elements, *then it must be true, in obedience to the principle of action and reaction, that the sum of all these reactions internal to the structure, as the elements vary, has to be null.* If we take the sum with respect to all elements "e", we can write this condition thus

$$\sum \mathbf{A}_e\ \mathbf{r}_{\mathbf{eg}}=\mathbf{0}. \qquad [5\text{-}6]$$

In effect, this condition says something new, because it takes into consideration the fact that all the finite elements are part of a single structure, and thus that the sum of all the internal reactions exchanged between the finite elements has to be null. If we place a glass on a table, the glass acts on the table, but the table acts on the glass, with a force that is equal and opposite. The sum of the reaction exchanged between the table and the glass has to be null.

Note that in equation 5-6 the sum is possible because we have converted the description of what takes place from a local point of view (equation 5-3) to a global point of view: all the vectors $\mathbf{A}_e\, \mathbf{r}_{\mathbf{eg}}$ have the same dimension (equal to the overall number of degrees of freedom of the structure), regardless of "e", and thus can be summed. If we apply equation 5-6 to equation 5-5, we obtain the condition that eliminates the integral reaction forces and leaves only the forces connected to actions $\mathbf{A}_e\, \mathbf{f}_{\mathbf{eg}}$ and the forces of elastic recall:

$$\sum \mathbf{A}_e\, \mathbf{K}_{eg}\, \mathbf{D}_e\, \mathbf{u} - \sum \mathbf{A}_e\, \mathbf{f}_{\mathbf{eg}}=\mathbf{0}. \qquad [5\text{-}7]$$

In the first summation in equation 5-7 the term **u** can be factored out since it is always the same nodal displacement vector of the structure **u**

$$(\sum \mathbf{A}_e\, \mathbf{K}_{eg} \mathbf{D}_e)\, \mathbf{u} - \sum \mathbf{A}_e\, \mathbf{f}_{\mathbf{eg}}=\mathbf{0}.$$

Thus we can *define* two new matrices. The first is a square matrix of an order equal to the number of degrees of freedom of the structure, and is called the *global stiffness matrix*

$$\mathbf{K}\equiv\sum \mathbf{A}_e\, \mathbf{K}_{eg}\, \mathbf{D}_e. \qquad [5\text{-}8]$$

This is obtained by taking the sum of as many contributions as there are finite elements in the structure. Each local stiffness matrix is "exploded" by the assembly operation until it has the dimension of the final matrix, then the various thus obtained are summed. If $\mathbf{K}_{eg}$ is the stiffness matrix of the element in the global reference system *but with the local dimension and order of the nodes*, then the *exploded* stiffness matrix of the element is

$$\mathbf{A}_e\, \mathbf{K}_{eg}\, \mathbf{D}_e.$$

The second matrix that we *define* is a column vector whose order is equal to the number of degrees of freedom of the structure, and is called the *equivalent nodal forces vector*

$$\mathbf{f}\equiv\sum \mathbf{A}_e\, \mathbf{f}_{\mathbf{eg}}. \qquad [5\text{-}9]$$

This is also obtained by summing all the contributions of the various elements, and is called "equivalent" because it does not generally deal with forces that are actually applied, but rather with forces that, when multiplied by the corresponding nodal displacements, give an amount of work that is "equivalent" to that of the forces originally applied.

With equations 5-8 and 5-9 we can write an equation in matrix form that is the solution to the structural problem in question

$$\mathbf{K}\, \mathbf{u} - \mathbf{f}=\mathbf{0}. \qquad [5\text{-}10]$$

In this equation, the nodal displacement vector **u** is unknown, **K** acts as the coefficient matrix, and **f** as the vector of known terms.

5.4 ENERGY METHOD

In obtaining the result in equation 5-10 we made use of the equilibrium conditions of the individual element, found by means of the principle of the stationarity of total potential energy applied to the situation of the individual element. Nothing prevents us from establishing the equilibrium of the overall structure in the same way, that is, by writing first the total potential energy, and then establishing the stationarity.

The total potential energy Π_e of the individual finite element can be written as

$$\Pi_e = 0.5\, \mathbf{u_e}^T\, \mathbf{K}_e\, \mathbf{u_e} - \mathbf{u_e}^T \mathbf{f_e} - \mathbf{u_e}^T \mathbf{r_e}$$

(see equation 4-16). If we express the quantities transforming them into the global reference system, we can write

$$\Pi_e = 0.5\, \mathbf{u_{eg}}^T\, \mathbf{K}_{eg}\, \mathbf{u_{eg}} - \mathbf{u_{eg}}^T \mathbf{f_{eg}} - \mathbf{u_{eg}}^T \mathbf{r_{eg}}.$$

If we introduce equation 5-2 into this expression, we obtain the total potential energy of the element written as a function of the nodal displacement vector of the structure

$$\Pi_e = 0.5\, \mathbf{u}^T\, \mathbf{D}_e^{\mathbf{T}}\, \mathbf{K}_{eg}\, \mathbf{D}_e\, \mathbf{u} - \mathbf{u}^T \mathbf{D}_e^T\, \mathbf{f_{eg}} - \mathbf{u}^T \mathbf{D}_e^T\, \mathbf{r_{eg}}.$$

Recalling equation 5-3, we can write

$$\Pi_e = 0.5\, \mathbf{u}^T\, \mathbf{A}_e\, \mathbf{K}_{eg}\, \mathbf{D}_e\, \mathbf{u} - \mathbf{u}^T \mathbf{A}_e\, \mathbf{f_{eg}} - \mathbf{u}^T \mathbf{A}_e\, \mathbf{r_{eg}}.$$

If we want to obtain the total potential energy of the entire structure, we need only take the sum of all the contributions of the elements that make up the structure :

$$\Pi = \sum(0.5\, \mathbf{u}^T\, \mathbf{A}_e\, \mathbf{K}_{eg}\, \mathbf{D}_e\, \mathbf{u} - \mathbf{u}^T \mathbf{A}_e\, \mathbf{f_{eg}} - \mathbf{u}^T \mathbf{A}_e\, \mathbf{r_{eg}})$$

(the sum varies depending on "e").

In the above expression the last term of the summation has to be null, in that the work performed by the internal reaction forces is globally equal to 0, thus, factoring out of the summation the terms **u** and $\mathbf{u}^T$, which don't depend on "e", we can write:

$$\Pi = 0.5\, \mathbf{u}^T\, (\sum \mathbf{A}_e\, \mathbf{K}_{eg}\, \mathbf{D}_e)\, \mathbf{u} - \mathbf{u}^T\, (\sum \mathbf{A}_e\, \mathbf{f_{eg}}).$$

If we now introduce equations 5-8 and 5-9, we arrive at

$$\Pi = 0.5\, \mathbf{u}^T\, \mathbf{K}\, \mathbf{u} - \mathbf{u}^T\, \mathbf{f}.$$

The condition of the stationariness of total potential energy is written

$$\delta\Pi = \delta\mathbf{u}^{T}\,(\mathbf{K}\,\mathbf{u} - \mathbf{f})$$

and must be verified for every possible $\delta\mathbf{u}$ compatible with the constraints. It follows that

$$\mathbf{K}\,\mathbf{u} - \mathbf{f} = \mathbf{0},$$

which is equation 5-10 found by other means.

5.5 REMARKS ABOUT THE SYSTEM OF SOLUTION EQUATIONS

The matrix equation

$$\mathbf{K}\,\mathbf{u} - \mathbf{f} = \mathbf{0}$$

corresponds to a system of equations in which the stiffness matrix **K** acts as a coefficient matrix, and vector **f** as the vector of known terms, while vector **u** is the vector of unknown terms. In chapter 8 we will discuss in depth the numeric problems that arise during the solution of this system.

Finding the values of **u** that satisfy this equation means finding the displacements that correspond to the distortions of the finite elements that are able to generate the reaction forces that establish global equilibrium and that put the structure in equilibrium under the loads applied. Finding vector **u** means finding the configuration of the structure such that an infinitesimal variation of configuration, made starting from the configuration found, leads to a null first variation of total potential energy Π. The strain energy Z of the structure increases, but this variation is exactly compensated for by the work performed by the loads W, and thus there cannot be any kinetic energy C or any movement. If there exists some variation of configuration capable of allowing the onset of a variation of kinetic energy, the structure would move from that configuration, and could not be in equilibrium.

The matrix equation found expresses all of these facts by means of a configuration that is not described in the most general way possible, but rather starting from simplifying hypotheses made about the displacement field within each element.

Once the displacements of the structure are known, we can disassemble these displacements, to arrive at the displacements of the nodes of each element (thanks to equation 5-2), and from these, by means of the relationships between displacements and strains and the constitutive relations, to the stresses.

5.6 Remarks about the stiffness matrix

5.6.1 Assembly of a simple structure

Let's look more closely at what is involved in the assembly of global stiffness matrix **K**, formally defined in equation 5-8.

If we look again at the simple structure shown in fig. 5-1, we can see what happens regarding the assembly of the stiffness matrix of the various elements. Let's first consider the stiffness matrix of element 1.

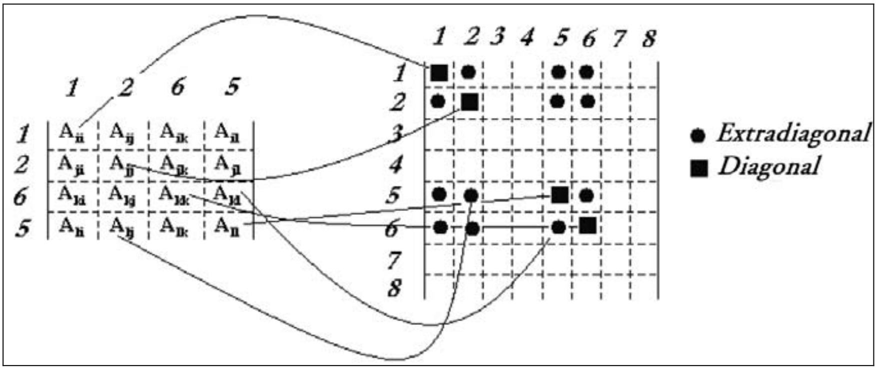


Figure 5-4: Schematic representation of the assembly of element 1 shown in fig. 5-1.

The element has as nodes (i, j, k, l) the nodes (1, 2, 6, 5). On the left in fig. 5-4 is the matrix of the element before its "explosion". The matrix is divided into small blocks that can be identified by the node number corresponding to the "row" and by the node number corresponding to the "column". For example, block $\mathbf{A}_{lj}$ corresponds to the row marked by node "5" and to the column marked with node "2". On the right in fig. 5-4 is the global stiffness matrix of the structure. The block marked with the same node on the row and on the column is the "target" block on which the block of the matrix on the left will position itself. Fig. 5-4 shows the positioning of a certain number of blocks. Each individual block is a small sub-matrix with a certain number of terms, depending on the type of finite element and on the formulation. Practically speaking, *the assembly of the stiffness matrix of the individual element* $\boldsymbol{K}_{eg}$ *consists in the appropriate redistribution of the blocks of the starting* $\boldsymbol{K}_{eg}$ *into the stiffness matrix of the structure* $\boldsymbol{K}$*.*

Fig. 5-5 shows the same kind of reasoning applied to the matrix of element 3. The matrix $\mathbf{K}_{eg}$ of this element *does not have the same dimension as element 1*, because element 3 has only 3 nodes, but final matrix **K** remains the same. We begin with an empty matrix, filled only with zeroes, and then we sum all of the blocks of the various matrices of all the elements. If, when a block is "placed" it does not find null values in the destination block, *the new block is summed (in keeping with 5-8) with the pre-existing block.*

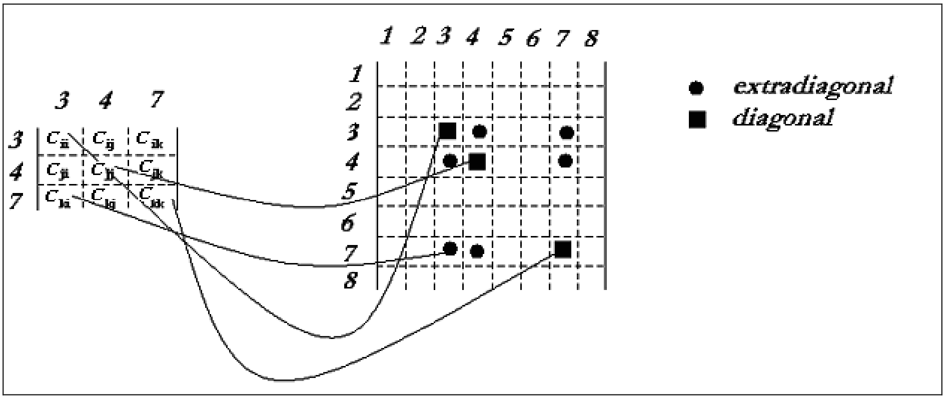

1 2 3 4 5 6 7 8
1
2
3
4
5
6
7
8
3 4 7
3
4
7
C_{ii} C_{ij} C_{ik}
C_{ji} C_{jj} C_{jk}
C_{ki} C_{kj} C_{kk}
● extradiagonal
■ diagonal

Figure 5-5: Schematic representation of the assembly of element 3 shown in fig. 5-1.

For example, consider the diagonal block corresponding to node 6 on the global stiffness matrix. Because of what is shown in fig. 5-4 after the assembly of the stiffness matrix of the first element, we will have

$$\mathbf{K}_{66} = \mathbf{A}_{kk}.$$

But node 6 is also connected to element 2; more precisely, node 6 corresponds to "node 1" of element 2. Thus, if we imagine that the blocks of the matrix of element 2 are identified by the letter "B" (in the same way we used the letter "A" for the first element and the letter "C" for the third), after the assembly of the second element we will have

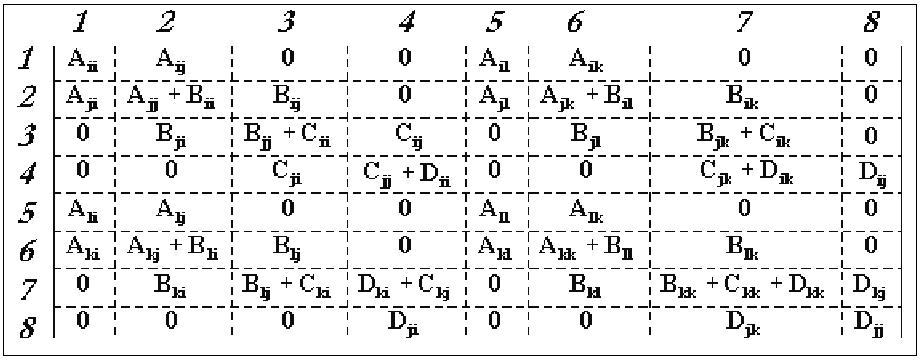
$$\mathbf{K}_{66} = \mathbf{A}_{kk} + \mathbf{B}_{ll}.$$

Summarising the data about the four elements in the following table, we can write how the stiffness matrix of the structure looks after assembly.

Element	Letter of sub-blocks	Local nodes	Global nodes
1	A	i, j, k, l	1, 2, 6, 5
2	B	i, j, k, l	2, 3, 7, 6
3	C	i, j, k	3, 4, 7
4	D	i, j, k	4, 8, 7

Table 5-1: Data relative to the structure shown in fig. 5-1.

The final matrix is shown in fig. 5-6:



	1	2	3	4	5	6	7	8
1	A_{ii}	A_{ij}	0	0	A_{il}	A_{ik}	0	0
2	A_{ji}	$A_{jj} + B_{ii}$	B_{ij}	0	A_{jl}	$A_{jk} + B_{il}$	B_{ik}	0
3	0	B_{ji}	$B_{jj} + C_{ii}$	C_{ij}	0	B_{jl}	$B_{jk} + C_{ik}$	0
4	0	0	C_{ji}	$C_{jj} + D_{ii}$	0	0	$C_{jk} + D_{ik}$	D_{ij}
5	A_{li}	A_{lj}	0	0	A_{ll}	A_{lk}	0	0
6	A_{ki}	$A_{kj} + B_{li}$	B_{lj}	0	A_{kl}	$A_{kk} + B_{ll}$	B_{lk}	0
7	0	B_{ki}	$B_{kj} + C_{ki}$	$D_{ki} + C_{kj}$	0	B_{kl}	$B_{kk} + C_{kk} + D_{kk}$	D_{kj}
8	0	0	0	D_{ji}	0	0	D_{jk}	D_{jj}

Figure 5-6: Stiffness matrix for the structure shown in fig. 5-1 after assembly

Although a matrix like this can be obtained formally by applying equation 5-8, no computer program does it this way. Instead, the procedure used by the programs follows the kind of distribution of blocks described in figs. 5-4 and 5-5. The first thing a program does is to create an empty space that corresponds to the space required for the stiffness matrix.[1] Then it cycles through the various elements that make up the structure and constructs the stiffness matrix of each element $\mathbf{K}_e$. Next, the program converts this stiffness matrix to the global reference system by means of expressions like 4-2, obtaining $\mathbf{K}_{eg}$. Once $\mathbf{K}_{eg}$ has been obtained, the program partitions it into blocks and sums each block in the position appropriate for *that* block of *that element*, in the context of the more general stiffness matrix **K**. The transfer of blocks into the various positions is what effectively constitutes the operation of assembly. Naturally, analogous reasoning is also valid for vectors of known terms, which are also partitioned into blocks and appropriately transferred.

If we analyse the matrix in fig. 5-6, we can note the following important features:

1. Some blocks remain completely null, and they have no corresponding stiffness terms;
2. Some blocks are particularly full (for example, block 7-7);
3. Some contain only a single term;
4. The global stiffness matrix is symmetrical;
5. The only blocks that lie on the main diagonal are those that lie on the main diagonal of the starting matrix (that is, blocks with equal indices).

[1] To be more precise, it prepares a space relative to *part* of the stiffness matrix, that is, the part that contains terms that are sure to be non-null. This will be discussed in detail in chapter 8.

5.6.2 THE STATIC INTERPRETATION OF THE TERMS OF THE STIFFNESS MATRIX

Let's consider a generic row of the solution equation. In algebraic terms, presuming *N* to be the total number of degrees of freedom, we can write the generic *i*-th row of the solution system in the following way:

$$K_{i1}u_1+ K_{i2}u_2+ K_{i3}u_3+\ldots. K_{ij}u_j+\ldots K_{iN}u_N - f_i = 0$$

or more concisely, introducing the summation with *j* going from 1 to *N*,

$$\sum K_{ij}u_j-f_i=0.$$

The terms K_{ij} represent the elements of the *i*-th row of the stiffness matrix. The written equation corresponds to the displacement of a certain node in a certain direction, to the *i*-th degree of freedom, and expresses the fact that the sum of all the action and reaction forces acting on *that* node in *that* direction, and thus on *that* degree of freedom must be null. The K_{ij} terms correspond to the forces on the *i*-th degree of freedom because of the *unitary* movement of the *j*-th degree of freedom, *all the other degrees of freedom being null.*

In contrast, $K_{ij}u_j$ terms correspond to the forces on the *i*-th degree of freedom because of the fact that the *j*-th degree of freedom has undergone a displacement u_j. It shouldn't come as a surprise that forces appear on the *i*-th degree of freedom when the *j*-th degree of freedom moves. Imagine a cord attached to a pole; we pull the cord to its full length and then move its free end. When we exert a force on the free end, *a force acts on the constrained end.* The cord transmits the stresses to the other end, which hasn't actually been touched. Thus, in order for some force to be transmitted to the *i*-th degree of freedom when the *j*-th degree of freedom is moved, the two must be connected by something. In the context of the FEM we can see that in order for the displacement of degree of freedom *j* to transmit something to degree of freedom *i*, *all of the other movements different from* u_j *being null*, the two degrees of freedom have to be directly connected by at least one element that procures a reaction like the one being sought. In order for this to take place, it is necessary (but not sufficient, as we will see) that the node to which degree of freedom *i* belongs and that to which degree of freedom *j* belongs are both connected to a single element that is distorted. In fact, let's suppose that degree of freedom *j* moves. If none of the elements containing degree of freedom *i* contain degree of freedom *j* , then none of these elements will be deformed (since the displacements of their nodes are null) *and thus they will not transmit any force due to movement* u_j *to the nodes belonging to them.* As a consequence, the corresponding term in the stiffness matrix has to be null:

$$K_{ij}=0.$$

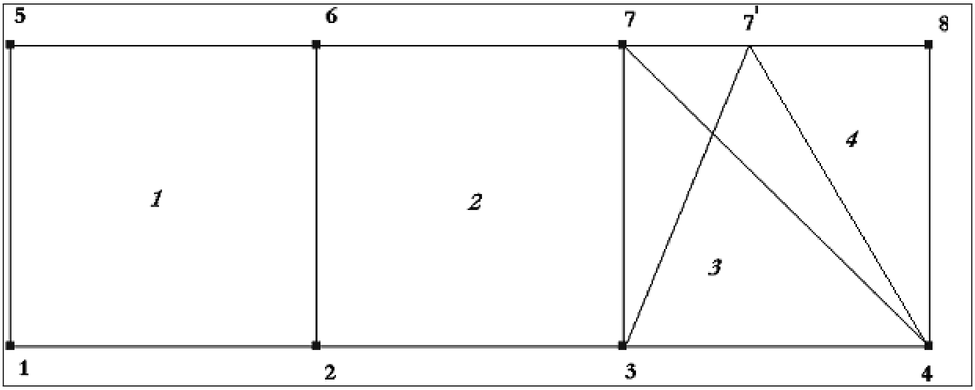


Figure 5-7: Unitary displacement of node 7 to 7'.

The term K_{ii} represents that force on degree of freedom *i* due to its own movement, called *direct stiffness*. As we can see, the terms of direct stiffness lie on the main diagonal of the stiffness matrix. If all of the nodes of the structure are connected to elements, then the diagonal terms of the stiffness matrix associated to movements limited by the stiffness of the element are all non-null. This is a translation into mathematical terms of the fundamental circumstance that *if one node attached to an element is moved while all the others remain fixed, a certain resistance is encountered.*

Let's look at fig. 5-7. If we imagine that displacement u_j is the translation of node 7 in direction *x*, such that it goes to 7', we can see that element 1 remains undeformed, while elements 2, 3 and 4 all deform. Because of the deformation of these elements, forces are transmitted to all of the nodes *except for nodes 1 and 5, which are not connected to node 7 by any element.* This is why, if we look at the matrix in fig. 5-6, the blocks obtained by combining numbers 1 and 7 and numbers 5 and 7 – that is, blocks 1-7, 7-1, 5-7 and 7-5 – are null.

If we look at the nodes one at a time, by examining the distorted and undistorted elements we can predict the blocks where there will be a null stiffness (see table 5-2).

If we look at what we have found, we see that all of null blocks present in the global stiffness matrix have been identified. The most crowded terms are those in which the displacements of a node distort the greatest number of elements (for example, term 7-7), while those that correspond to a single term are those in which the movements of a node transmit forces to another node by means of a single finite element (for example, 5-1 or 1-5). Choosing a row and a column of blocks in the stiffness matrix means choosing one of its blocks: the number of blocks in the stiffness matrix converging to that "global" block is equal to the number of finite elements that are distorted by the movement of the node corresponding to the row in question.

Node movement	**Distorted elements**	**Undistorted elements**	**Nodes without reactions**	**Nodes with reactions**
1	1	2, 3, 4	3, 4, 7, 8	1, 2, 5, 6
2	1, 2	3, 4	4, 8	1,2,3,5,6,7
3	2, 3	1, 4	1, 5, 8	2,3,4,6,7
4	3, 4	1, 2	1, 2, 5, 6	3,4,7,8
5	1	2, 3, 4	3, 4, 7, 8	1,2,5,6
6	1, 2	3, 4	4, 8	1,2,3,5,6,7
7	2, 3, 4	1	5, 7	1,2,3,4,6,8
8	4	1, 2, 3	1, 2, 3, 5, 6	4,7,8

Table 5-2: Movements of the nodes and resulting distortions of the elements.

5.6.3 Energy interpretation of the Stiffness Matrix

Let's look at the quantity expressed by the quadratic form in **u**:

$$Z=0.5\mathbf{u}^{\mathbf{T}}\mathbf{K}\mathbf{u}. \qquad [5\text{-}11]$$

In terms of its physical meaning, this is a number that indicates the elastic strain energy stored in the structure as a consequence of the displacements **u**. An examination of the physical significance leads us to the conclusion that there must be

$$\mathbf{K}^{T}=\mathbf{K},$$

since according to Betti's theorem the work performed by displacements $\mathbf{u_1}$ times the forces $\mathbf{Ku_2}$ has to be equal to the work performed by the displacements $\mathbf{u_2}$ times the forces $\mathbf{Ku_1}$:

$$\mathbf{u_1}^{\mathbf{T}}\mathbf{Ku_2}=\mathbf{u_2}^{\mathbf{T}}\mathbf{Ku_1}= \mathbf{u_1}^{\mathbf{T}}\mathbf{K}^{\mathbf{T}}\mathbf{u_2}.$$

That matrix **K** is symmetrical had already emerged from equation 5-8 together with equation 5-3. In fact, recalling that the matrix of each $\mathbf{K}_{eg}$ is symmetric, by transposing equation 5-8 we obtain

$$\mathbf{K}^{\mathbf{T}}= (\sum \mathbf{A}_e\, \mathbf{K}_{eg}\, \mathbf{D}_e)^{T}=(\sum \mathbf{D}_e{}^{\mathbf{T}}\, \mathbf{K}_{eg}{}^{\mathbf{T}}\, \mathbf{A}_e{}^{\mathbf{T}})= (\sum \mathbf{A}_e\, \mathbf{K}_{eg}{}^{\mathbf{T}}\, \mathbf{D}_e)=\mathbf{K}.$$

The quadratic form described by equation 5-11 would appear to say that given any set of displacements **u** there must be a non-negative strain energy Z. In reality, we see that this cannot be true: if we rigidly translate the entire structure then the strain energy must be null. In effect, the quadratic form is not strictly positive (that is, it is not always greater than zero for every possible non-null vector **u**), but is only *semi-defined positive* (that is, it is always *equal to or greater than* zero for every possible non-null vector **u**), since the stiffness matrix **K** written up to this point is actually singular, that is,

$$\mathrm{Det}(\mathbf{K})=0.$$

Example.

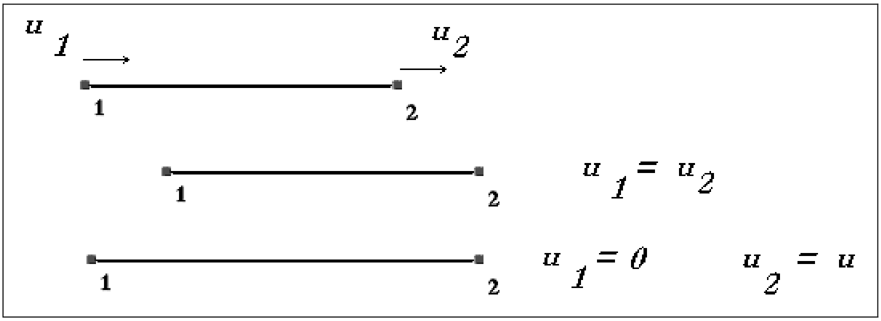


Figure 5-8: Various possible movements of a structure composed of rods.

In order to demonstrate this clearly, let's consider a structure composed of a simple rod and look at what form the strain energy takes. The rod has two nodes, and there are two unknown displacements, u_1 and u_2. The rod can be likened to a spring of stiffness *k*. The strain energy Z can be written

$$Z=\frac{1}{2}\left|\begin{array}{c:c} u_1 & u_2 \end{array}\right|\left|\begin{array}{c:c} k & -k \\ \hdashline -k & k \end{array}\right|\left|\begin{array}{c} u_1 \\ \hdashline u_2 \end{array}\right|$$

which, when multiplied out, becomes,

$$Z=0.5(ku_1^2-2k\ u_1\ u_2+\ ku_2^2).$$

Let's look now at what happens when there is a rigid translation, such that $u_1=u_2=u$ (cfr. fig. 5-8):

$$Z=0.5(ku^2-2k\ u\ u+\ ku^2)=0.$$

We can see that the strain energy is null. If instead we consider the movement defined by $u_1=0$, $u_2=u$ (that is, an elongation of the rod, cfr. fig. 5-8), we obtain

$$Z=0.5(ku^2)\neq 0.$$

If we calculate the determinant of the stiffness matrix of the structure composed of a single rod, we obtain

$$\mathrm{Det}(\mathbf{K})=k^2-k^2=0.$$

that is, the determinant of the stiffness matrix is null, as was said earlier.

♣

What we have just established for a simple rod is equally valid for a complex structure: the assembly itself of the elements leads to a singular stiffness matrix, since the rigid motions have not yet been excluded. It is possible to have these rigid motions without the system storing any strain energy Z.

Note that system 5-10 cannot be solved as long as the stiffness matrix **K** remains singular: it is thus necessary to modify the system in order to make it solvable. The modifications mainly regard establishing the constraints that prevent there from being any rigid motions in the structure, and this will be discussed in detail in chapter 7.

5.6.4 TYPICAL CHARACTERISTICS OF THE STIFFNESS MATRIX

To summarise the characteristics of the stiffness matrix discussed up to this point (and to list others that have not yet been discussed), we can say that after assembly the stiffness matrix **K** has the following characteristics:

- It is symmetrical;
- It is semidefined positive;
- It has a null determinant;
- It is *sparse* (that is, it contains many null terms and few non-null terms);
- It is *banded* (that is, the non-null terms tend to be grouped around the main diagonal).

The last two characteristics require comment. The fact that the matrix is sparse depends on the fact that if we consider a generic node (and thus, a generic "row" of the matrix partitioned into blocks), this node will turn out to be connected to only a few of the elements that make up the structure (the example shown in fig. 5-1 is misleading; in real models there are many elements and many nodes), and so only a few nodes are in a condition to receive reactions when the node in question moves. The majority of the nodes will not be directly connected to the node in question, and thus all of the "mixed" blocks of the stiffness matrix involving that node and nodes not directly connected will be null.

The fact that the matrix is "banded" is related to the fact that in general the nodes are located in geometric proximity, and are not numbered very differently, and thus the nodes connected to the elements converging at a generic node will have numbers that are not too different from it. The terms will thus tend to arrange themselves about the main diagonal, including, as we said earlier, the terms of "direct" stiffness.

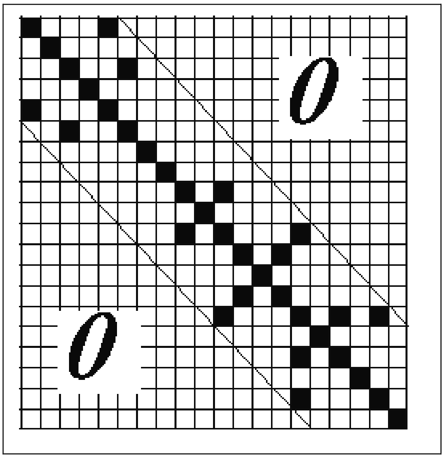


Figure 5-9: Typical appearance of a stiffness matrix: symmetrical, sparse and banded.

As we shall see in more detail in chapter 8, the fact that the matrix is symmetrical, sparse and banded is very helpful for solving the problem numerically, since there are special solving techniques that make it possible to put these characteristics to good use.

5.7 HETEROGENEITY OF ELEMENTS

Up to now we have implicitly hypothesised that all the finite elements of a structure are identical, and thus that all the blocks $\mathbf{A}_{ij}$ or $\mathbf{B}_{ij}$, and so forth belonging to the global stiffness matrix are of the same dimension. However, this is never the case in practice, since there can be various elements in a structure, either because they are used to model different parts of the structure subjected to different formulations (beams, plates, membranes, etc.), or perhaps because, even within the context of a single formulation, there are elements designed in different ways.

In order to understand how to deal with these cases, we need to understand better how to describe the generic "block" of stiffness matrix **K**.

Let $\mathbf{K}_{ij}$ be the block relative to the forces acted on node *i* caused by movements of node *j*. If at each node there are *m* possible components of unknown displacement and *m* degrees of freedom, then block $\mathbf{K}_{ij}$ is of order *m*. The problem is that the unknown displacements can be different for the various elements, given that different formulations are possible.

The best way to address the problem is to consider the most general case possible. In ordinary finite element software programs each node can have at most 6 different unknown nodal displacements, 3 translations and 3 rotations.

Therefore we have to start from the principle that the order m *of the typical sub-block of the stiffness matrix is equal to 6, and that the generic vector of unknown displacements in a node* i *is made like this:* [2]

$$\mathbf{u}_i=\{u, v, w, \theta, \varphi, \psi\}^T$$

where

u is the translation of the node in direction x ;

v is the translation of the node in direction y ;

w is the translation of the node in direction z ;

θ is the rotation of the node about the x-axis;

φ is the rotation of the node about the y-axis;

ψ is the rotation of the node about the z-axis.

In some particular cases, when the structure is made up of homogeneous finite elements, and when the formulation does not make systematic use of certain nodal displacements, only some of these nodal displacements will be associated with stiffness, while the others – that is, those that are not used and those to which there are no corresponding stiffness terms in the formulation – are *inactive.*

For the sake of clarity, let's first consider a rod element defined in plane xz. The only two components of the displacement of the two nodes of the element associated with stiffness are translation u and translation v: $\mathbf{u}_{eg}=\{u, w\}^T$. Neither translation v nor any of the rotations of the nodes of the element provoke any reaction on the part of the element itself.

The assembly procedure is briefly summarised in fig. 5-11. Note that here, in contrast to what was shown in figs. 5-4 and 5-5, *we don't see the assembly of the blocks*, but rather the assembly of the terms that correspond to *a* block. For example, imagine that the end nodes of the rod in fig. 5-10 are nodes *i* and *j*, and that, in the stiffness matrix of the element, the block corresponding to the reactions transmitted to node *i* by the movements of node *j* is block $\mathbf{A}_{ij}$. Given the formulation of the rod element, this block has dimension 2x2. In the global stiffness matrix **K**, block $\mathbf{A}_{ij}$ corresponds to the position of block $\mathbf{K}_{ij}$, *but block* $\boldsymbol{K}_{ij}$ *does not have dimension 2x2, but rather dimension 6x6*. In fig. 5-11 we see that, in the sub-block of the global matrix, each term is put exactly in its proper place, *while the stiffnesses associated with the global degrees of freedom that are not connected to any stiffness of the rod remain null.*

[2] There are of course cases that are still more general, in which are used, for example, higher order derivatives, but they are not normally found.

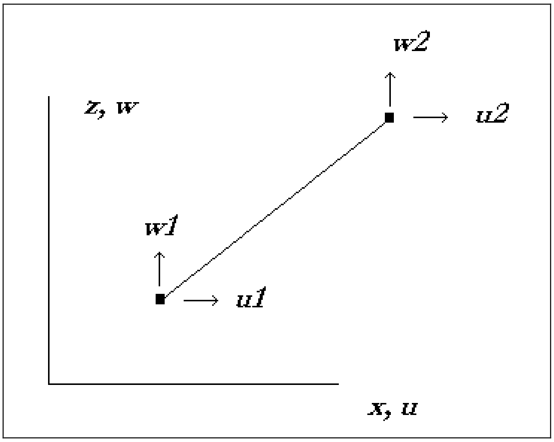


Figure 5-10: Rod element in plane *xz*.

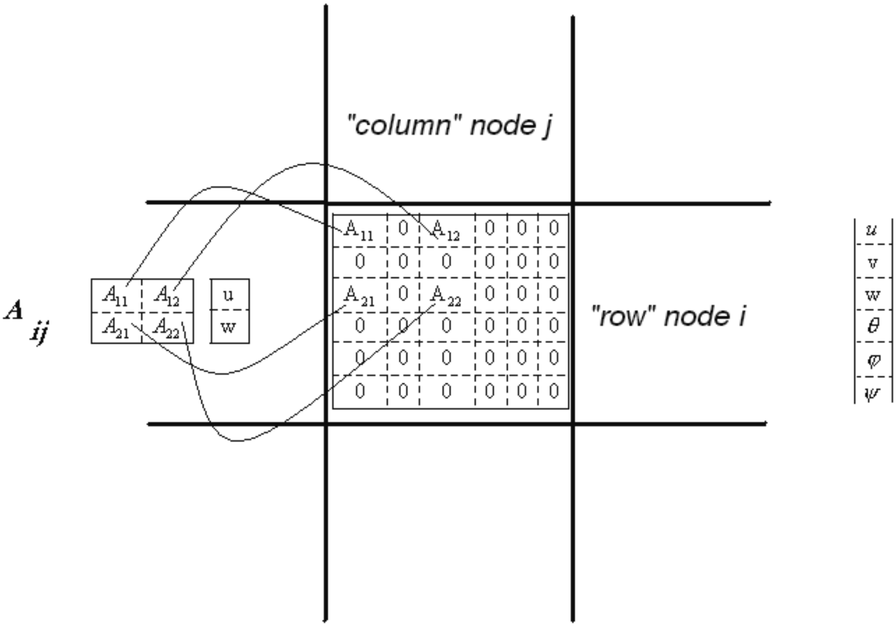


Figure 5-11: Inside a sub-block: assembly of a sub-block of the matrix of a rod element of dimensions 2x2 and with only 2 translational degrees of freedom in a sub-block of matrix K of dimensions 6x6 and with 3 degrees of freedom in translation and 3 degrees of freedom in rotation.

We can now understand what happens when different elements are assembled: these elements have stiffnesses that generally involve different degrees of freedom of the nodes of the structure (in this example only u, w; in other examples there might be only w, θ, φ, and so forth). During the assembly procedure these stiffnesses are gradually added in their proper positions (whatever these might be, and even different among themselves), until the list of the elements is exhausted. *At the end of assembly there might be rows of the global stiffness matrix that are composed of only zeroes: these rows will be associated with inactive or fictitious degrees of freedom.* To ascertain if a degree of freedom is inactive or fictitious, it is only necessary to imagine that the node to which it is connected moves in the direction associated with that degree of freedom: *if the finite elements with their formulations possess at least one stiffness capable of opposing that movement, the degree of freedom is active; if there is no element that possesses such a stiffness, then the degree of freedom is inactive.*

Each row of the stiffness matrix that corresponds to an inactive degree of freedom is entirely made up of null terms, and so is the corresponding column (that is, the one with the same number as the row). Hence, we can see that each inactive degree of freedom is an independent cause for the determinant of the coefficient being null, and thus of the impossibility of solving the system of fundamental equations. *Like rigid motions, inactive degrees of freedom may also make it impossible to solve the system of equations.*

Note that the considerations about inactive degrees of freedom *do not pertain to the actual structure, but rather to the idealisation and schematisation inherent in the finite element model.*

For example, if we are considering the nodes of a membrane, there is no stiffness at all of the membrane element associated to the rotation of a node connected to it, but this does not mean that when we rotate a T-bar embedded in a wall (modelled with membrane elements), that rotation is free! Nor can we say that a beam "hinged" at its ends and thus modelled with end releases effectively rotates freely around its ends, given that there are connecting plates and three or four bolts.

The inexperienced user of finite elements typically confuses the physical reality, which is one thing, with the finite elements model, which is another thing altogether. Thus to the question, "Can the finite element that simulates a T-bar, connected to the membrane, rotate freely?", the inexperienced user, thinking of the physical reality, answers "no", while in fact, in the context of the FEM, the answer is "yes".

6 ACTIONS

6.1 INTRODUCTION

Now that we have seen how to assemble the stiffness matrix for the structure as a whole and the vector of equivalent nodal forces, we will now go into detail about how to generate the known terms of the system of equations that solve the problem, that is, the vector of equivalent nodal forces.

It is clear that assembly of the vector of known terms follows rules that are quite similar to those already described for the matrix of the system: to each element corresponds a vector of equivalent nodal forces, and each of these vectors will come to be assembled, "exploded" to the complete dimension of the problem, situating the various blocks that make up the vector itself in their appropriate positions in the global vector. These are transformed into the system of global coordinates and then assigned to the vector of known terms in the appropriate position for each node.

We must better explain how these equivalent nodal forces are generated, what they correspond to, and what kind of approximation we can expect when the loads are schematised in one way rather than in another.

"Nodal forces" or "nodal couples" are actions that are applied directly to the nodes of a structure. "Distributed loads" are actions that are applied to points within the finite elements and which thus do not have any particular node that corresponds to the point where they are applied. Note that in the vocabulary of finite elements the term "distributed" has a slightly different meaning than when it is used in statics and structural engineering in general. Specifically, a "concentrated" force applied at a point in a finite element is designated as a "distributed" load in the terminology of finite elements.

In addition to actions that are directly applied in the form of forces or couples (whether concentrated or not), there are also actions that act on the structure without their being any direct application of forces or couples: these are the important family of "thermal loads", that is, those belonging to the family of actions that correspond to a warming or cooling of the structure as a result of atmospheric agents and/or artificial causes.

In this chapter we will also look at a series of other questions related to modelling the actions in the context of the FEM.

6.2 NODAL FORCES OR COUPLES

The first and most immediate type of action that can be applied to a structure modelled with the FEM is constituted by *nodal* forces or couples. As a convention we will use the term "forces" in a generalised sense, that is, to refer

Structural analysis with finite elements
978-0-7277-4093-9

to forces as well as couples, and thus for the sake of brevity we will speak of *nodal forces*.

Nodal forces produce an amount of work given by the product of their modulus by the displacement (translation or rotation) of the node to which they are applied, in the direction of the forces themselves. Since the displacements of each unconstrained node are part of the vector of unknowns, the work performed by nodal forces can be evaluated directly by multiplying the (known) forces by one or more of the primary unknowns of the problem at hand (the nodal displacements). Even though these are approximate, we can see that in the context of the FEM this is the best possible kind of schematisation, since the work is modelled with the same degree of precision used for the displacement field of the nodes.

On the other hand, having to position a node in correspondence to each force acting on the structure can be too tedious, because this approach can lead to both excessive breaking or distortion of the mesh, and an excessive number of nodes (and thus of degrees of freedom). Further, in the cases of loads that are spread out rather than concentrated, the schematisation by means of "nodal" forces can turn out to be impossible. In the next section we will see how this difficulty can be overcome.

In general we can say that it is good practice to place a node where there are significant concentrated applied loads – significant in the sense that, presumably, a significant part of the structural response depends on them –, with the aim of modelling the loads in question as nodal forces.

The nodal forces have to be applied to nodes to which elements are attached, and furthermore, these elements have to have a degree of stiffness (translational for forces, bending or rotational for couples) capable of opposing the action of the force. If this is not the case, the force is applied to an inactive degree of freedom and creates a problem during solving (cfr. chapter 7,activated mechanism).

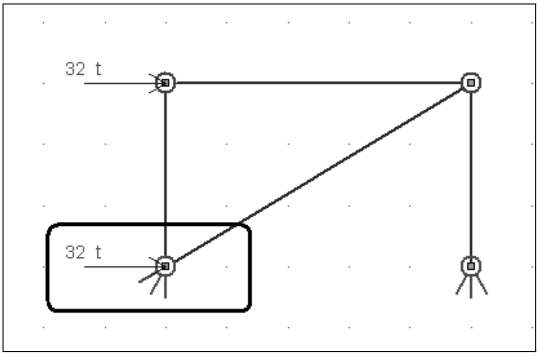


Figure 6-1: An example of a "grounded" nodal force.

A particular case of nodal forces occurs when these are directly applied to nodes that are completely or partially constrained. If the force is such that all of its components in reference to the global axes act on a constrained degree of freedom, it is completely "grounded"; if instead only some of its components are in this condition while the others are directed according to effective degrees of freedom, then it is partially "grounded". If we consider the components, we can see that the component of a nodal force can only be either completely "grounded" or not at all.

"Grounded" forces do not lead to any additional work, since the displacements that cause them are null. Thus the unknown nodal displacement field does not depend on them. The software programs for finite elements follow two possible routes for their treatment: the first consists in eliminating these forces altogether, since they are irrelevant; the second consists in eliminating these forces while, however, taking the reactions of the constrains into account in the calculations. This second approach is clearly preferable, because it is quite possible for the algorithms that automatically generate the forces that result in the creation of these "grounded" nodal forces. It is a good idea then to remain aware of these forces, even though they are neutral as far as stress and strain are concerned.

Nodal forces do not require a genuine "assembly", since they are directly applied to the nodes. They are usually defined directly in the global reference system, thus no coordinate transformation is required either.

In defining the nodal forces it must be borne in mind that not all software programs allow forces to be applied additively. In fact, in some programs the application of a force on a node eliminates all the forces previously applied on the node.

6.3 LOADS "DISTRIBUTED" ON ELEMENTS

When the generalised force to be applied does not act on a node but rather on a point within a finite element, or when the action – by its nature – does not act on a particular point but is spread over a certain area of the structure, then it is necessary to use "distributed loads".

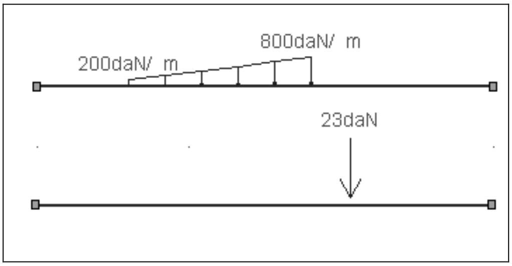


Figure 6-2: Examples of distributed loads in a beam element.

How do we evaluate the effect of distributed loads on the nodes of each element? There are two methods for doing this, each with its pros and cons.

6.3.1 THE *CONSISTENT* APPROACH

The first approach is the one described in chapter 4, and uses an energy criterion: the equivalent nodal forces are evaluated so that they produce a quantity of work identical to that produced by the original loads in the context of the most general possible displacement described by the shape functions. The work performed by the actions is obtained by multiplying the force modulus by a displacement that is not modelled directly but rather *indirectly* by means of the shape function. This is thus an approximation that is intimately related to what we called the *astrological principle*, that is, to an ulterior approximation with respect to what was predicted by the nodal forces. The displacement of the nodes is explicitly set as an unknown and its value is directly calculated in an approximate way, while the displacement of the points within an element is obtained from the nodal displacements and the shape functions, that is, by means of the simplifying hypotheses made about the set of displacements within the finite elements.

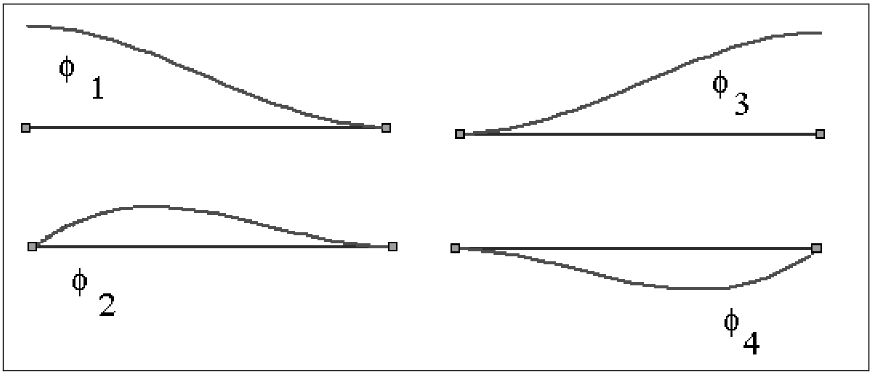


Figure 6-3: Shape functions for beam elements.

For greater clarity, let's look at how to calculate the equivalent nodal forces with the consistent approach, for an ordinary finite beam element with a load concentrated at ¾ of its clear length. In this case the displacement function is the elastic line v(x), and the nodal displacements are given by the vector $\mathbf{u}=\{v_i, \psi_i, v_j, \psi_j\}^T$. In practical terms, we take into consideration the translations and the rotations of the element's two nodes. It results that

$$v(x) = \boldsymbol{\phi}\mathbf{u}$$

where

$$\boldsymbol{\phi}=\{\phi_1, \phi_2, \phi_3, \phi_4\} = \begin{vmatrix} 1-\frac{3x^2}{L^2}+2\frac{x^3}{L^3} & x-2\frac{x^2}{L}+\frac{x^3}{L^2} & 3\frac{x^2}{L^2}-2\frac{x^3}{L^3} & -\frac{x^2}{L}+\frac{x^3}{L^2} \end{vmatrix}.$$

The work W performed by force P applied at ¾ of the clear length is obtained by multiplying the force by the deflection w(3/4L), thus

$$W=Pv(3/4L)=P\{5/32, 3L/64, 27/32, -9L/64\}\{v_i, \psi_i, v_j, \psi_j\}^T.$$

To this expression for the work correspond the following equivalent nodal forces:[1]

$$\{5P/32, 3PL/64, 27P/32, -9PL/64\}.$$

The two forces are not divisible in a way that can be predicted *a priori*: while it is true that the sum of the two nodal forces gives rise to the applied load (5+27=32), the division between nodes is not ¼ - ¾ as we might think. Further, couples appear whose values likewise cannot be predicted *a priori*.

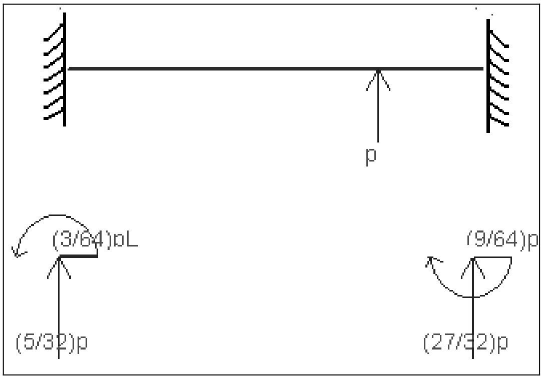


Figure 6-4: Equivalent nodal forces evaluated using the consistent approach.

The use of energy criteria to evaluate the equivalent nodal forces, rigorously in keeping with the hypotheses made about the displacement field, gives rise to a set of *equivalent* nodal forces which can have properties that cannot be immediately traced back to balanced elementary schemes. For example, a load can be divided unequally between the various nodes of the element, or it is possible that along with a translational component there is also an unforeseen rotational component (couples in addition to forces).

The so-called *consistent* approach, which makes use of shape functions, is the one that is most coherent with the concepts of the FEM, but as we said, it is not the only one.

[1] We omit the calculation of the variation of work for a generic variation in configuration, since this can be immediately obtained through the variation of the vector **u**: **δu**.

6.3.2 THE *LUMPED* APPROACH

A second method, quicker than the *consistent* method, is often used for evaluating the equivalent nodal forces. This method consists in dividing up the resultant of the distributed load on the nodes in proportion to the respective areas of influence, or by means of considerations about equilibrium (instead of the energy equivalents of quantities of work). This second approach, called the *lumped* approach, has the advantage of making calculation of the equivalent nodal forces simpler and faster, but it is only reliable when the elements are sufficiently small in size or when the effects of the loads on the elements is negligible with respect to the distortional effects, related to movement relative to the element's nodes.

The stress within an element can always be evaluated as the sum of a "local" stress and a "global" stress.

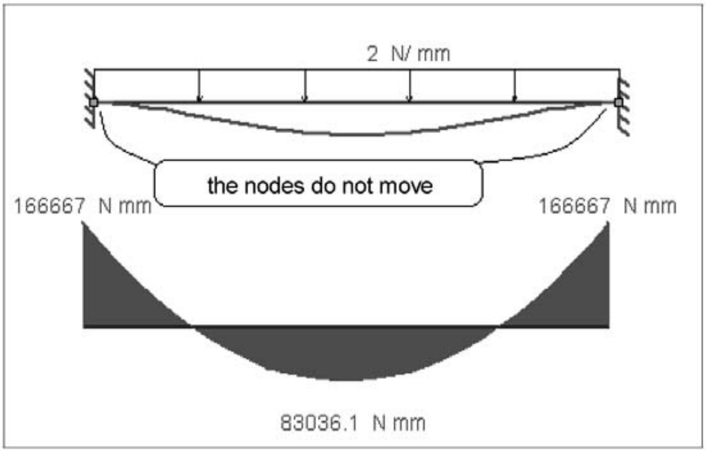


Figure 6-5: The effect of keeping nodes fixed within an element: deformed configuration and "local" bending moment diagram of a element fixed at both ends.

The "local" stress can be obtained by imagining that the element's nodes do not move, and evaluating the effect that the loads applied to the element has on it. This stress corresponds to a strain that cannot be represented by the shape functions, because it occurs at fixed nodes. For example, for a beam element constrained at both ends and subject to a uniformly distributed load, only the stress to which a strain that does not imply any rotation or translation of the nodes can be read.

On the other hand, the "global" stress is the one that depends on the fact that the nodes of the element move, and in moving have generated distortions in the element itself that are responsible for strains and thus for stresses. These stresses arise in the element because of the displacements of its nodes – connected to the rest of the structure – caused by the structure itself.

In the context of the FEM, usually only the "global" part of the stress is evaluated, the part due to the movement of the nodes, and thus in general the "local" part has to be negligible with respect to this, a condition that is obtained only by adopting finite elements that are sufficiently small.

Using the lumped approach in the same example used for evaluating the forces with the *consistent approach*, we see that in a lumped approach, the equivalent nodal forces corresponding to the element with a load at ¾ of its clear length, are as shown in fig. 6-6, with ¼P at the end furthest from the load and ¾P at the end closest to it, on the basis of simple considerations of equilibrium. There are no couples.

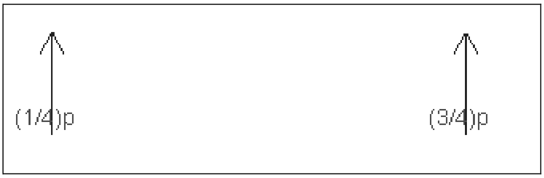


Figure 6-6: Equivalent nodal forces using the lumped approach.

6.3.3 FIXED END FORCES (BEAMS)

As we have seen, the second method for evaluating equivalent nodal forces divides the resultant of the loads applied on the element between the nodes on the basis of elementary considerations. The aim is to evaluate the forces that act on the nodes and make them want to move. These forces are evaluated as a lump sum, and thus an error is committed. What has to be done in order to evaluate these forces *exactly*?

We have to think of the finite element as a small, isolated structure (fig. 6-7), keep the nodes fixed, apply the loads, and read the constraint reactions on the nodes. The constraint reactions would have the same modulus and the opposite sign of the equivalent nodal forces we are looking for. Normally, this approach isn't useful on generic finite elements, because examining this kind of "little structure" would present all the problems typical of the beginning structure, and so the "little structure" could not be solved. There is, however, an important special case in which the "little structure" can be solved in closed form: the case of the beam.

In this particular case the constraint reactions under the most diverse load conditions can be calculated separately with relative ease. In the case of beam elements, the equivalent nodal forces are also called *fixed end forces*. In essence, the equivalent nodal forces are – except for the sign – constraint forces applied on the nodes, when the structure is considered to be made like the single finite element, with the distributed loads applied to the finite element itself. The sign changes, because the distributed loads and the nodal forces equivalent to them would tend to move the nodes, while the constraint reactions instead keep them still.

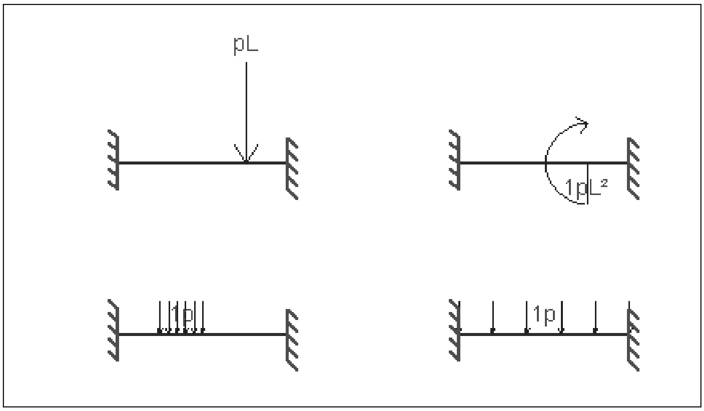


Figure 6-7: Distributed loads on beam elements and schemes for calculating the corresponding fixed end forces. It is possible to calculate the exact values in all cases.

If we look again at the case already examined of the element with the load concentrated at ¾ of its clear length, the fixed end forces are evaluated by solving the structure given by a beam whose clear length is equal to the clear length of the finite element, fixed at both ends, and subject to a load equal to P at ¾ of its clear length.

The result obtained, shown in fig. 6-8, is the vector of equivalent nodal forces opposite to the vector of the constraint reactions:

$$\{5P/32, 3PL/64, 27P/32, -9PL/64\}.$$

As we can see, this vector coincides with that obtained using the *consistent* approach; this is because the cubic shape functions represent the exact solution in the case where the displacements and rotations of the ends of the beam element are set. This case is an exception, and is valid only for beam elements. The proof that the identity does not occur by chance follows from the application of the principle of virtual work and by observing that the cubic shape function is the exact displacement field as a consequence of the displacement of the end nodes.

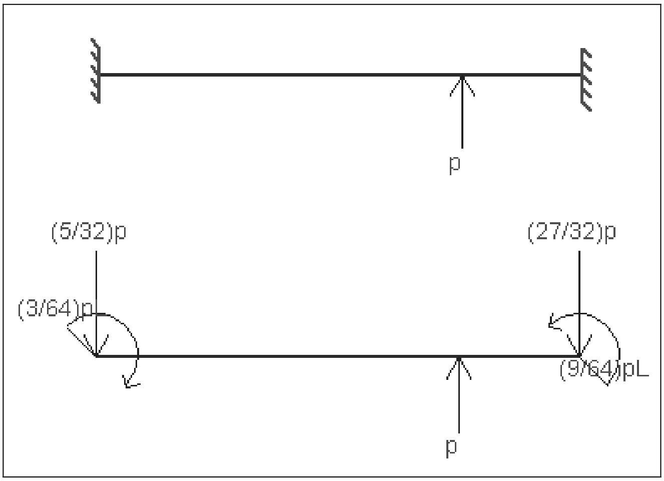


Figure 6-8: Constraint reactions opposite to equivalent nodal forces.

6.3.4 WHICH METHOD?

We have looked at two possible approaches for calculating equivalent nodal forces, but which method should be used?

The choice of which method to use to calculate equivalent nodal forces is usually made, not by the engineer, but rather by the software program. The engineer might go around what is predicted by the program and apply equivalent nodal forces evaluated with some criteria *in place of* the distributed loads, but if distributed loads are applied, then it is the program that calculates the equivalent nodal forces.

However, defining a load as distributed generally only means that we are asking the program to evaluate in our stead the equivalent nodal forces. Indeed, *the program always transforms distributed loads into equivalent nodal forces*, and usually once this is done the program forgets that “distributed loads” exist: the equivalent nodal forces are added to the nodal forces proper, the displacements are calculated, and from these the stresses are calculated, without any further evaluation of local effects.

There is an important exception to this methodology in the case of beams, where the local effects of distributed loads can be evaluated accurately. In this case during the phase of stress recovery it is possible to add the effects of distributed loads on the individual element, and evaluate the local component of the stress, thus arriving at a calculation of the overall stress that is quite precise. Not all software programs make use of this possibility; it is typical of specialised programs for calculating frames.

Analogous to what we have seen for stresses, knowing the exact deformation of the beam with two fixed ends under various distributed loads makes it possible to evaluate the local component of the displacement, which can be added to the global displacement predicted by the shape functions, finally arriving at an exact solution for the displacements as well. When we discuss the beam element we shall see better why it is possible in this case to arrive at exact results if the local effects are re-added, but here we need only say that not all programs make use of this possibility because it appertains only to the beam element and is not therefore typical.

The *lumped* approach is implicitly utilised when a software program transforms a pressure acting on a region of the model into many forces divided up between the nodes on the basis of the areas of influence (area loads). Examples of these kinds of load are wind loads, hydrostatic pressures, the action of pressure loads that are uniformly divided, and so forth. In these cases the speed with which the loads are generated often makes the *lumped* approach preferable to the *consistent* approach.

The *consistent* approach is the preferred method of software programs for evaluating the equivalent nodal forces, and is generally the one adopted, especially if the formulation of the finite element is in the displacements. However, the *lumped* approach is also frequently used to evaluate the effects of distributed loads in the elements: in this case it is implicitly taken for granted that the discretisation is sufficiently dense.

On the other hand, even though it is visually satisfying, it is wrong to use a *consistent* approach on a mesh that is not sufficiently dense. If, for example, given a square plate measuring 4 m. to a side modelled with only a few finite elements (say 3x3), it might appear satisfactory to have positioned a concentrated force right in the middle of one of the elements and far from the nodes, because the model appears to reflect the actual situation: the force is shown where it should be, and from a *pictorial* point of view things look right. However, if the dimensions of the elements are significant (that is, the mesh is not very dense), given that that force will in any case be converted by the software program into equivalent nodal forces, it is clear that the precision is purely apparent.

We will discuss the pitfalls of the *pictorial* approach to modelling finite elements in depth in chapter 12: the model should not *appear* correct; it has to *be* correct in light of the theory. Many inexperienced users and practitioners settle for a pictorial resemblance at the expense of effective precision.

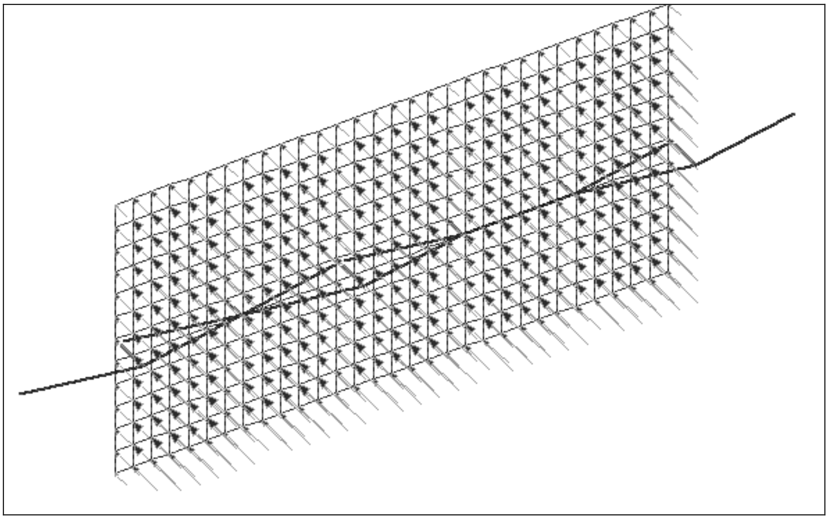


Figure 6-9: Use of area loads to simulate wind on a glass curtainwall supported by a tensile structure: the use of the *lumped* approach is implicit.

6.3.5 EXAMPLES

To illustrate these ideas let's first look at the example shown in fig. 6-10.

Let there be a beam of two meters clear span subject to a uniformly distributed load equal to 1kN/m.

In fig. 6-10a the beam is shown modelled with two finite beam elements, and the distributed loads are considered exactly by the software programs by means of the techniques used to re-add local effects discussed in the previous sections. The exact displacement at the mid-point is equal to 5.49. In this case the distributed loads were transformed into fixed end loads, and the final displacement is obtained by summing the "local" displacement found by calculating the contribution of the distributed load (see, for example, fig. 6-5) and that found with the shape functions (two cubics, one per element). The result is completely exact.

In fig. 6-10b the distributed loads have been transformed into equivalent nodal forces by evaluating the fixed end forces, as in the preceding case, but then, however, the local effects were not re-added. To simulate what would be done by a software program that does not re-add the local effects in a program that does re-add them, the loads were directly described as equivalent nodal forces and not as distributed loads. At the mid-point there is a force equal to (pL/2+pL/2)=pL, where L is the length of each element (L=1m, pL=1kN). The two couples that should be added to the central node have opposite directions and equal moduli and therefore cancel each other out. On the end nodes there is a force equal to pL/2=0.5kN, and a couple equal to pL^2/12=0.083$\underline{3}$kNm.

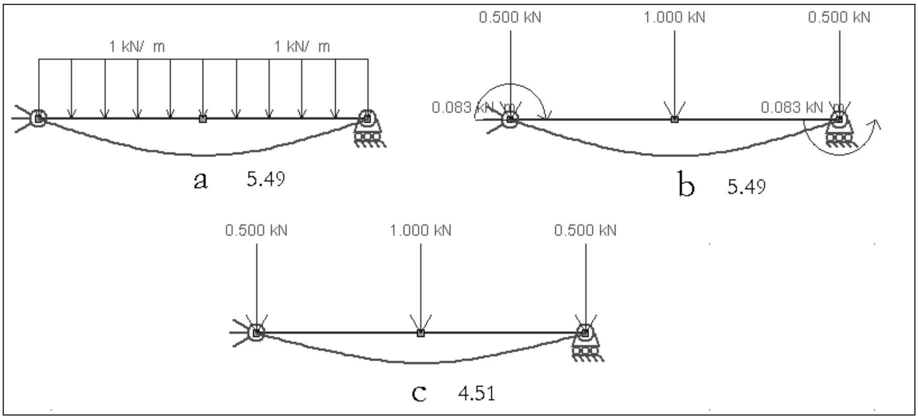


Figure 6-10: Example of the approach for fixed end forces, re-adding the local effects (a) and not re-adding the local effects (b), and of the *lumped* approach (c) in calculating the equivalent nodal forces (from the author's C.E.S.CO software program).

The displacement calculated at the mid-point is exact, and coincides with the theoretical value, *but intermediate displacements are not.* The program has used only the global part of the displacement, that is, only the shape functions (two cubics). Even if it cannot be seen visually, the deformation in 6-10a is different from that in 6-10b; in particular, the displacements (except at the mid-point, where there was a node) are less than the exact displacements. As far as the internal actions are concerned, those of 6-10b obviously do not appear to be equal to those of 6-10a. For example, the moment diagram is linear rather than parabolic, and the moment at the ends is not null.

In 6-10c, the distributed loads have been transformed into equivalent nodal forces using the *lumped* approach, that is, the load has been divided up between the nodes in proportion to the beam length of influence. There are no couples. The local effects of the distributed loads are not added to the final calculated results. The calculated value of the displacement at the mid-point is equal to only 4.51, and thus there is an error of 17.8%. The moment diagram is triangular.

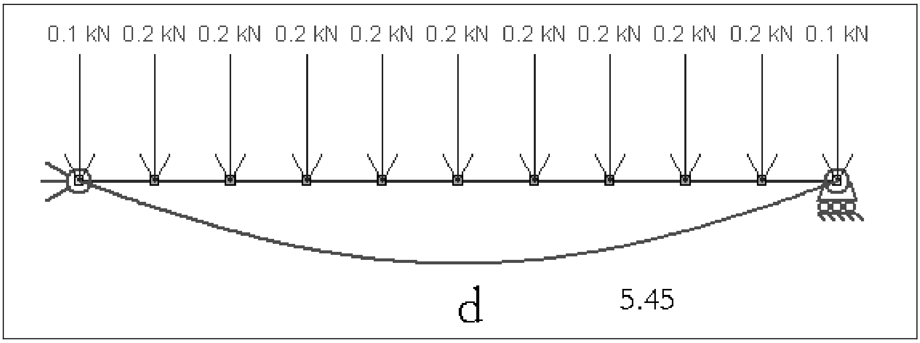


Figure 6-11: Lumped approach with a high level discretisation.

Fig. 6-11 shows what happens when the same *lumped* approach is used after the mesh is made much more dense (20 elements instead of 2). The calculated displacement is equal to 5.45, an error of only 0.7%.

This shows that the concept of "greater" or "lesser" discretisation has to be understood in relation to the particular problem being dealt with. Take, for example, the structure shown in fig. 6-12.

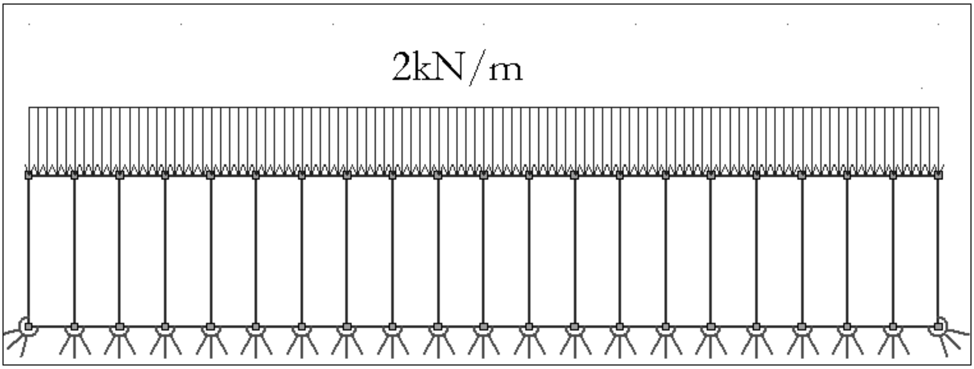


Figure 6-12: Structure partitioned by stiffeners under a distributed load.

A thin rigid element is stiffened by a series of vertical stiffeners placed close together. The purpose of the stiffeners is to diminish the clear span of the horizontal element, which will come to function as a kind of continuous beam with an elevated number of spans. Fig. 6-13 shows the bending moment diagram, which shows that not only can we expect to have a bending moment for each span, but a bending moment at the centre of each bay as well, so that the thin horizontal element has to be dimensioned in order to absorb these stresses.

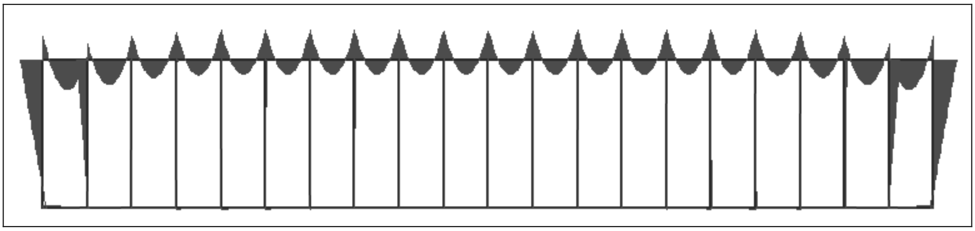


Figure 6-13: Bending moment field for the structure shown in fig. 6-12.

Now let's look at what might happen if the engineer decides that – given the close spacing of the stiffeners – it is only necessary to model each segment with a single finite element. Fig. 6-14 shows the model (made using the *lumped* approach). In this case we have to understand that almost all of the stress is due to the distributed load on the element and not to the movement of the nodes. *Thus if the software program does not readjust the local effects on the elements the result will be wrong.*

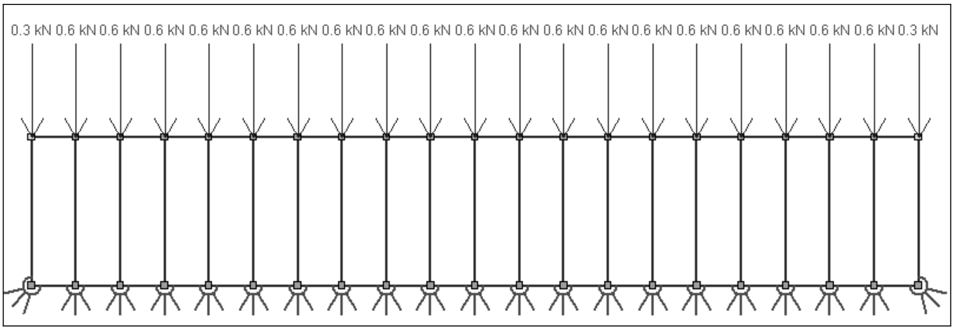


Figure 6-14: Insufficient discretisation.

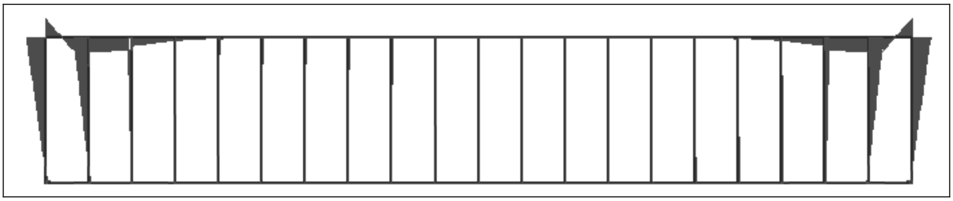


Figure 6-15. The effects of insufficient discretisation on the calculation of the stresses.

The result is shown in fig. 6-15: the stress situation is completely contorted, especially in the horizontal elements of the central area that are typically the most relevant. All of the bending actions of the typical bay have disappeared, the forces are transferred directly to the stiffeners, the bending moment regime typical of the problem has disappeared. An analysis made in this way might lead to a dangerous under-dimensioning of the elements, and possibly even a collapse of the structure as soon as it is erected.

Note that in a case like this one, even the use of the *consistent* approach or the fixed end forces might have led to the same problems. The problem typical of this structure cannot be correctly evaluated when taking into account only a single finite element on a typical bay *unless the software program re-adds the local effects in evaluating the stresses, which can occur only in certain programs and only for beam elements. If the problem had been relative to a plate it would have been necessary to increase the discretisation*, or lose a good part of the stress. In fact, no software program readjusts the local effects of distributed loads on plate elements[2]; this can only be hoped for with beam elements.

In order to evaluate whether or not a software program adds the effects of local loads to the stress and to the deformation *of beam elements* it is sufficient to create a model that includes a single finite element fixed at both ends subject to a distributed load. If the answer given is that shown in fig. 6-5, then the program readjusts the local effects; if instead the answer given is an undeformed beam with a null moment diagram, then the program does not readjust local effects. In this case it is necessary to adopt a rather dense discretisation, even when beam elements are used.

In the case of finite elements other than beams, software programs do not add local effects, and it is thus necessary to use meshes that are sufficiently dense in order to evaluate the stress and strain.

6.4 THERMAL LOADS

The presence of elements that are at a temperature that is different from the temperature of reference leads to the generation of a set of actions that act on the model in question so that the effects of these thermal variations can be implicitly evaluated. In physical terms, we can imagine that an element subject to a certain variation in temperature tends to expand (or contract), and that there is a set of forces applied to the nodes that impedes this. The containing forces exerted by the nodes over the element are immediately matched by the forces that the element transmits to the nodes, forces that will be added to the vector of equivalent nodal forces.

From a formal point of view it is possible to understand what takes place if we consider that the fundamental relation between the generalised stresses and generalised strains,

[2] This example is adapted from a case that actually happened. The horizontal element was a circular plate, stiffened by ribs in both directions. The error in modelling was made by a person expert in physics but inexpert in finite elements. The piece was constructed, and then failed when loaded.

$$\boldsymbol{\sigma}=\mathbf{D}\boldsymbol{\varepsilon},$$

has to be modified when the stresses are not entirely due to the stresses present in the structure but are partly due to non-elastic actions such as heat or cold. Denoting as $\boldsymbol{\varepsilon}_0$ the anelastic strains, that is, strains not caused by stresses, the relationship between the stresses and strains must be corrected to read

$$\boldsymbol{\sigma}=\mathbf{D}(\boldsymbol{\varepsilon}-\boldsymbol{\varepsilon}_0).$$

The elastic component of the strain is obtained as the difference between total strain $\boldsymbol{\varepsilon}$ and anelastic strain $\boldsymbol{\varepsilon}_0$.

It must be borne in mind that the anelastic strains $\boldsymbol{\varepsilon}_0$ are known *a priori* and thus are not new unknowns. If we retrace the various steps that we took, in chapter 4, to write the equilibrium conditions for the individual finite element, we can see that to the vector of equivalent nodal forces we have to add the term

$$\mathbf{f}_T\equiv(\int\mathbf{B}^T\mathbf{D}\ \boldsymbol{\varepsilon}_0 dV),$$

which quantifies the forces due to thermal effects so that, for the individual finite element, we must have

$$\mathbf{r_e}=\mathbf{K}_e\ \mathbf{u_e}\ -\ \mathbf{f_e}\ -\ \mathbf{f_T}$$

(cfr. 4-17).

The effects due to a change in temperature applied to an element can be quantified by adding a vector of nodal forces *equivalent to applied thermal loads* to the vector of equivalent nodal forces due to applied loads. This additional vector is assembled in the same way the other one was and contributes to the creation of the vector of known terms of the problem.

It should be noted that thermal actions are by no means negligible. They should always be taken into consideration, since they can generate significant stresses when the structure is not free to expand or contract. Take, for example, the case of a structure supporting a glass curtainwall which had significant problems because it was designed without taking thermal loads into consideration (fig. 6-16).

The analysis of the structure shows that each vertical and horizontal element (without any play between bolts and boltholes) transmitted to adjoining reinforced concrete beams a load amounting to 10.7 tons. An increase in temperature of only 15° C. was sufficient to generate a thrust equal to ten times the weight of an automobile.

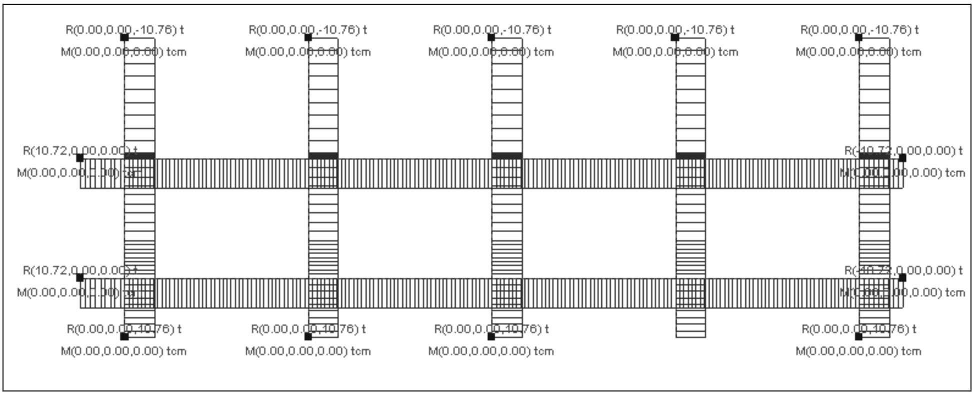


Figure 6-16: Axial actions and reactions on the edges of a frame whose temperature is increased by 15 °C.

6.5 APPLIED ACTIONS AND CONSTRAINT REACTIONS

Here we want to bring up an aspect that will be discussed in further detail in chapter 9, when we examine the checking of results.

Since the structure can be viewed ultimately as a single rigid body in equilibrium, it follows that the sum of all the actions applied to the structure has to be matched by the reactions exerted by the constraints on the structure. This fact can be used in two different ways, both to extreme advantage.

The first way is to use the constraint reactions to check that the solution is correct. Except for residuals, which should be modest, the reactions of constraints have to balance the applied loads: significant differences are never acceptable, and must lead to an in-depth search for the causes (these discrepancies can even be due to an error in entering the values for the loads, an error that can remain undiscovered for a number of reasons).

The second way, in programs that do not make it possible to know the resultant of the applied loads before solving, is to use the reactions of the constraints to evaluate the magnitude of the loads. If hundreds of actions are added it is possible to lose track of what has effectively been done. It is clear that in this way we have to take for granted that the program has correctly performed the calculation, although this is something that, strictly speaking, should never be taken for granted. For this reason the programs should always be equipped with tools for evaluating the resultant of applied loads *before* solving is performed and independent of it.

6.6 LOAD CASES AND COMBINATIONS

6.6.1 THE CONCEPT OF "LOAD CASE"

In performing structural calculations on a finite elements model we normally use the concept of "load case". By "load case" we mean the set of a certain number of actions applied in various ways at different points on the structure, presumed to be acting contemporaneously, in order to arrive at an overall vision of what takes place in the structure as a result of a common physical cause.

Even though it is possible to group the actions on a structure into arbitrary load cases, it is good practice to keep in the same load case only those actions that have a common origin or nature. This is because different physical causes are treated in different ways by the practices of checking. It is helpful to list some of the most common independent physical causes acting on a normal structure:

- Gravity loads (*GL*)
- Dead loads (*DL*)
- Live loads (*LL*)
- Maintenance
- Wind loads (*WL*)
- Snow loads (*SL*)
- Thermal loads (*TL*)
- Earthquake loads (*EL*)
- Constraint failure
- Malfunctions
- Impacts, explosions, attacks

Dead loads are usually independent from gravity loads on structural elements since gravity loads are normally subject to revision as the project evolves. It thus makes sense to keep gravity in its own load case, generally the first one, which can be annulled at will and redefined automatically.

Each physical cause can generate several load cases. For example, wind is often presumed to act along two orthogonal axes, but in some cases it is necessary to plan for other cases, either in order to evaluate the effects of wind acting along other axes, or in order to evaluate the effects of a change in direction along the same axis. In the case of wind, it is not always the case that

$$\text{Wind } -X = -\text{Wind } X$$

since the effect of the coefficients of exposure can vary depending on the direction the wind comes from.

Corresponding to each load case is a different vector of equivalent nodal forces, a different set of displacements, and a different set of stresses on the structure. In terms of the computational profile, the solution of a problem where different load cases are present involves a one-time only triangularisation of the stiffness matrix **K**, and the solution of as many systems of linear equations as there are load cases to be solved. To each load case there corresponds a vector of known terms **f** and a set of nodal displacements **u**, and to those, a given set of structural stresses $\boldsymbol{\sigma}$.

6.6.2 Normative combinations: limit states and Allowable stresses

Laws currently in force specify that the effects of various physical causes acting contemporaneously on a structure must be evaluated by means of the concept of load combination. By "load combination", what the norms are asking for is an evaluation of the contemporaneous presence in the structure of a certain number of elementary actions, each of which is suitably factored.

In the case of the norms regarding allowable stresses, it is generally up to the designer to determine the worst-case scenario in which the various elementary physical causes combine to create the greatest threat to the structure. In general, this leads to just a few combinations whose physical significance is immediately clear. The same cannot always be said for limit states, which sometimes require the application of algorithms that can lead to many hundreds of combinations, the physical implications of which are not immediately clear (for example, see the European standard Eurocode 0 - EN 1990). The idea is to generate a vast quantity of cases that are meaningful statistically.

In each of these combinations the elementary physical causes are weighted by a factor that takes into account the composite probability of a particular physical cause presenting itself in contemporary with another one that is assumed to be the basis for the combination in question. For example, if the base hypothesis assumes the presence of an exceptionally strong wind, then an exceptional amount of snow and an exceptional earthquake are considered after being appropriately weighted by a coefficient that takes into account the (im)probability that there will be contemporaneously an exceptionally strong wind *and* an exceptional amount of snow, an exceptionally strong wind *and* an exceptional earthquake, and so forth. The combination coefficient reduces the exceptional value to a frequent value and so should be seen as a "intensity corrective", and not as a linear combination coefficient. The normative combinations do not imply a *linear combination of effects*, but only the *contemporaneous presence of physical causes with different values*, each of which is obtained by weighting the exceptional values with reductive factors.

The practical consequence of this requirement varies a great deal depending on whether the structure behaves linearly or non-linearly. In the case of linear behaviour, it is possible to use the fundamental principle of the superposition of effects, but not in the case of non-linear behaviour.

6.6.3 THE PRINCIPLE OF SUPERPOSITION OF EFFECTS

In every load case there are presumed to be many – sometimes even several hundred – different individual actions in effect at the same time, and their overall effects are evaluated as a whole.

Even when the effects of the actions are not evaluated contemporaneously, it is possible to evaluate the overall contemporaraneous effects, independent of the order in which the actions are applied. This is possible because the conditions of elastic linearity permits the use of the principle of the superposition of effects, which says that the overall effect of a set of actions is obtained by summing the effects of individual, separately evaluated actions.

Thanks to this principle it is possible to evaluate the effects on the structure of a set of actions by simply summing the effects of individual actions. In particular, it is possible to evaluate the effect of a weighted sum of actions by weighting the effects of the individual actions in the same way. This leads directly to the concept of "load combination", which in terms of the FEM, takes on the particular meaning of "weighted sum of the effects of a given number of elementary load cases".

The principle of the superposition of effects is not valid when the problem is non-linear. Thus it is not valid when the equilibrium equations have to be written with respect to the deformed configuration ("external" geometric non-linearity, physically large displacements), nor when the deformations have to be written to include the second-order terms ("internal" geometric non-linearity, geometrically large displacements), and not even in the simplest case of "tension only" rods (rods that are always only in tension; if not, they are eliminated from the structure), or of monolateral constraints.

In all of these cases, the contemporaneous effect of a certain set of actions arising from different physical causes (the "combinations" specified in the norms) has to be obtained by means of a single load case that contains all of the elementary actions appropriately combined, or more properly, by means of a genuine reconstruction of the load path taken in order to arrive at the desired contemporaneousness of actions. Given that there can be a high number of normative combinations, it is easy to understand how in non-linear structures the amount of "load cases" increases drastically.

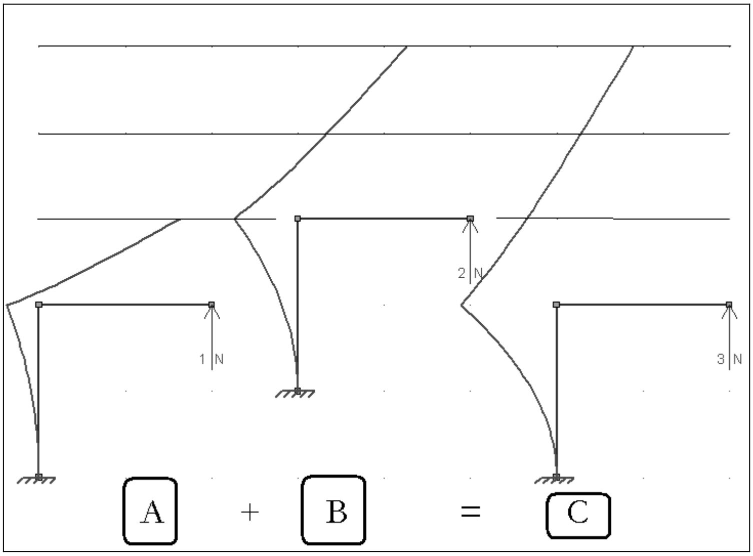


Figure 6-17: Effects of linearity: the superposition of effects.

Instead, when the problem is linear, it is possible to solve the load cases separately, and then evaluate the effects of the normative "combination" by means of a genuine linear combination of the effects foreseen for the individual cases. *Obviously, this kind of "combination" can occur at any time after solving, since the effects of the elementary load cases are by then known.*

The most recent software programs make it quite easy to evaluate the combined effects, by simply defining the weights to be assigned to each elementary load case. In general, the combination does not necessarily require writing "combination" files to disk, since for problems of a normal dimension the speed of the processors is such that the operations can be performed in real time.

6.7 EQUIVALENT STATIC ANALYSIS (EARTHQUAKE)

6.7.1 OVERVIEW OF THE PROBLEM

In order to present a simplified view of the effects of an earthquake on a structure a simplified system of calculation called "equivalent static analysis", is often used. We will briefly discuss it here.

Equivalent static analysis can be used when treating structures with a high degree of structural regularity in plan as well as in elevation in terms of both stiffness and mass. Given these hypotheses we can expect that the structure has a dominant way of vibrating, one whose form can be roughly represented by a straight line deformed configuration.

Each mass present in the structure is found to generate a d'Alembert's force (an "apparent" or fictitious force) equal to the value of the mass itself times the peak acceleration expected in the course of oscillatory motion. On generic mass *i*, m_i, will act a force F_i equal to

$$F_i = k m_i g h_i (\sum m_l) / (\sum m_l h_l)$$

where g is the acceleration of gravity, equal to 9.81m/sec^2; k is the reference acceleration in unit g (a pure number). This reference acceleration is evaluated with the response spectrum method; h_i is the height of the generic mass m_i with respect to the top of the foundations.

An examination of the formula, which varies slightly in standards around the world, makes it possible for us to make the following observations.

Each mass is subject to a force equal to its weight ($m_i g$) suitably multiplied by two factors. The first factor is k, which indicates the intensity of the reference acceleration. When the response spectrum is known for an earthquake expected in the area where the structure is found, we see that a simple oscillator has a maximum response that depends on the period with which it oscillates. Generally, the structure oscillates in many modes, but of all these only the dominant mode is considered (that which effectively predominates over all the other modes). The period of the dominant mode makes it possible to read the peak acceleration expected on the curve of the response spectrum. For example, if in a given location the peak ground acceleration for an earthquake with a return period of 2000 years is 0.1g, and the structure oscillates in the dominant mode with T=0.8 sec, the peak acceleration can be read as 0.25g. In that case, k=0.25.

The second factor is called the distribution coefficient; we indicate it here with the letter γ

$$\gamma_i = h_i (\sum m_l) / (\sum m_l h_l)$$

where the summations are with respect to the index "l".

Given the same earthquake and structure (total mass and mass distribution), the force grows linearly with the mass and with the height. The greater the height, the greater the force; the greater the mass, the greater the force.

Thus for the application of the method of equivalent static analysis, the correct evaluation of both the size of the masses and of their distribution is fundamental.

6.7.2 CONVERSION OF ACTIONS INTO MASSES

In order to be able to generate automatically the set of forces related to equivalent static analysis for earthquakes we must have information about the position and intensity of the masses present in the structure. There is one particularly satisfying way to generate the masses which maintains both their distribution and their intensity, both determining factors for obtaining a realistic analysis. In most cases there is a precise correspondence between the masses present in the model and the static actions. Many static actions are effectively tied to the presence of masses in the structure. This is certainly the case for all members of the important family of actions related to gravity, while it is not true for the actions related to wind, actions related to temperature variations, and actions related to structural settling.

In general the masses necessary for evaluating seismic effects are obtained with a step-by-step procedure starting with the applied actions in cases when there are non-seismic loads. Each load case is investigated in order to establish whether or not it corresponds to the masses. There are three possible sets: the set of load cases that generate masses that correspond to the sum total of applied loads; the set of load cases that do not generate any masses at all; the set of load cases that generate masses that correspond to a part of the applied loads.

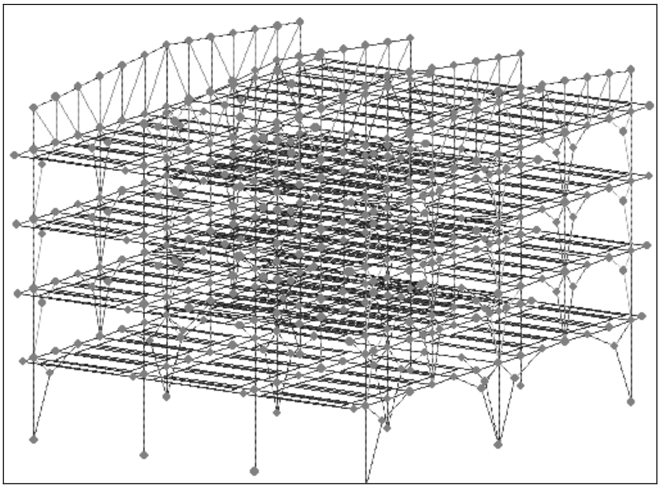


Figure 6-18: Model of a building. The nodal masses are shown as solid circles. These masses were generated automatically starting from the applied static loads.

The dead loads due to the presence of weights are part of the first set; loads related to wind and temperature belong to the second set; the live loads, or those due to snow, generally belong to the third set. In this last case, it is necessary to evaluate the partial quota of mass present at the moment of the earthquake. The existing norms generally have different rules for evaluating how much of variable mass is present at the moment of the earthquake. In any case, the corresponding masses are generated by appropriately factoring the load case in question. The resulting masses are scaled with respect to the overall masses with an analogous factor.

Note that it is possible to generate the masses we are interested in, those which must be attributed to each node, with a "consistent" or "lumped" approach like the ones seen earlier for static actions. In this case, the rotational part of the mass will be effectively associated to its moment of inertia with respect to the axis of rotation of the node to which it is attributed. In equivalent static analysis the moments of inertia are not taken into consideration.

Generally speaking, the "lumped" approach is the one used. Masses on the nodes are made to correspond to the forces applied directly to the node. Masses corresponding to the resultant, appropriately distributed among the nodes of the element, are made to correspond to the distributed loads. When the mesh is sufficiently dense, the effect related to lumping, as we have already seen in the context of statics, tends to be negligible.

6.7.3 EQUIVALENT STATIC ACTIONS

If the masses were generated starting from the applied loads and with the same degree of precision, assuming that the structure effectively has a dominant vibration mode that is more or less linear, we can expect a quite realistic distribution of masses. This has the immediate effect of generating a realistic distribution of forces on the structure, which has at least two advantages. The first is that all the zones of the structure will be appropriately solicited, without there being any zones that are unrealistically without suitable forces.

The second advantage is that the resultant of the seismic forces can be evaluated exactly *without necessitating any manual calculation aimed at identifying the centre of the masses*. The elementary forces will automatically produce a set of actions consistent with the masses in the structure, without requiring separate evaluation of those effects.

In this regard, note that when the structure is highly symmetrical, we should always presume that there will be conditions of asymmetric masses for at least the variable part, so that an appropriate accidental eccentricity between the two barycentres (masses and stiffness) can be generated.

Today seismic analyses are usually performed by modal analysis, which evaluates the acting forces more correctly than equivalent static analysis does. However, what we have said about the transformation of forces into masses still holds, because a realistic distribution of masses is required in order to perform modal analysis correctly.

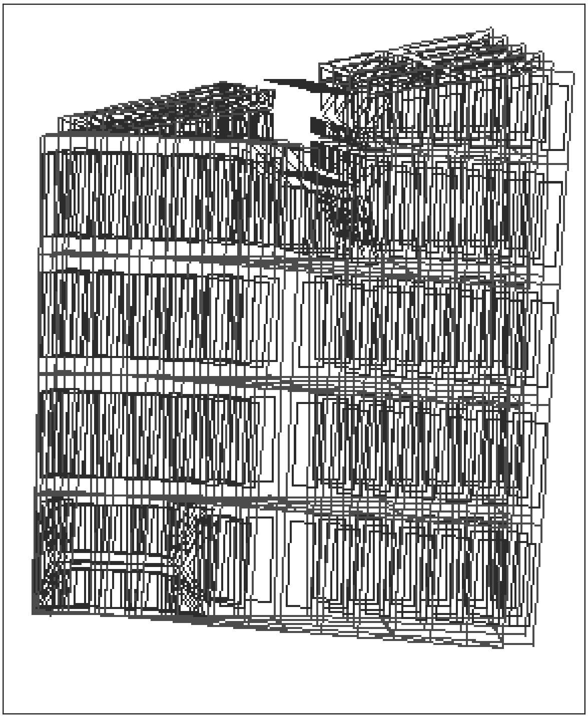


Figure 6-19: Rotational effect of a building evaluated with equivalent static analysis (deformed configuration under an earthquake in direction X, seen from above). There is an eccentricity between the centre of the masses and the centre of stiffness. The centre of the masses is automatically evaluated thanks to the addition of numerous elementary forces distributed with the same accuracy as the masses where they originated.

7 CONSTRAINTS

7.1 INTRODUCTION

We have seen that after assembly the solving system turns out to be unsolvable, either because of rigid motions or inactive degrees of freedom within the structure. In both cases, the determinant of the square matrix becomes null, which makes it operatively impossible to solve the problem.

Although both of these circumstances lead to the same result, that is, they cause the determinant of the stiffness matrix to become null, there are actually many different reasons for why this happens.

In the first case, that of possible rigid motions, the structure can effectively be subject to rigid motions which can involve either part or all of the structure itself. Without suitable constraints, the structure can move without developing any strain energy. In this case it is either *hypostatic* or *labile*. A structure is hypostatic when the number of constrains is objectively insufficient to keep it immobile: this is the case when a solid has fewer than six constraints, for example. A structure is labile when, even though in theory there is a sufficient number of constraints to prevent rigid motions, they still do not eliminate the possibility of the structure's moving because they are not independent of each other: the classic example is an arch with three aligned hinges. We will be referring to both of these situations when we speak of "mechanism".

In the second case, that of inactive degrees of freedom, there are no actual movements associated with the structure's degrees of freedom, but there is a (free) movement of a node. If we consider the node as a minute rigid body, then these motions effectively constitute "an act of rigid motion". Actually, there is no real movement of the structure associated with these movements, and thus we speak of "pseudo-mechanism".

In order to eliminate actual rigid motions it is necessary to set *boundary conditions*, that is, to impose that the displacements by the structure's boundary are all equal to certain predetermined values (which are generally null). This will be discussed in the next section about "real constraints".

In order to eliminate the inactive degrees of freedom it is necessary to impose that such degrees of freedom are not part of the problem, or, if they are part of the general problem, then it has to be clearly stated that they are identically null. The techniques used to achieve this will be discussed in the sections that follow about ad hoc formulations (§7.3) and fictitious constraints (§7.4).

Structural analysis with finite elements
978-0-7277-4093-9

7.2 Real Constraints

7.2.1 Overview

In general, in order to function correctly structures need to be constrained in some way. The presence of constraints makes it so that the applied loads and actions force the structure to deform in such a way that a suitable quantity of strain energy is stored. All structures are composed of solids, and their constraint takes place by establishing that the displacements of their boundaries assume certain values. In the context of the FEM, however, structures may resemble domains that might not be volumes: it is thus necessary to translate the components of constraint of the real structure into the corresponding components of constraint of the ideal model being used. Thus, for instance, a cantilever clamped into a wall can be modelled by imposing that the translation and the rotation of the end face of the solid are null. A beam or a slab that is simply supported will have their translations blocked, but their rotations with respect to the end point of the axis line (in the case of the beam) or along the boundary considered at the median plane (in the case of a slab) will be free.

This first observation, related to the necessity of having to translate the conditions of real constraints into the formulation being used, is followed by another one: the displacement field of the boundary of the model follows the pre-established rules defined by each finite element that that particular boundary simulates, and *it is therefore completely dependent on the displacements of appropriate nodes of the model according to the different possible shape functions used by the various elements.* Take, for example, the (part of) a structure shown in fig. 7-1, which simulates the end part of a wall embedded into the ground, modelled in plane stress. If we presume that the system of reference has axes X and Y, we can be certain that what counts are the displacements (u, v) of the nodes, since the other components of displacement remain outside of the formulation that we are using. We want to impose that the entire boundary corresponding to nodes 7, 11, 12, 13 and 8 be completely fixed.

If we consider the first element on the lower left, we want to set that all the points corresponding to segment 7-11 are immobile. In order to do this we need only make nodes 7 and 11 immobile. Actually, for the displacements u and v on segment 7-11 the equations related to the shape functions used for the element are valid:

$$u(\lambda)= u_7+ (u_{11}-u_7)\lambda$$

$$v(\lambda)= v_7+ (v_{11}-v_7)\lambda$$

where λ is a non-dimensional abscissa comprised between 0 and 1. If we set

$$u_7= u_{11}= v_7= v_{11}=0,$$

then for each λ we must have

$$u(\lambda)=v(\lambda)=0.$$

Setting the boundary conditions therefore translates into making it so that the displacements of a suitable set of nodes assume certain values (generally 0). Because of the way itself in which the elements are formulated, these conditions *on the nodes* are actually translated into conditions *on all the points* belonging to the boundary influenced by the intended constraints.

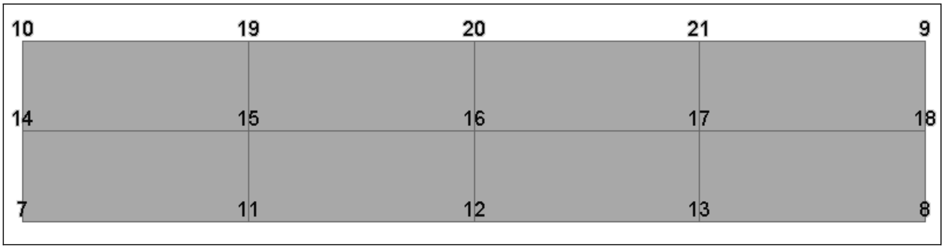


Figure 7-1: The end part of a vertical wall modelled as plane stress.

In order to constrain the side corresponding to nodes 7, 11, 12, 13 and 8 it is necessary to constrain all the nodes of the group (7, 11, 12, 13, 8) without exceptions. Actually, if there were an unconstrained node, the constraint would be only partial and not complete: fig. 7-2 shows clearly what would happen if, for example, nodes 12 and 13 were not constrained: the displacements of those two nodes do not satisfy the necessary conditions of congruity.

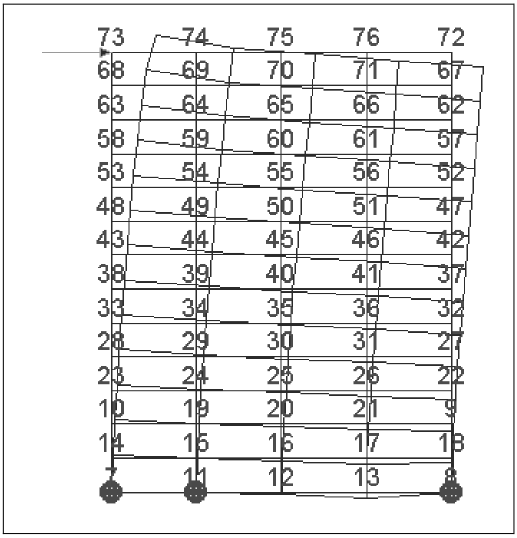


Figure 7-2: Partial constraint of the nodes of the side (nodes 12 and 13 are not constrained).

The lack of appropriate constraints is not always evident when examining only the deformed configuration of the structure. Fig. 7-3 shows the same structure shown in fig. 7-2 modelled more finely, "forgetting" to constrain a node. No problems are evident when the deformed configuration is examined, but an examination of the maximum principal stress clearly shows a suspicious discontinuity.

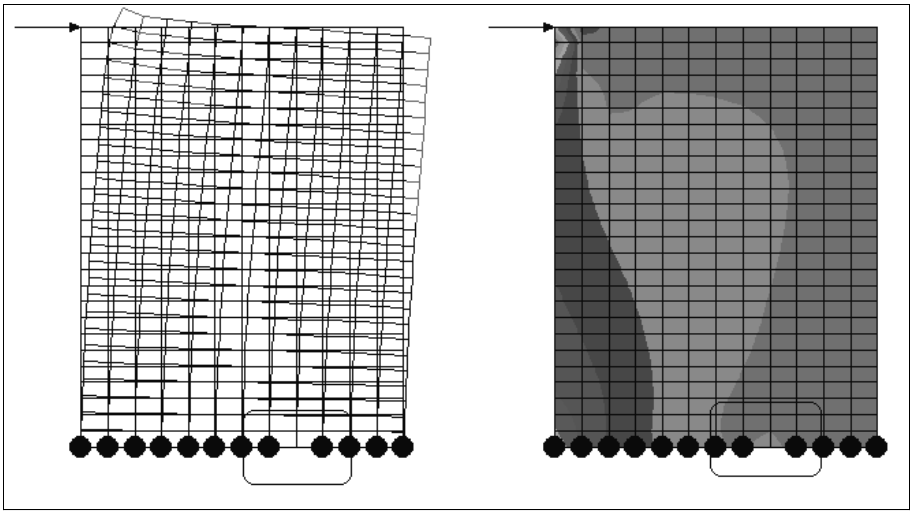


Figure 7-3: The effects of partial constraint on a structure modelled more finely.

It is very important that setting the boundary conditions does not eliminate all possibility of the structure's moving, a situation that can only lead to an erroneous solution. If we take, for example, a beam clamped at both ends modelled with a single finite element, we can see that the boundary conditions have been set so as to eliminate any possible movement of the structure. In cases like this, generally related to a coarse discretisation, it is necessary to increase the number of elements so that free intermediate nodes appear. If not, an identically null solution is found for no apparent reason. In effect, in this example, the only cubic function that respects the boundary conditions is a straight line coincident with the axis of the beam, that is, with the undeformed structure.

7.2.2 FIXED CONSTRAINTS

"Fixed" constraints are those that call for a null displacement, while constraints that call for a known displacement not equal to zero are called "settles". If the displacements whose values are known (null or non-null) are parallel to the structure's axes of reference, then the constraints are *orthogonal*. If the displacements whose values are known are inclined with respect to the reference axes, then the constraints are *skew*.

Fixed orthogonal constraints do not pose any particular problems for the engineer: it is always possible to assign a null value to the "orthogonal" degrees of freedom by simply imposing constraints on the components of displacement of the nodes that correspond to the degrees of freedom to be made null. Actually, there are generally six degrees of freedom associated to each node, all of which can be either free (in which case their value is unknown), or constrained (in which case their value is null when the constraints are fixed). Constraints are usually assigned to the nodes by selecting particular ones and specifying for them which degrees of freedom are fixed. In the 1970s and 1980s when the alphanumeric software programs were just being developed, it was necessary to provide an appropriate series of constraint codes for each node, essentially "0" and "1".

In contrast, fixed skew constraints do pose some problems because the low- and medium-range software programs for finite elements do not offer the option of explicitly applying skew constraints. From an analytical point of view a skew constraint on a node is simply an equation of this type:

$$\sum A_{ij} u_j = 0,$$

that is, a linear relation between unknown displacements. If, for example, in a plane problem in plane *xy* (unknown displacements, *u* and *v*) the translation in the direction normal to the straight line *y*=*x* is constrained along a certain row of nodes, we need only write the obvious condition

$$u - v = 0$$

in all the nodes that are part of that row. This problem comes up in all the software programs that do not make it possible to apply constraints that are not orthogonal. How can this be resolved?

The solution consists in adding very stiff elements that are capable of forcing the unknown displacements to respect the required conditions of skew constraint.

For example, fig. 7-4 shows that the points on the left cathetus have been forcibly prevented from translating in the direction perpendicular to that cathetus by means of the addition of very stiff springs arranged normal to the cathetus. Note that it is essential that the springs be sufficiently stiff to result in displacements so small that they can be considered null (in the example shown here the displacements *u* and *v* are equal to at least the seventh significant digit).

Both stiff springs and rod elements either made of stiff material or with a very large area can be used to introduce skew constraints. Both of these make it possible to model the desired kinematic behaviour.

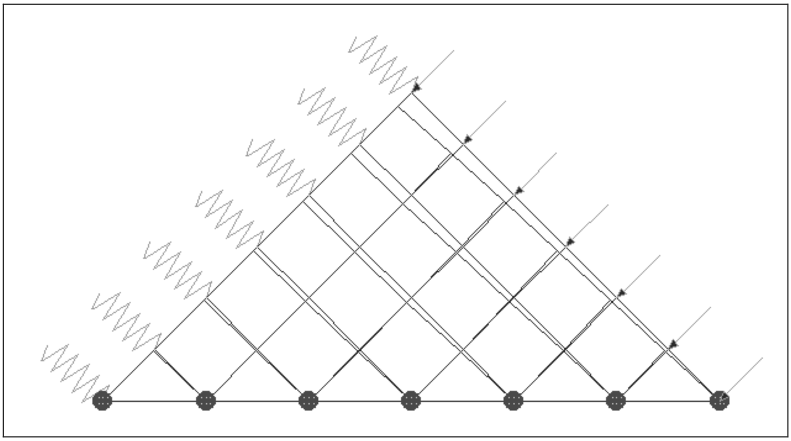


Figure 7-4: Setting skew constraints thanks to the addition of rigid springs.

7.2.3 SETTLING CONSTRAINTS

7.2.3.1 ASSIGNED SETTLEMENTS

It is possible that some points of the structure are forced to move in a predefined way as a result of causes external to the structure in question. This is the classic case of settlement in foundations or other parts of the structure in contact with external agents such as other structures, which "push" on the given structure, imposing a certain set of displacements. It is also correct to speak of "constraints" in this case, even though they do not impose any null displacements.

From a strictly mathematical point of view, the presence of this kind of contraint translates into a certain number of equations of this type:

$$u_i\text{=constant}$$

where u_i is one of the nodal displacements that was originally unknown. From a numerical point of view, in software programs currently in use, the condition just written can be set both directly and indirectly. If this condition is set directly, the program has to eliminate the corresponding equation (eliminating a row and a column of the stiffness matrix) and add a certain number of fictitious forces to the vector of equivalent loads. Actually, the generic equation on the generic *j*th row of the system, which takes this form:

$$K_{j1}u_1 + K_{j2}u_2 + \quad + K_{ji}\underline{u}_i + \ldots. = f_j$$

can be rewritten

$$K_{j1}u_1 + K_{j2}u_2 + \ldots. = f_j - K_{ji}\underline{u}_i,$$

that is, the terms related to the displacements, whose (non-null) value $\underline{u}_i$ are known *a priori*, can be taken to the second member and added to the vector of known terms. Even if this operation is extremely clear from an analytical point of view, from a computational point of view it might not be efficient. For example, the fact that different settlements can be applied in different load cases might lead to matrices that are different for the various load cases, which is quite inefficient computationally.

In order to overcome this, an indirect system is often adopted to apply the desired settle, that is, applying a spring of elevated stiffness R in the precise direction of the settle, and then assigning, in the load case involving that settle, a force of an appropriate value. In this method the row and column relative to the degree of freedom involved are not eliminated. The *i*th row corresponding to the degree of freedom whose value is known, which was originally given as

$$K_{i1}u_1 + K_{i2}u_2 + \quad + K_{ii}u_i + \ldots. = 0,$$

with the addition of a very high value of stiffness R and an appropriate force F becomes,

$$K_{i1}u_1 + K_{i2}u_2 + \quad + (K_{ii}+R)u_i + \ldots. = F.$$

Given that R is very large with respect to K_{ii}, the equation is *numerically* equivalent to

$$Ru_i = F.$$

If we are careful to set F= $R\underline{u_i}$, that is, to apply a force F calculated by multiplying the elevated stiffness R by the displacement $\underline{u_i}$, the equation above becomes

$$u_i = \underline{u_i}.$$

This method is very convenient, because it makes it possible to not have to manipulate the stiffness matrix in different ways from load case to load case. Moreover, in other load cases, those in which there is no settle, since no force F has been applied, but where there is still a stiffness R related to the spring, equations of this type are obtained:

$$Ru_i = 0,$$

which are equivalent to imposing null displacements (and thus non-settling constraints).

The only practical problem in applying this method is deciding what value to give stiffness R. Actually, a too-low value for stiffness R might mean that the added stiffness cannot predominate over existing stiffness, while a too-high value can give rise to numeric problems in the solution of the solving system.

One efficient method for determining what stiffness R to apply to the spring is the following. If K is the maximum stiffness existing in the model, then R can be set such that $R=(10^5 \div 10^6)K$, that is, applying to the spring whose stiffness is equal to a hundred thousand, a million times the maximum stiffness of the structure. In an ordinary frame structure, it is easy to determine K since the maximum stiffness is generally tied to the axial behaviour of the columns. We can thus set

$$K= EA/L,$$

where E is the modulus of elasticity, A the area and L the length of the stiffest column in the model. Note that what counts is not the mantissa but the exponent: the order of magnitude is enough as far as simplicity is concerned. If, for example, it turns out that EA/L=6.57989e7, we set $R=1.0(10^{12}\div 10^{13})$. Note that when K cannot be determined with precision, or in order to exclude numerical instability, a couple of trial-and-error calculations can be made in order to determine how sensitive the solution is with respect to this parameter.

7.2.3.2 Assigned stiffnesses (soil-structure interaction, Winkler foundation)

One particular case of not fixed constraints are those in which the amount of the displacement is unknown *a priori*, while the stiffness provided by the soil, presumed to behave in a way that is linearly elastic, is known. This approach is often used when a simplified way is needed to determine the effects of interaction between the soil and the structure. Although it may appear overly optimistic to presume that the behaviour of the soil is linearly elastic,[1] it should be remembered that the principal objective of this kind of investigation is to determine the extent of the possible settlement at least approximately, in order to be able to estimate the order of magnitude of the stresses that such settling places on the structure.

We can distinguish between two cases: the case of point foundations (plinths) and that of continuous foundations (reverse beams, slabs).

In the case of point foundations, in correspondence to the nodes where joints are ordinarily placed, springs of a suitable stiffness are placed instead, calibrated by means of formulas that take into account the dimensions of the foundation, its shape, the type of soil, its state of consolidation, and even the static or dynamic context in which the investigation is taking place. Six springs can be placed in correspondence to a foundation plinth: three translational springs, and three rotational springs for the three possible rotations of the plinth about the axes parallel to the axes of reference.

The degree of the stiffnesses is usually such that the displacements that are measured in the model are quite small. The advantage of this approach over that which involves setting particular displacements (or settlements) in one or more load conditions, is that the displacements of the foundations will be to a certain

[1] Anyone with even a vague idea about soil behaviour knows quite well that practically anything can be presumed *except* that it is linearly elastic. Nevertheless, the hypothesis is used in a technical context in order to avoid calculations that would be too complex and costly otherwise.

extent related to the different load configurations presumed for the structure, and thus more in keeping with them.

In the case of continuous foundations, putting a few springs in suitable directions is not sufficient: the surface of contact between structure and soil is not negligible and, unless more complex systems that explicitly model the soil are employed, it is necessary to use the hypothesis of a "Winkler foundation". In this scheme, the foundation is modelled thanks to a bed of springs that have a given appropriate stiffness. If k is Winkler's constant (for example, k=2Kgf/cm^3), then the stiffness of each spring to be placed is obtained by multiplying the given k by the surface of each node's influence. The operations that have to be performed differ slightly depending on whether there is a slab or beam foundation.

In the case of a slab, the surface S pertaining to the given node is determined by taking into account the finite elements of the slab attached to that node. The stiffness of each spring will be kS.

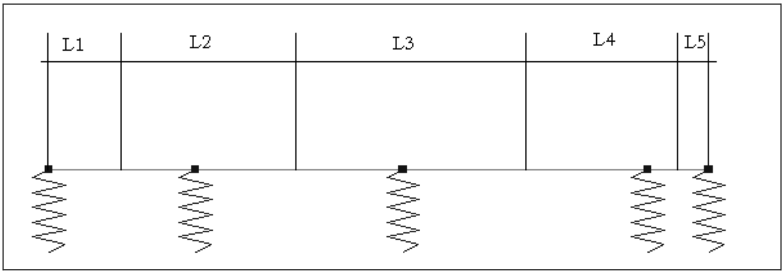


Figure 7-5: Example of calculating the length of influence for various nodes (beam foundation).

In the case of beam foundations, if the beam has width B, and L is the segment of the beam pertaining to the given node, the stiffness of the spring to be placed on the node is kBL. Note that if the elements are not all equal (as usually happens because of meshing requirements), the various springs will have different stiffnesses.

Some software programs (for example, Sargon, the author's own program) make it possible to calculate automatically the stiffness to put on each node in a group of selected nodes, simply starting from the Winkler constant "k" for slabs and from the product "kB" for beams. In this case the program itself orders the nodes and determines, for each node, S in the case of a slab or L if for a beam, and applies the appropriate spring automatically. If, as should always be the case, the mesh is sufficiently dense, this approach is not any more difficult and leads to results that are very close to the "exact"[2] ones, obtained by means of more

[2] Given that the Winkler formulation already involves significant approximations, it seems appropriate to maintain "exact" in quotation marks.

refined finite elements (beams on Winkler's soil or plates on Winkler's soil, both also available in Sargon) formulated so that they contain equilibrium equations that are modified to take the resistance provided by the soil into account.

Element number	Maximum moment	Average displacement	% Deviation of moment from exact moment in the case of 32 elements
4	1.605e6	3.195e-1	6.03%
8	1.684e6	3.169e-1	1.41%
16	1.703e6	3.160e-1	0.29%
32	1.708e6	3.158e-1	-

Table 7-1. Response variation with mesh density

For example, table 7-1 shows the series of values obtained by doubling the number of elements in each of the three bays of the problem shown in fig. 7-6. It can be seen that already by increasing the number of finite elements in the bay from 4 to 8 the value of deviation of the moment from the "exact" value (assumed to be that obtained when there are 32 elements) has decreased from 6.03% to only 1.41%.

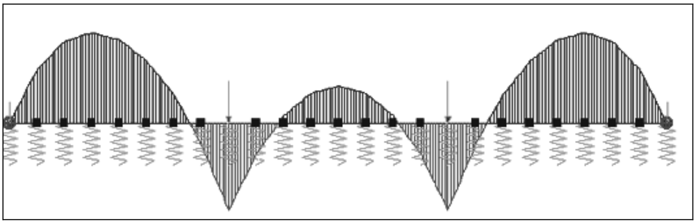


Figure 7-6: Beam on Winkler's soil (cfr. table 7-1).

7.2.4 KINEMATIC CONSTRAINTS (MULTIPOINT CONSTRAINT)

A generalisation of the preceding type of constraints leads to consideration of the case in which there necessarily exists some kinematic constraint among *various* degrees of freedom of the model *in relation to each other* (multipoint constraint, or MPC, in the language of the FEM). This situation occurs, for example, when we want the various nodes belonging to a floor of a building to move in such a way so that the floor remains undeformed, and is thus made to translate and rotate as a single body. A similar situation occurs when we want the distance that originally separated two nodes to remain unaltered, even when the nodes move (*rigid link*), and so forth.

In this case, as in the previous one, it is worthwhile to note that there are different techniques for simulating the desired kinematic conditions. Some software programs use techniques that are more evolved, making use, for example, of Lagrange multipliers, while others do not offer explicit tools for setting these conditions, but only implicit tools.

In effect, it is generally possible to simulate the desired kinematic conditions by means of the addition of a certain number of elements of a high degree of stiffness that impose, through their indeformability, the required constraint. If we want to ensure that the distance between two nodes remains constant, we need only add a "stiff" rod that joins them. If we want a floor to move as a rigid body, we need only add a sufficient number of "rigid" membrane elements in order to impose the undeformability of the floor.

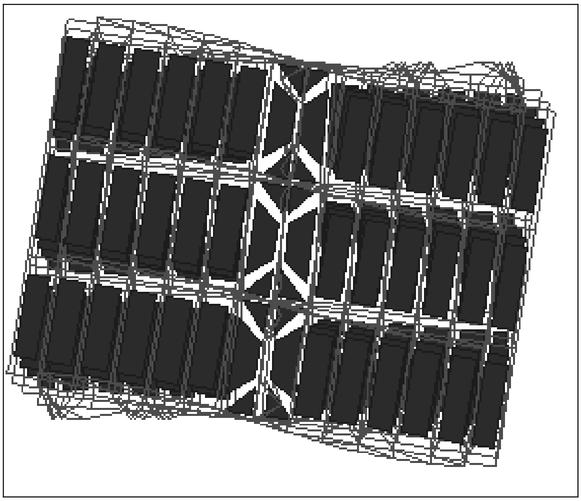


Figure 7-7: Deformed configuration in twisting of a building viewed from above: here the MPC is implicitly related to the presence of the membrane elements on the floors.

In some cases the requirement that there be an MPC is a conditioned reflex tied to outdated techniques of calculation. For example, in the case of building floors, there is no reason why they have to be necessarily modelled as a "rigid" plane today, when it is clear that no floor is like that. Instead, years ago, when the number of degrees of freedom was a problem, setting such a condition could mean saving a significant number of degrees of freedom, and so it made sense to do so. Today we can solve systems of thousands of degrees of freedom, and it is much more correct to model the floors of a building with appropriate two-dimensional elements, leaving it up to the model to keep track of the effects that derive from that.[3]

[3] In the context of earthquakes as well, the hypothesis of a rigid plane and the manual calculation of the mass centre and the stiffnesses no longer appears to be necessary, except to check the results of the analysis. By setting the masses on the model where they are effectively found, thanks to there being many nodes available,

7.2.5 LABILITY AND HYPOSTATICITY (MECHANISMS)

7.2.5.1 TRUE MECHANISMS

The problem of lability and structural hypostaticity, or static indeterminacy, is generally related to errors during the design phase, so that the structure, rather than being rigorously free of the possibility of rigid motions (mechanisms), is still capable of possible movements that can occur without the expenditure of any work.

The presence of hypostaticity can be made glaringly evident in the model by an abrupt halt in the procedure of solving, a halt due to the singularity of the stiffness matrix. Unfortunately, as we shall see, there are solvers which, through an exaggeration of the concept of "user-friendliness" go forward with the calculation of stresses and displacements without signally any problems.

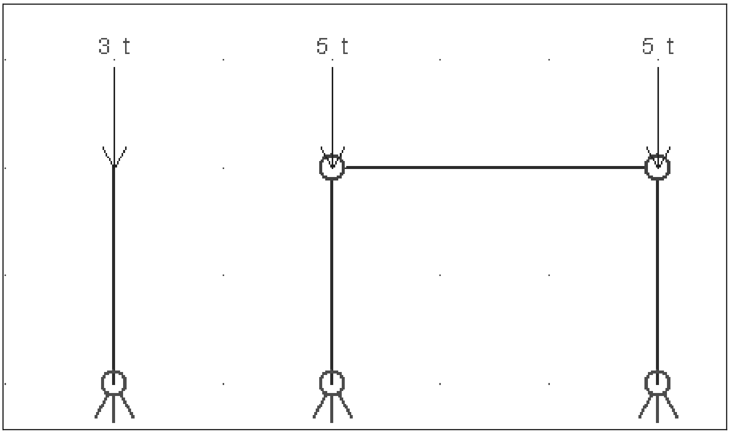


Figure 7-8: A statically indeterminate structure with unactivated mechanism.

First we have to distinguish between *numerical* instances and *design* instances. From the numerical point of view a system has a mechanism when the determinant of the stiffness matrix is null. From a design point of view a system has a mechanism when either there are not sufficient constraints to restrain all the rigid motions of a structure, or when, even though these constraints exist, they are not capable of completely restraining the rigid motions of the structure. For example, if we were to construct the structures shown in fig. 7-8, they could be made without placing an actual hinge at the base, but with a constraint whose stiffness was so inconsequent that it could be considered like a hinge: for instance, a thin base plate without ribs, connected only at the centre, etc. Thus,

and using the available techniques in appropriate ways, it is possible to obtain results that are much more precise than those that can be obtained by presuming a rigid plane.

the static indeterminacy is established in the model, while in the structure the static indeterminacy is actually tied to the impossibility of the constraint at the base to guarantee congruity. Because of this it might not be immediately apparent (as is often the case) but becomes obvious only after a certain period of time, when chance actions lead to deformations or even failures within the structure itself.

Note that in the structures shown in fig. 7-8 the static indeterminacy is not activated because the forces are perfectly vertical, and thus in theory it is possible to do away with the equations of horizontal equilibrium and consider only the vertical equilibrium of the nodes, thus arriving at a solution that is acceptable in terms of displacements, stresses and strains. Some software programs behave like this: they see the mechanism, but since it is not activated by any force, they go ahead.

This kind of behaviour is extremely dangerous because it can lead to problems as severe as actual collapse. It should always be remembered that:

1. No force in nature is actually "vertical";
2. No member can ever be made perfectly straight;
3. Mechanisms that are not activated in the model are *certain* to be activated in the real project;
4. It is not always possible to discern *a priori* if there is a local mechanism; the solver has to examine the structure node by node in order to spot possible mechanisms.

It is thus extremely important that the solver refuse to solve a structure where the possibility of rigid motions exists, even if these are not directly activated by the forces placed on the model.

Once the mechanism has been identified, we have to hope that it is possible to see clearly what rigid motion the structure might perform (fig. 7-9). This result can be obtained numerically by means of a technique called *singular value decomposition* (SVD). Because this involves a significant calculation cost, not many solvers offer this feature. Generally, when in the course of the solution the solver encounters an abnormal situation (such as when a term of direct stiffness becomes null, or when a negative number appears under the radical sign), the solver stops, and the mechanism has to be looked for starting with the scant information available: the cryptic message that often appears is "*check dof 7856*" (or "*3478"* or any other number).

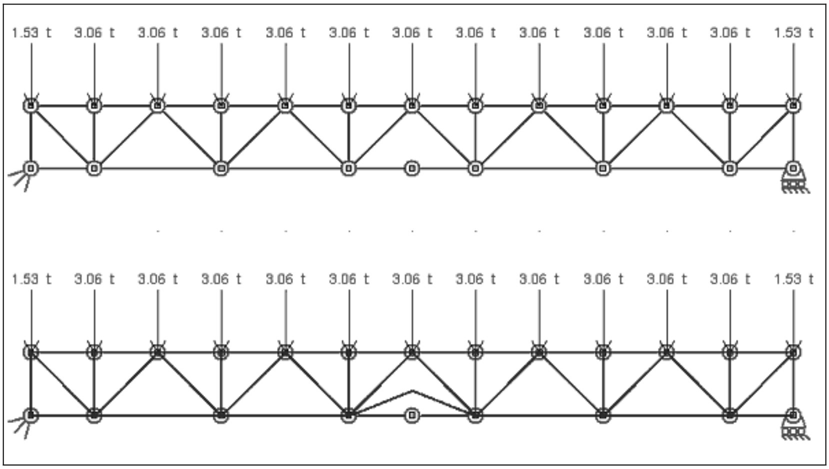
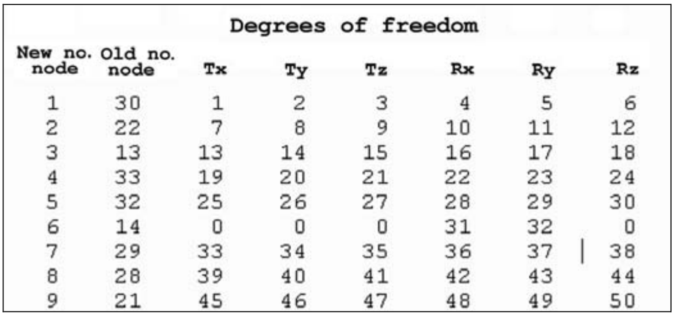


Figure 7-9: Local lability and how it is evidenced (from the author's C.E.S.CO software program).

In this case it is necessary to consult the table that associates the various degrees of freedom to the nodal displacements in order to understand exactly which degree of freedom (dof) is being referred to (fig. 7.10). In our example we might find that "dof 7856" corresponds to "translation Y of node 1489", or that "dof 3478" corresponds to "rotation Z of node 786". This information is not complete, but in most cases it is enough to understand the nature of the mechanism and see to its elimination.



Degrees of freedom

New no. node	Old no. node	Tx	Ty	Tz	Rx	Ry	Rz
1	30	1	2	3	4	5	6
2	22	7	8	9	10	11	12
3	13	13	14	15	16	17	18
4	33	19	20	21	22	23	24
5	32	25	26	27	28	29	30
6	14	0	0	0	31	32	0
7	29	33	34	35	36	37	38
8	28	39	40	41	42	43	44
9	21	45	46	47	48	49	50

Figure 7-10: Sample table of correspondences between degrees of freedom and nodes with old and new node numbers (Sargon solver, CLEVER, from the ASCII output).

A concise way of providing information about a mechanism without solving an entire SVD problem is that of creating a fictitious displacement field that gives a unitary value (1) to the degree of freedom involved in the arrest of the solving process, and a null value (0) to all the others. While in some fortunate cases, this immediately produces an image of the desired rigid deformed configuration (fig.

7-9), more frequently it produces an unrealistic deformed configuration which, however, at least puts us on the road to understanding the problem. For example, in the cases shown in fig. 7-8, the fictitious deformed configuration might be that shown in fig. 7-11.

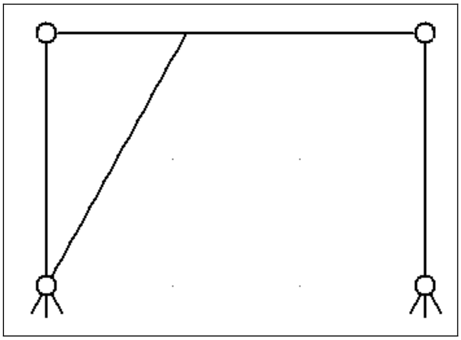


Figure 7-11: Fictitious deformed configuration aimed at finding static indeterminacy.

7.2.5.2 QUASI-MECHANISMS: LOOKING FOR AREAS AT RISK

Even when we are dealing with systems that are strictly speaking non-singular, and thus which can be inverted and calculated, it is still possible to find structures with insufficient stiffness to resist all possible motions. In this case, the structure's "weak spot" can be identified through modal analysis. Actually, to the large period vibration modes, which are the first to be extracted, correspond displacements that activate the structure's particularly flexible zones (those with a lower stiffness value "k"; recall the formula that gives the pulsation ω of the simple oscillator: $\omega = \sqrt{\frac{k}{m}}$). Modal analysis can thus be used as a "structural weakness detector".

Consider, for example, the multi-storey building shown in fig. 7-12. The first vibration mode is the one that deforms the typical floor by means of an oscillation on the weak axis of the longest beams. Actually, the structure doesn't have any floor bracing, and thus is particularly weak in regard to all those displacements that involve a variation in the shape of the floor itself. The first vibration mode (0.23 sec, the structure in the example carries only its own weight) is followed by various other modes that all have more or less the same period (0.23 sec), and all of which show bending on the weak side of the beams with the greatest length. Note that the only elements that are distorted are essentially the beams on the long side, deflected along their axis of least inertia. The rods that act as bracing neither elongate nor shorten, since the node moves perpendicular to the floor that they identify.

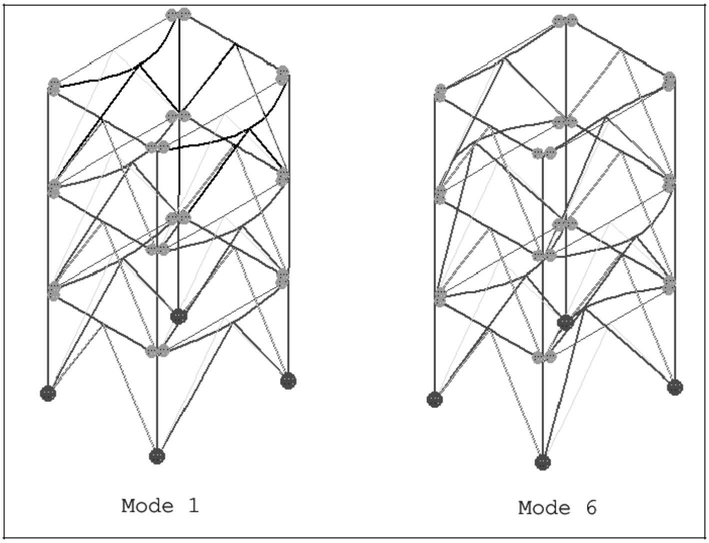
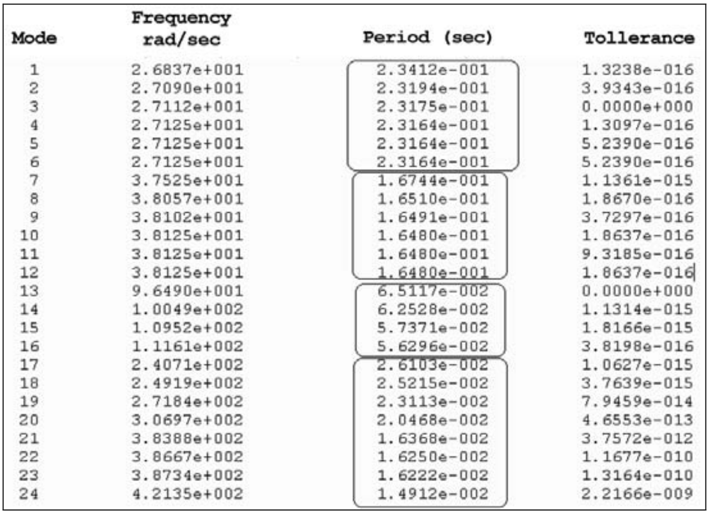


Figure 7-12: Multi-storey building, modes 1 and 6.



Mode	Frequency rad/sec	Period (sec)	Tollerance
1	2.6837e+001	2.3412e-001	1.3238e-016
2	2.7090e+001	2.3194e-001	3.9343e-016
3	2.7112e+001	2.3175e-001	0.0000e+000
4	2.7125e+001	2.3164e-001	1.3097e-016
5	2.7125e+001	2.3164e-001	5.2390e-016
6	2.7125e+001	2.3164e-001	5.2390e-016
7	3.7525e+001	1.6744e-001	1.1361e-015
8	3.8057e+001	1.6510e-001	1.8670e-016
9	3.8102e+001	1.6491e-001	3.7297e-016
10	3.8125e+001	1.6480e-001	1.8637e-016
11	3.8125e+001	1.6480e-001	9.3185e-016
12	3.8125e+001	1.6480e-001	1.8637e-016
13	9.6490e+001	6.5117e-002	0.0000e+000
14	1.0049e+002	6.2528e-002	1.1314e-015
15	1.0952e+002	5.7371e-002	1.8166e-015
16	1.1161e+002	5.6296e-002	3.8198e-016
17	2.4071e+002	2.6103e-002	1.0627e-015
18	2.4919e+002	2.5215e-002	3.7639e-015
19	2.7184e+002	2.3113e-002	7.9459e-014
20	3.0697e+002	2.0468e-002	4.6553e-013
21	3.8388e+002	1.6368e-002	3.7572e-012
22	3.8667e+002	1.6250e-002	1.1677e-010
23	3.8734e+002	1.6222e-002	1.3164e-010
24	4.2135e+002	1.4912e-002	2.2166e-009

Figure 7-13: Multi-storey building, period table, modes 1-24.

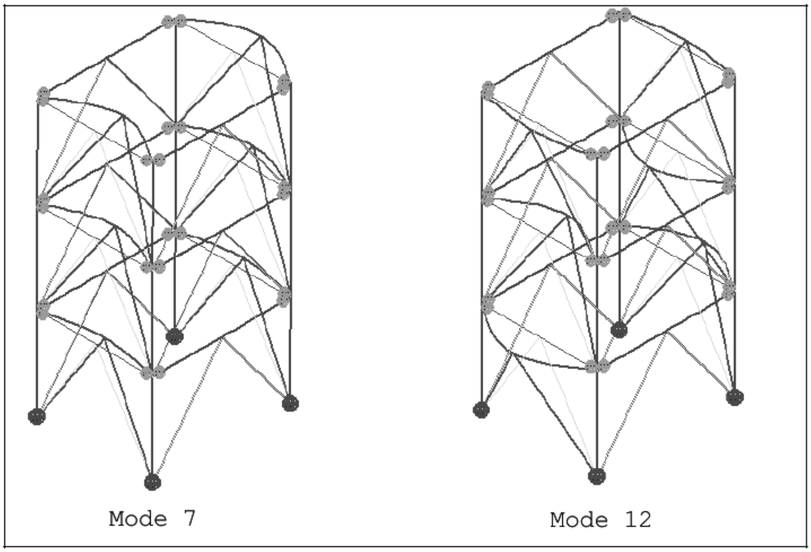


Figure 7-14: Multi-storey building, modes 7 and 12.

From mode 7 to mode 12 we find a second group of modes, all of which show the deflection of the shortest beam on the side of least stiffness. In this case as well the only deformed elements are the beams, but this time those of the short side (and thus stiffer than the previous ones).

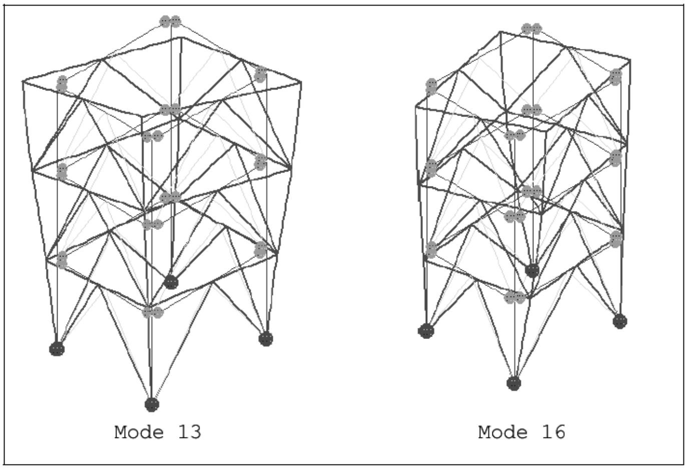


Figure 7-15: Multi-storey building, modes 13 and 16.

From mode 13 to mode 16 we have modal deformed configurations that are distorsional or torsional, with a period that is clearly smaller on average, since, resulting in the bending of the verticals, they solicit a stiffness that is much greater than the previous ones. Finally, from mode 17 on, we find modal deformed shapes that are closer to those that we can expect to find in a structure of this kind. The examination of the values that fall on the diagonal of the stiffness matrix allows us to see that for the minimum and maximum diagonal term we have, respectively,

Max Kii = 1.3348e+010

Min Kii = 9.3695e+001.

The difference between these two is quite significant, about 8 orders of magnitude.

The examination of this type of modal shapes naturally leads the designer to stiffen the floor by adding suitable bracing, in the form of either diagonals or as continuous plane structures. For purposes of comparison, we have added membrane elements to the previous model, leaving the overall mass unaltered. The membrane elements thus have an effect similar to those of MPC-type constraints, even if they actually simulate more closely what effectively takes place in the structure, which is not undeformable. Incidentally, with regard to the size of the mesh adopted for the membrane, it should be noted here that the determination of the stress within those elements does not enter into this problem: otherwise the dimensions of the elements would have to have been completely different.

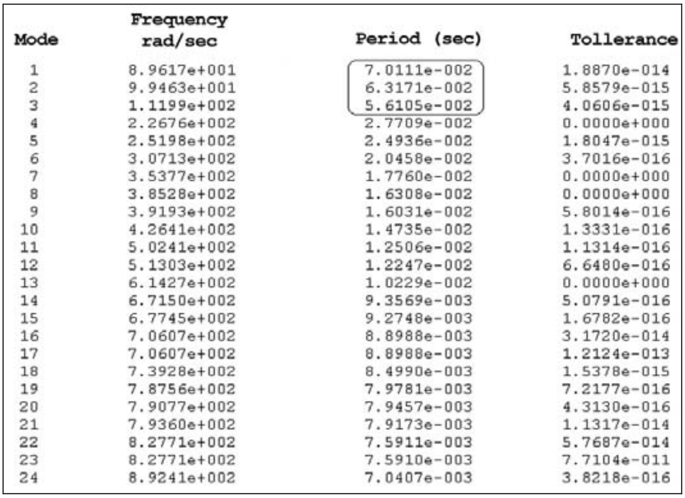
The maximum and minimum values on the diagonal of the stiffness matrix have become:

Max Kii = 1.3348e+010

Min Kii = 6.3474e+004,

with a difference of 5.5 orders of magnitude (note that the term of maximum stiffness has remained unchanged). An examination of the mode table shows period values that are clearly smaller (the first is 7.e-2 sec), and much more uniformly distributed, except for the first three modes, which refer, as expected, to the translation in the X and Y directions and to the torsion.

Thus we see that there can effectively be an insufficient or inappropriate internal or external constraint even without it having to generate an actual mechanism. Modal analysis is therefore also extremely useful searching out areas that are insufficiently constrained within the structure.



Mode	Frequency rad/sec	Period (sec)	Tollerance
1	8.9617e+001	7.0111e-002	1.8870e-014
2	9.9463e+001	6.3171e-002	5.8579e-015
3	1.1199e+002	5.6105e-002	4.0606e-015
4	2.2676e+002	2.7709e-002	0.0000e+000
5	2.5198e+002	2.4936e-002	1.8047e-015
6	3.0713e+002	2.0458e-002	3.7016e-016
7	3.5377e+002	1.7760e-002	0.0000e+000
8	3.8528e+002	1.6308e-002	0.0000e+000
9	3.9193e+002	1.6031e-002	5.8014e-016
10	4.2641e+002	1.4735e-002	1.3331e-016
11	5.0241e+002	1.2506e-002	1.1314e-016
12	5.1303e+002	1.2247e-002	6.6480e-016
13	6.1427e+002	1.0229e-002	0.0000e+000
14	6.7150e+002	9.3569e-003	5.0791e-016
15	6.7745e+002	9.2748e-003	1.6782e-016
16	7.0607e+002	8.8988e-003	3.1720e-014
17	7.0607e+002	8.8988e-003	1.2124e-013
18	7.3928e+002	8.4990e-003	1.5378e-015
19	7.8756e+002	7.9781e-003	7.2177e-016
20	7.9077e+002	7.9457e-003	4.3130e-016
21	7.9360e+002	7.9173e-003	1.1317e-014
22	8.2771e+002	7.5911e-003	5.7687e-014
23	8.2771e+002	7.5910e-003	7.7104e-011
24	8.9241e+002	7.0407e-003	3.8218e-016

Figure 7-16: Table of vibration modes of a structure stiffened with floor bracing.

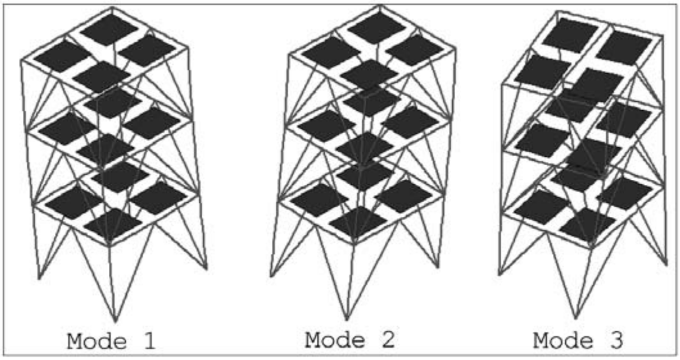


Figure 7-17: Stiffened structure, modes 1, 2 and 3.

7.2.6 SYMMETRICAL CONSTRAINTS

There are cases in which the problem to be studied is symmetrical, not only in its geometry but also in the constraints and loads. In such cases, when there is a pressing need to save time and degrees of freedom, it is possible to study only a part of the model by setting suitable conditions of constraint.

The need to avail ourselves of this possibility occurs much less frequently today than in the past, because of the significant increase in the performance of computers. Not so very many years ago, available RAM amounted to only 640 kilobytes, while today we easily find computers with several gigabytes of RAM, an increase by a factor of more than 10,000. The requirements for mesh density have not increased at the same rate, and so it is often more convenient and simpler to generate the entire structure rather than only a part of it.

However, when we are grappling with a very large problem (say, from 20,000 degrees of freedom up), it still makes sense – where possible – to take advantage of the symmetries that exist in the structure.

In order for a structure to be considered "symmetrical" there has to be symmetry in the geometry, in the material properties, in the constraints and end releases present, and in the applied loads. When all of these requirements are met, then "the problem is symmetrical" and it can be simplified by modelling only half of the structure (the half either to the left or the right of the plane of symmetry) with half of the applied loads and suitable "symmetrical" constraint conditions on the particular plane of symmetry. Table 7-2 summarises the symmetrical constraints to be applied at nodes lying on the plane of symmetry for the various possible symmetry planes. It should be noted that the forces applied exactly on the plane of symmetry have to be halved (fig. 7-18); if not, the problem is not correctly modelled.

Plane of symmetry	**Null degrees of freedom on the plane of symmetry (symmetrical loads)**
XY	Tz, Rx, Ry
YZ	Tx, Ry, Rz
XZ	Ty, Rx, Rz

Table 7-2: Constraints on various possible planes of symmetry.

Note that when a symmetrical structure – or one believed to be symmetrical – is modelled without taking advantage of the symmetry conditions, the calculated displacement field *must* satisfy the structure's expected symmetry conditions. If this is not the case then we can expect to see errors in modelling (i.e., the model is not symmetrical as believed). This circumstance is quite useful for checking complex models. For example, fig. 7-19 shows a model that is apparently symmetrical, but whose deformed configuration is obviously not symmetrical. In cases such as these there is always an unwanted asymmetry in the modelling, often difficult to track down.

In the model shown in fig. 7-19, two finite elements (shown in fig. 7-20) were were lacking. This resulted in a deformed configuration that was decidedly asymmetrical.

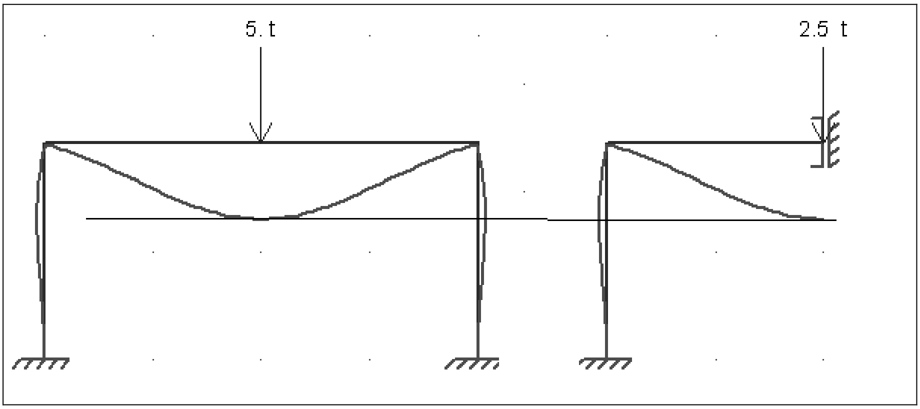


Figure 7-18: Halving of the forces (necessarily parallel to the plane of symmetry) acting on the plane of symmetry (from the author's C.E.S.CO software program).

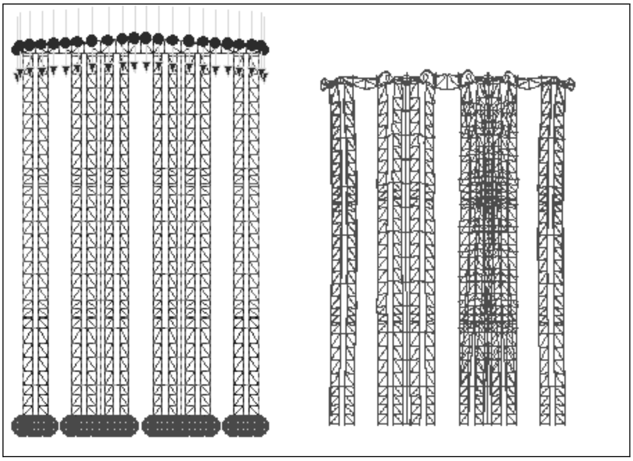


Figure 7-19: Asymmetry in the deformed configuration of a structure that is apparently symmetrical (SARGON model, reproduced courtesy of Gruppo Marcegaglia – Spa Divisione Ponteggi Dalmine, which provided the model for the example).

In addition to the case of a symmetrical structure, the case of a *symmetrical structure with antisymmetrical loads* is interesting. We call a load situation "antisymmetrical" when a single reflection of the structure with the loads followed by an inversion of sign of all the loads leads back to the initial structure [cf. Cook, Malkus, Plesha 1989].

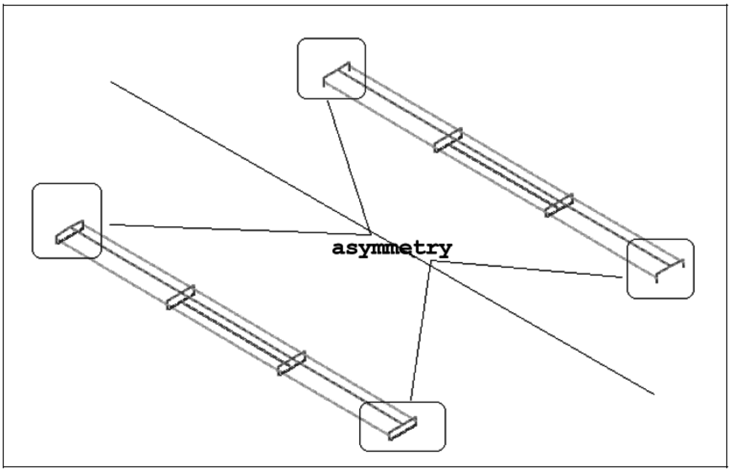


Figure 7-20: From among the circa 10,000 elements of the model shown in fig. 7-19, the two that are highlighted were lacking. As a result the structure was effectively asymmetrical. Note that in the orthogonal view it was impossible to see that the two end elements were missing because the intermediate elements were present.

In essence, the loads are antisymmetrical if the "symmetrical" forces acting on the structure are perpendicular to the plane of symmetry and all with the same sign, or if the forces acting parallel to the plane of symmetry have opposite directions left and right to the plane of antisymmetry. In this case the answer in terms of displacement is antisymmetrical and the constraints to be placed on the plane of symmetry are antisymmetrical constraints. Table 7-3 shows the antisymmetrical constraints to be placed in these cases, for the structure's various planes of antisymmetry.

Plane of symmetry	Null degrees of freedom on the plane of symmetry (antisymmetrical loads)
XY	Tx, Ty, Rz
YZ	Ty, Tz, Rx
XZ	Tx, Tz, Ry

Table 7-3: Constraints for a symmetrical structure with antisymmetrical loads.

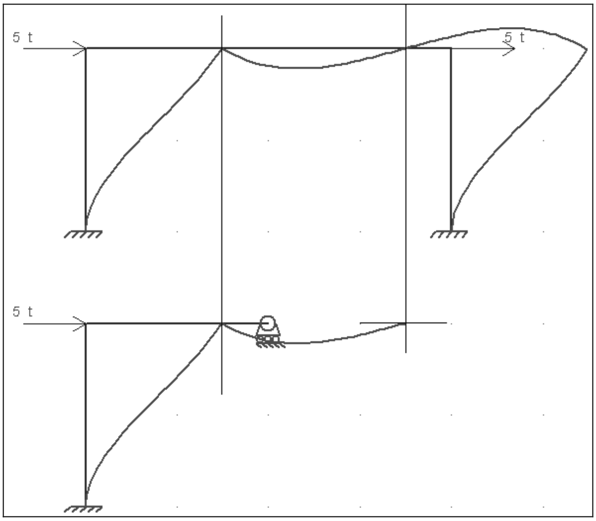


Figure 7-21: Symmetrical structure with antisymmetrical loads. Note the null vertical displacement of the point that was originally on the plane of symmetry, and the identity of the displacements found (from the author's C.E.S.CO software program).

7.3 INACTIVE DEGREES OF FREEDOM: *AD HOC* FORMULATIONS

Now that we have discussed real constraints, that is, those aimed at eliminating actual occurrences of rigid motion within the structure, let's look at the techniques that can be used to exclude *a priori*, if not all, then at least many of the inactive degrees of freedom.

When the structure in question satisfies the given requirements of homogeneity, we can expect that all of the structure's nodes possess a certain identical set of degrees of freedom which are effectively inactive.

Computer applications address this situation in one of three ways:

1. Some software programs require nothing more than explaining to the program what particular kind of problem is concerned (plane stress, plane grid structure, etc.);
2. Some software programs require choosing a certain "constraint mask" for the problem;
3. Some programs don't use constraint mask and ask the user instead to set appropriate fictitious constraints.

The first approach requires the user to choose from a menu of possible typical problems, and the program generally requires that the structure be defined through the use of a very precise coordinate system. The advantages are that the user gets to solve a problem that is much simpler, and that from the outset the algebraic problem is much less complex because, since none of the inactive degrees of freedom are even taken into consideration, when stiffness matrix **K** is created, it is already free of irrelevant degrees of freedom. The disadvantage is a certain unwieldiness that is tied to the fact that the structure has to be defined in the reference system contrived by the software developer, rather than the one that is most convenient for the user (for example, in plane *xy* instead of *xz*, or *xz* instead of *x*y). Further, the adoption of a correct formulation *doesn't mean that there are no longer any inactive degrees of freedom in any of the nodes of the structure*, and these still have to be eliminated explicitly using the techniques described in the section that follows. For example, given a plane frame, if *n* beams, all hinged, converge in a node, then the rotation of that node is inactive (we will see why this is so in more detail in chapter 12) but this could not be established *a priori for all nodes*.

The second approach, which can be considered an intermediate one between the first and the third, requires that the user tell the software which degrees of freedom are inactive in the problem. Defining these inactive degrees of freedom leads to the creation of a constraint mask, and thus none of the nodes in the structure will have these degrees of freedom. The structure is effectively simpler, since these inactive degrees of freedom have been eliminated from the beginning. A constraint mask is a detailed specification of a given set of global degrees of freedom that are to be null *for all nodes of the structure* without exception. For example, a constraint mask might be

$$(R_x, R_y, R_z),$$

meaning that in all nodes of the structure the rotations about the global axes (*x*, *y*, *z*) must be null. While the advantage of this approach is flexibility, the disadvantage is that in order to define the constraint mask one has to know exactly what one is doing and why, and unfortunately this cannot always be taken for granted. Further, as in the first approach, the constraint mask does not necessarily eliminate all inactive degrees of freedom. To help make this clear, table 7-4 gives a list of problems and their corresponding constraint masks; the reasons behind what is shown here will be treated in detail when the finite elements proper to each formulation are dealt with in chapter 11.

In the third approach, the unnecessary degrees of freedom are explicitly eliminated through their association with a constraint that causes them to become null. In this way the "constraint masks" are made to be equivalent to actual (fictitious) constraints assigned to the nodes of the structure (of course, many constraints can be applied at the same time). The disadvantage is that in principle every single node has to be treated (though this is not true in practice: it is only necessary to select all the nodes that have the same constraints and apply them). The advantage is that this approach does away with unwieldy options and choices that don't make the problem easier anyway.

Problem	**Constraint template**
Truss structure plane xy	Tz, Rx, Ry, Rz
Truss structure plane yz	Tx, Rx, Ry, Rz
Truss structure plane xz	Ty, Rx, Ry, Rz
Space grid	Rx, Ry, Rz
Frame plane xy	Tz, Rx, Ry
Frame plane yz	Tx, Ry, Rz
Frame plane xz	Ty, Rx, Rz
Space frame	-----
Plane stress, plane strain, axial symmetry xy	Tz, Rx, Ry, **Rz**
Plane stress, plane strain, axial symmetry yz	Tx, **Rx**, Ry, Rz
Plane stress, plane strain, axial symmetry xz	Ty, Rx, **Ry**, Rz
Solid problems (solid elements)	Rx, Ry, Rz

Table 7-4: Constraint masks for various possible structural problems. It is presumed that the loads also lie in the plane of the structure (otherwise there might be no possible solution). It is presumed that the formulation of the elements does not use drilling DOF (see chapter 11, § 11.5.3). If this is not the case, then the degrees of freedom shown in bold should not be constrained.

7.4 INACTIVE DEGREES OF FREEDOM: PSEUDO-MECHANISM

7.4.1 ACTIVATED AND INACTIVATED PSEUDO-MECHANISM

If the inactive degrees of freedom are not all eliminated by an appropriate formulation of the problem, then it is possible that during assembly some of the rows (and columns) of the stiffness matrix will remain null. The degrees of freedom that correspond to null rows are called inactive degrees of freedom; it is to these that fictitious mechanisms, called pseudo-mechanisms, are associated. If we examine more closely the corresponding equation of the solving system, we will find that the general form

$$\sum K_{ij}u_j - f_i = 0$$

can degenerate into two possible limit situations when there are inactive degrees of freedom:

$$0 = 0$$

$$-f_i = 0.$$

The first degenerate form is an identity, and it is found when there are no equivalent forces acting on the inactive degree of freedom (*unactivated pseudo-mechanism*). Although this poses a difficulty in the solving process by introducing a pseudo-mechanism, in principle there is no difficulty in solving the problem: we need only discard this identity as irrevelant, or modify the system in some way – we will see how later – so that we can calculate some fictitious value for the inactive displacement u_i.

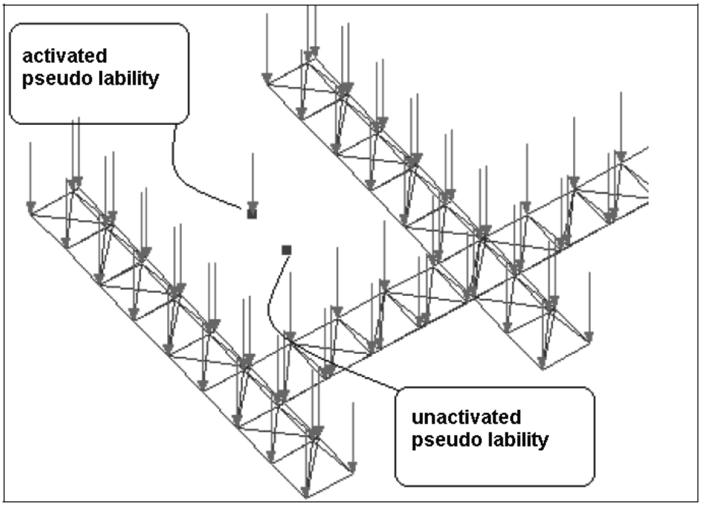


Figure 7-22: Examples of activated and unactivated pseudo-labilities.

The second degenerate form is not an identity, nor even an equation. It is a false statement, and it occurs when equivalent forces are applied to the inactive degree of freedom (activated pseudo-mechanism). Since no element is connected to the inactive degree of freedom, these equivalent forces must be either the result of numerical errors in calculation, or the result of an error on the part of the engineer, *who applied a force to a degree of freedom that did not have any stiffness*. It is important to note that in the case of an error by the engineer it is impossible to solve the problem, because it is imprudent to discard *a priori* the inequality in question. Actually, the presence of a force might not be an error of the engineer, but rather the absence of a stiffness that is wrong (due, for example, to the the use of inappropriate finite elements).

7.4.2 ELIMINATING PSEUDO-MECHANISMS

Software programs usually have at least two possible approachees to pseudo-mechanisms.

The first approach is to discard them, both when they are activated and when they are inactivated. This approach has the advantage of not requiring any action aimed at identifying and removing the pseudo-mechanisms on the part of the engineer (and it is thus quite popular among non-experts). However, it has the disadvantage of making it impossible to trace back to unintentional inconsistencies and problems in the mesh. The presence of tens or hundreds of inactive degrees of freedom is not a good indicator of the engineer's skill, and it is also possible that the automatic removal of all of these pseudo-mechanisms – especially the activated ones, which almost certainly indicate that there is a problem – can compromise the reliability of the analysis. The software program will come up with a solution, but there is no way of knowing if that solution is dependable or not.

The second approach is that of always interrupting the execution and refusing to go forward until the pseudo-mechanism has been explicitly removed by means of the attribution of as many added constraints as there are pseudo-labilities in the model. This approach has the disadvantage of requiring the analyst to make a careful investigation of the causes of the pseudo-lability, and see to their removal using the instructions explicitly provided by the software. However, it has the advantage of guaranteeing that the model is a properly constructed one, consistent with itself and with the rules of the FEM. This does not yet prove that the model is reliable, but it is certainly an important step in this direction. Once the presence of a certain number of pseudo-mechanisms has been identified in the model, these can be eliminated by means of the addition of a certain number of fictitious constraints on the inactive degrees of freedom. Actually, if a given degree of freedom is constrained, its equilibrium equation is essentially discarded by the solving procedure and at the end of solving the value of that degree of freedom is set equal to 0. The addition of these constraints doesn't have anything to do with the actual constraining of the structure: in the most widely-used software programs each node is "born" with six independent degrees of freedom (unless a particular kind of formulation is specified), but it is possible that only a part of these degrees of freedom are effectively necessary for solving the problem. In this case it is up to the analyst to remove all of the inactive degrees of freedom from the structure through the addition of fictitious constraints.

To these two approaches we can add a third one. It is possible to establish that the standard functioning of a software program is the one typical of the second approach, that is, it stops the procedure when there is a pseudo-lability, which is signalled in an output file. However, this behaviour can be accompanied by an option that makes it possible to eliminate the pseudo-labilities automatically, *as long as these are not activated.* This route allows the user to calibrate the degree of protection of the software as a function of his or her own individual needs, and in any case provides protection against activated pseudo-mechanisms, which are the most dangerous ones.

Pseudo-mechanisms are one of the most frequent causes for the interruption of the procedure of solving, and one of the typical reasons why the beginner finds it impossible to carry out his work. It is thus of the utmost importance to understand thoroughly the reasons for pseudo-mechanisms and to learn to identify them *before* the solver locks up.

An in-depth discussion of the techniques for recognising pseudo-mechanisms requires that we first explain the various finite elements that are usually used, and thus we refer to the examples in chapter 12.

7.4.3 IDENTIFYING PSEUDO-MECHANISMS NUMERICALLY

From a numeric point of view, the identification of pseudo-labilities can pose some problems for the solver, because due to rounding errors, or because of the inevitable small inaccuracies related to the position of the local tern of the elements, it is also possible that the terms of the stiffness matrix or the vector of known terms are not rigorously null, but instead have some very small value.

The condition

$$K_{ii} = 0$$

is no longer sufficient to identify a pseudo-lability, because it might be that

$$K_{ii} \approx 0$$

that is, that the diagonal term is not rigorously null, but that it is very close to zero.

In this case the task of the solver is to identify the pseudo-lability by means of a comparison between the order of magnitude of the term in question of the diagonal of the stiffness matrix K_{ii} and the maximum diagonal term of the stiffness matrix itself, M (see also chapter 9). The quantity

$$K_{ii} / M$$

is compared to an appropriate value of tollerance T, which depends on the precision with which the floating point operations have been performed. The condition

$$(K_{ii} / M) < T$$

is then used to distinguish the case of a very small stiffness (>T) from that of a "null" stiffness (<T). Naturally, just as it is possible that a pseudo-mechanism can be hidden by rounding errors, it is also possible that a structural quasi-mechanism, related for example to the way in which the structure has been built, can be numerically confused with a pseudo-mechanism. The boundary can be defined in a purely numerical way, and it is up to the skilful analyst to understand whether he is dealing with the first or the second case.

7.5 SELF-EQUILIBRATING STRUCTURES

It sometimes happens that we have to study structures which actually don't have any constraints at all, but which are subject to a set of self-equilibrating actions that bring about a certain regime of stresses and strains. Not being constrained to anything that is fixed, the structure is free to move rigidly without any work being performed, and in this sense it can be considered hypostatic. At the same time, the fact that the actions are globally in equilibrium means that no constraint is actually needed for the study of the regime of stresses and strains.

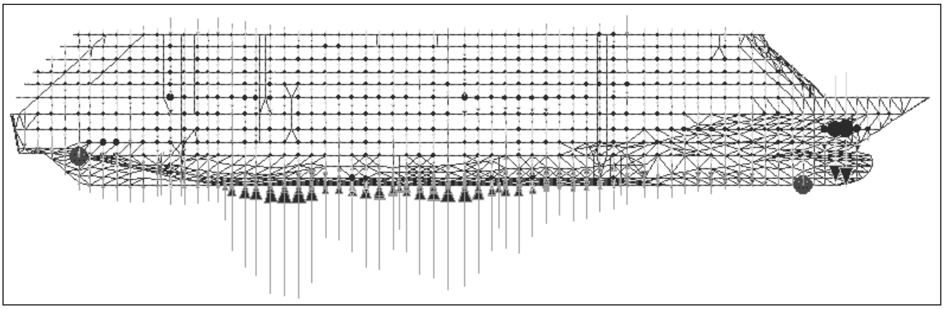


Figure 7-23: A cruise ship subject to a design wave (SARGON model). The dots represent fictitious constraints.

The situation is such that, in the absence of any constraint, the structure cannot be solved, because the stiffness matrix of the system of solving equations is singular (as many times singular as there are independent rigid motions allowed to the structure). The problem is solved by adding fictitious "ground" constraints, in just sufficient number to prevent rigid motions, and then checking, once solving is finished, that the reactions exerted by the constraints are effectively close to zero. If the structure is effectively self-balanced, the constraint reactions of the isostatic system created have to be null, and thus do not essentially influence the analysis.

For example, fig. 7-23 shows the study of an ocean liner subject to a design wave. In this case the weight of the ship (variously distributed along its entire surface both in length and in height) is exactly compensated for by a complex distribution of pressures on the ship's shell, which in their turn depend on the form of the wave over which the ship is passing. Constraints are of no use, but in order to solve the structure numerically it is necessary to take three nodes, two at the stern and one at the stem. The node at the stem is constrained in this way

$$Tx=Ty=Tz=0;$$

the first node of the stern in this way

$$Ty=Tz=0;$$

and the second node of the stern in this way

$$Tz=0.$$

As you can see, the total number of constraints is equal to 6, that is, exactly equal to the overall number of global degrees of freedom of the ship. The nodes were placed quite far apart from each other, so that the constraint reactions capable of perturbing the stress locally were reduced to a minimum. Actually, while it is true that in principle the model could be considered self-balancing, in practice this objective could only be reached approximately, given the complexity of the form of the wave and the dimensions of the plate elements simulating the shell. The calculated constraint reactions thus had to be "null" in a numerical sense, that is to say, negligible with respect to the overall resultant. The ship shown in fig. 7-23 weighs more that 22,000 tons, and thus constraint reactions on the order of 3t can be considered negligible (3/22000=0.0136%).

8 SOLVING AND STRESS RECOVERY

8.1 SOLVING

The main problems related to the actual solution of the system of solving equations are connected to the precision and the efficiency of the solution. We will deal with how to determine the precision of the solution in the next chapter. This present chapter will be dedicated to the efficiency of the solving procedure, an efficiency that can only be attained by making the best possible use of the computer resources available according to the requirements of each individual problem.

When the FEM was just coming into use, even ordinary problems were significantly difficult in terms of computation, and this immediately gave rise to a series of studies aimed at achieving the objective of performing efficient numerical analyses even though the available quantity of memory was quite modest.

The discerning user should be familiar with the basic problems related to the solution of very large systems of linear equations, so that he or she knows what to do should situations arise where it is necessary to use particular methods in order to manoeuvre around obstacles to computation. Further, knowing at least summarily the problems that the solver must be able to address, makes the user better able to understand why possible interruptions might occur during this phase of the job.

8.1.1 BRIEF REVIEW OF METHODS OF SOLVING LINEAR SYSTEMS

The solution of large systems of linear equations is one of the most important topics in numerical calculation, and certainly one with a seemingly endless bibliography. It is beyond the scope of this book to provide a detailed discussion of the most commonly used techniques, and thus we will limit ourselves to providing some fundamentals relative to the three most widespread solutions as far as the FEM is concerned, namely, the method of Gaussian reduction, the Cholesky method, and the Gauss-Seidel method.

As we have seen, the problem that must be solved takes the form

$$\mathbf{Ku}=\mathbf{p},$$

and stiffness matrix **K** has the important characteristics of being symmetrical and positive definite. Contrary to what might be believed, the problem is not normally solved by looking for the inverse matrix $\mathbf{K}^{-1}$ of the stiffness matrix **K**, and obtaining the solution by means of the product

$$\mathbf{u}=\mathbf{K}^{-1}\mathbf{p},$$

Structural analysis with finite elements
978-0-7277-4093-9

because this tactic would make the computational task decidedly more unwieldy than it needs to be (remember that we are talking about systems of thousands, tens of thousands, or even hundreds of thousands of equations). It should also be borne in mind that more than solving *a* system of equations, in reality it is necessary to solve a certain *number* of systems, that is, a number equal to the number of load cases that are expected to occur in the structure. Actually we haven't made just a single vector **p** of known terms, but many vectors $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_3$… of known terms, and therefore there are as many problems to solve as there are load cases.

It is thus of the utmost importance that the algorithms are such that they do not require repeating operations that only involve the stiffness matrix **K**, which should be performed only a single time.

8.1.1.1 DIRECT METHODS

Direct methods are those that make it possible to solve a system of linear equations with a finite, predetermined number of steps. All of these methods include a first phase called the "reduction" of the stiffness matrix **K** of the system. During this phase, matrix **K** is decomposed into the product of two triangular matrices[1], a lower triangle **L** and an upper triangle **U**, such that

$$\mathbf{K}=\mathbf{LU}.$$

This reduction is calculated only once, and it is the most onerous phase in terms of computational effort.

Once decomposed, the matrix of the original problem,

$$\mathbf{Ku}=\mathbf{p},$$

becomes

$$(\mathbf{LU})\mathbf{u}=\mathbf{p},$$

which is equivalent to solving the following two systems:

$$\mathbf{Lx}=\mathbf{p} \qquad [8\text{-}1]$$

$$\mathbf{Uu}=\mathbf{x}. \qquad [8\text{-}2]$$

This procedure might not appear advantageous, but in reality, since the two systems of linear equations 8-1 and 8-2 are triangular (given that their coefficient matrices are, as we said, lower triangular matrix **L** and upper triangular matrix **U**), they can be solved quite rapidly, because each equation directly provides the final value of an unknown, which can then be used as a known term in the equations that follow. In the first system, 8-1, we start with a first equation containing a single unknown, and go forward (*forward solving*). In the second

[1] Recall that a lower triangular matrix is one in which all the terms lying above the main diagonal are null, while an upper triangular matrix is the opposite, that is, all the terms lying below the main diagonal are null.

system, 8-2, we start with the last equation, which also contains a single unknown, and work backwards to the next to last, and so forth (*backward solving*). The first system determines x (which is an intermediate unknown), the second system determines the actual unknown **u**.

What is remarkable is that the phase of reduction is performed only a single time, and that the solution of n load cases requires the solution of n systems like those of 8-1 and n like those of 8-2.

REDUCTION

FORWARD SOLVING CASE 1

FORWARD SOLVING CASE 2

●●●●●●●●●●●●●●●●

FORWARD SOLVING CASE *n*

BACKWARD SOLVING CASE 1

BACKWARD SOLVING CASE 2

●●●●●●●●●●●●●●●●

BACKWARD SOLVING CASE *n*

Figure 8-1: Diagram of the typical steps in a generic solving with the direct method

The solving phase can be represented in diagram form, as shown in figure 8-1. Generally speaking, the majority of errors and problems take place during the reduction phase, which is that which typically brings to light problems of mechanisms or pseudo-mechanisms. The reduction phase is the slowest, while the phases of forward and backward solving are usually quite rapid.

8.1.1.1.1 GAUSSIAN REDUCTION

The technique of Gaussian reduction is the most widely used. The idea is that of transforming the system into triangular form by means of a series of steps aimed at eliminating the various unknowns one at a time, starting with the first one, from successive equations.

In order to achieve this end, each equation is multiplied by an appropriate term and then summed to the equation relative to the unknown to be eliminated. By repeating this procedure for all of the unknowns in the problem, we finally arrive at a system in triangular form that can be easily solved.

Let's consider the following system:

$$3x+y+2z=11$$

$$x+2y+6z=23$$
$$2x+6y+5z=29.$$

We multiply the second equation by 3, and subtract it from the first equation; we then multiply the third equation by 3/2 and subtract it from the first. Thus we obtain the equivalent system:

$$3x+y+2z=11$$
$$-5y-16z=-58$$
$$-8y-5.5z=-32.5.$$

As you can see, the *x* has disappeared from the second and third equations. Now we multiply the third equation by –5/8 and subtract it from the second. We thus obtain the following equivalent system:

$$3x+y+2z=11$$
$$-5y-16z=-58$$
$$-12.5625z=-37.6875,$$

which has the desired triangular form. Now, from the last equation – with a single unknown – we find that

$$z=3.$$

From the second,

$$-5y-16(3)=-58,$$

we find that

$$y=2.$$

Finally, from the first equation

$$3x+2+2(3)=11$$

we find that

$$x=1.$$

This simple example allows us to understand a series of things.

1. When multiplying the equations by the "convenient" factor aimed at eliminating the variable that has to disappear from an equation, we might have to generate "strange" numbers that give rise to rounding errors (see, for example, the last equation found in *z*: in this particular case there aren't any errors, but there could have been had the numbers been different).
2. The factors of multiplication depend on how the equations are ordered, thus they can be reordered so as to obtain situations that are more favourable in numeric terms;
3. The procedure of Gaussian reduction incorporates both the reduction phase as well as that of forward solving, since the final

result is a system with an upper triangular matrix (**U**), which must be followed by a backward solving phase.

An efficient implementation of the method of Gaussian elimination calls for various techniques that are intended to improve the numerical quality of the solution, which might otherwise turn out to be sufficiently impoverished because of rounding errors. Such techniques call for a pre-multiplication of the equations so that there are not too many dissimilar terms such as order of magnitude (scaling), as well as a reordering of the equations in order to be able to use the most favourable terms of the matrix for performing the necessary operations (pivoting).

8.1.1.1.2 THE CHOLESKY METHOD

The Cholesky method is particularly useful for solving structural problems, because it requires that the coefficient matrix not only be symmetrical but rigorously positive definite as well. Thus if the matrix contains any mechanism at all, the system cannot be solved and the algorithm locks up.

The Cholesky method decomposes the stiffness matrix **K** in the following manner

$$\mathbf{K}=\mathbf{L}\mathbf{L}^{\mathbf{T}},$$

that is, the upper triangular matrix **U** of the decomposition **K=LU** is simply the lower triangular matrix **L** transposed. The formulas that express the terms L_{ij} of the matrix can be written in the following way:

$$L_{ij} = \frac{K_{ij} - \sum_{l=1}^{j-1} L_{il} L_{jl}}{L_{jj}} \qquad j=1, 2, \ldots i\text{-}1.$$

$$L_{ii} = \sqrt{K_{ii} - \sum_{l=1}^{i-1} L_{il}{}^{2}}$$

These formulas make it possible to calculate the terms of **L** immediately starting from those of **K**. As you can see, the formulas contain a square root that can be calculated in the context of real numbers only if what is under the radical sign is greater than or equal to zero. This fact is a mathematical manifestation of the condition mentioned earlier that matrix **K** is rigorously positive definite. If during the solution a square root of a negative number appears, then this implies that the matrix was not positive definite.

8.1.1.2 ITERATIVE METHODS

In addition to *direct* methods for solving systems of linear equations, there are also *iterative methods*. In iterative methods, in contrast to what occurs with direct methods, the number of operations necessary to arrive at the result is not known *a priori*. The solving procedure is considered to end with the achievement of convergence, that is, to the minimisation, within a predetermined limit, of error. There is some debate about the exact definition of error, since in general this makes use of a norm for the vector error **e**, which can be calculated as

$$\mathbf{e}=\mathbf{K}\underline{\mathbf{u}}-\mathbf{p},$$

where $\underline{\mathbf{u}}$ is an approximation of the "true" vector **u**. Generally speaking the norm of the vector **e** is established within a certain tolerance, and the achievement of this condition leads to the achievement of convergence.

Iterative methods require a certain number of matrix-vector products of the order of the problem in question. Given that the matrix always remains the same, and that there is no reordering of its terms, iterative methods are especially suitable for techniques like *sparse matrix technology* (cfr. par. 8.1.2), which makes it possible to reduce the amount of memory required to an absolute minimum. In general, iterative methods are most suitable when there is a large number of equations. These methods can also be used to refine the solutions obtained with one of the direct methods ("cleansing" them of rounding errors).

8.1.1.2.1 THE GAUSS-SEIDEL METHOD

The Gauss-Seidel method is a refinement of another iterative method, Jacobi's method. Stiffness matrix **K** is written as the sum of three matrices: matrix **L**, lower triangular, which takes the terms under the main diagonal of **K**; matrix **D**, diagonal, which takes the diagonal terms of **K**; and matrix **U**, upper triangular, which takes the terms above the main diagonal of **K**; that is, bearing in mind that **K** is symmetrical, and thus $\mathbf{U}=\mathbf{L}^\mathbf{T}$

$$\mathbf{K}=\mathbf{L}+\mathbf{D}+\mathbf{L}^\mathbf{T}.$$

The Gauss-Seidel method consists in taking the succession of vectors **u**

$$(\mathbf{L}+\mathbf{D})\mathbf{u}_{k+1}=-\mathbf{L}^\mathbf{T}\mathbf{u}_\mathrm{k}+\mathbf{p}$$

where *k* is the index of iteration. Calculating $\mathbf{u}_{\mathrm{k}+1}$ starting from $\mathbf{u}_\mathrm{k}$ requires the forward solving of a triangular system, and thus it is extremely easy to do; the known term is obtained by multiplying the result of the previous iteration by $\mathbf{L}^\mathbf{T}$, and then summing **p**. The iterative procedure comes to a halt when the following condition is met:

$$\frac{\left\|\mathbf{u}_{\mathbf{k+1}}-\mathbf{u}_\mathbf{k}\right\|_2}{\left\|\mathbf{u}_\mathbf{k}\right\|_2}<tol$$

where the symbol used for the norm of the vectors indicates "norm 2" (the square root of the sum of the squares). In practice, the iterative procedure becomes blocked when two vectors corresponding to two successive iterations are practically identical (within pre-established limits of tolerance).

If the initial stiffness matrix **K** is symmetrical and positive definite, we can see that the Gauss-Seidel method converges for each initial choice of the trial vector $\mathbf{u}_0$.

8.1.1.2.2 The preconditioned conjugate gradient method

The preconditioned conjugate gradient method is a very general methodology that minimises the quantity

$$\Pi = \frac{1}{2}\mathbf{u}^{\mathrm{T}}\mathbf{K}\mathbf{u} - \mathbf{p}^{\mathrm{T}}\mathbf{u},$$

that is, the total potential energy, when **K** is presumed to be symmetrical and positive definite.[2] The vector $\underline{\mathbf{u}}$ that minimises this property is such that

$$\mathbf{K}\underline{\mathbf{u}}=\mathbf{p}.$$

We begin with an initial estimate of the solution $\mathbf{u}_1$, then determine the residual $\mathbf{r}_1$ as

$$\mathbf{r}_1=\mathbf{p}\text{-}\mathbf{K}\mathbf{u}_1.$$

We then set $\mathbf{s}_1=\mathbf{r}_1$.

Once this is done an iterative procedure in k (which is now equal to 1) is started:

$$\alpha_k = \frac{\mathbf{r}_k{}^{\mathrm{T}}\mathbf{r}_k}{\mathbf{s}_k{}^{\mathrm{T}}\mathbf{K}\mathbf{s}_k}$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k\mathbf{K}\mathbf{s}_k$$

$$\beta_k = \frac{\mathbf{r}_{k+1}{}^{\mathrm{T}}\mathbf{r}_{k+1}}{\mathbf{r}_k{}^{\mathrm{T}}\mathbf{r}_k}$$

$$\mathbf{s}_{k+1} = \mathbf{r}_k + \beta_k\mathbf{s}_k\,.$$

This comes to a halt when a suitable condition of convergence is reached.

If there are no errors in the floating point operations, then in matrices that are symmetrical and positive definite the procedure will converge in a number of steps equal to n degrees of freedom. Because of rounding errors, in reality the procedure is not direct but iterative, and it is not possible to know *a priori* how many iterations are necessary to achieve convergence. The lower the condition number of the matrix is, the faster the speed with which it converges; in physical terms, the speed of convergence increases as the quasi-mechanisms diminish (that is, when small or extremely small forces correspond to large displacements). To increase the speed of convergence, the *preconditioned* version of the algorithm

[2] A generalisation of the method that can also be used for matrices that are not symmetrical and not positive definite is that of preconditioned biconjugate gradients. However, this lies outside the scope of our discussion.

is often used, that is, the original system of linear equations is multiplied by a suitable matrix **A**, called the preconditioner:

$$\mathbf{AKu}=\mathbf{Ap}.$$

One possible choice of a preconditioner might be the matrix **D**, obtained from the main diagonal of **K**. Other choices make suitable preliminary numerical investigation possible, with the aim of analysing the structure of matrix **K**.

As we can see, this method only requires matrix-vector products, or scalar products, and so it is especially efficient when used with techniques for storing sparse matrices, which we will discuss in the next section.

8.1.2 TECHNIQUES FOR STORING STIFFNESS MATRICES

We have seen that each finite element problem requires solving a system of algebraic linear equations of the form

$$\mathbf{Ku}=\mathbf{p}.$$

Now that we have discussed the methods used to solve the kinds of systems typically encountered in the FEM, let's look at the problem of the space requirements (that is, the amount of computer memory) that the solution of such systems implies.

As is well known, there are two kinds of computer memory: cache memory on the hard disk, and random access memory, or RAM. Access to cache memory is noticeably slower than access to RAM, and so solvers tend to want to store the stiffness matrix entirely in RAM (or as it is often written, *all in core*, in order to be able to perform rapid calculations.

Up until just a few years ago, the amounts of both cache memory and RAM were rather limited (my first computer, bought in 1989, had a 100Mb[3] hard disk, and a 20MHz processor, with 1Mb of RAM). Only towards the end of the 1990s would large amounts of RAM begin to be available on normal computers. Today we commonly find computers with several gigabytes of RAM and hard drives of 100 gigabytes and more.

So here is the question: can the stiffness matrix **K** be entirely stored in RAM during the fundamental phase of reduction, and then during forward or backward solving?

Stiffness matrix **K** is square, and if n is the number of degrees of freedom in the problem then **K** contains n^2 real numbers. In today's computers, each real number in double precision requires 8byte for storage. Thus the memory required in to deal with the stiffness matrix is equal to $8n^2$. Table 8-1 shows what this means as the number of degrees of freedom increases.

[3] Mb = megabyte; 1kb = 1024byte; 1Mb = 1024^2byte = 1,048,576 byte; 1Gb=1 gigabyte; 1Gb=1024^3 bytes=1,073,741,824 bytes.

Number of Degrees of Freedom	**Number of bytes (approximate)**
100	80KB
1000	8Mb
10,000	800Mb
100,000	80Gb
1,000,000	8000Gb

Table 8-1: Nominal memory requirements per number of degrees of freedom to store stiffness matrix K.

As we can see, the amount of memory required to store the stiffness matrix increases quite drastically, so that a problem that has 10,000 degrees of freedom, not at all uncommon in technical practice today, becomes almost impossible to manage even with modern personal computers. Further, to the memory required to store the stiffness matrix we have to add the memory necessary to hold the vectors of equivalent nodal forces, the software program itself, the operating system and any other programs that might be running at the same time.

The stiffness matrix is the coefficient matrix of the system of linear equations and has the following characteristics:

1. It is symmetrical;
2. It is sparse;
3. It is banded;
4. It is (or should be) positive definite.

The fact that the stiffness matrix is symmetrical allows us to say right away that in reality we don't have to store n^2 terms (where n is the number of degrees of freedom) but only about half (the terms on the diagonal have no corresponding values). More precisely, NR1, the number of terms to be stored, turns out to be

$$\mathrm{NR1} = n^2 - (n^2 - n)/2 = (n^2 + n)/2$$

The fact that the stiffness matrix is banded means that non-null terms are actually found only up to a certain point in a given row, starting with the diagonal.

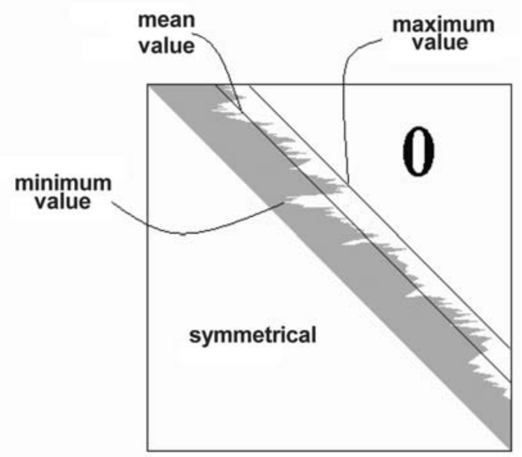


Figure 8-2: What the stiffness matrix looks like: maximum, minimum and mean values.

This has an important consequence. The number of *non-null* terms that has to be effectively stored is not equal to NR1, but is actually much smaller: all of the terms that lie beyond the maximum bandwith are certainly null, and thus it is not necessary to store them.

Thus, if BMAX is the maximum width of the half-band (*half*, because the matrix is symmetrical and the other half, being equal, does not interest us), the second approximation of the quantity of numbers to be stored, NR2, can be written as

$$\text{NR2} = n * \text{BMAX}$$

This second scheme makes it possible to save quite a bit of space in memory, because the maximum band is only a fraction of the order of the matrix. To be precise, the maximum band can be reduced to a fraction that is often less than 1/10, 1/20, or even 1/40 of the order of the matrix, thanks to the techniques of renumbering, which we will discuss in the following section.

The scheme using the maximum bandwidth to determine the quantity of numbers to be stored is used frequently because it makes it rather easy to improve the situation and does not lead to significant difficulties from a programming point of view.

However, this scheme can still be improved. Rather than storing rows that all have the same dimension (BMAX), it is possible to store each row with its exact dimension, and a vector of integer numbers that memorises, for each row, the exact dimension of that row. Thanks to this expedient, the quantity of numbers to be stored is:

$$\text{NR3} = n * \text{BAVE}$$

where BAVE is the *mean* band dimension of the stiffness matrix. Memorising each row with its exact dimension is the same as memorising each column with its exact dimension (given that **K** is symmetrical); since the columns are vertical, their differing heights recall a city skyline, and thus this technique is called “skyline” (fig. 8-3).

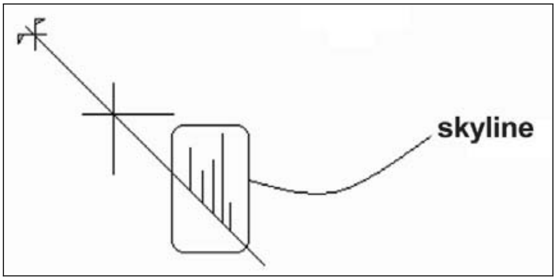


Figure 8-3: A “skyline” diagram of a matrix.

The skyline technique makes possible another drastic reduction in the number of terms to be stored, because the relationship between the maximum and minimum band values can be quite large (3, 4, 10 or even more, depending on the problem). Not using the skyline technique can result in taking up five or ten times more memory that what is actually necessary.

If R is the relationship between the mean band value and the order of the system *n*, the size in bytes of the stiffness matrix becomes

$$8*R*n^2$$

Table 8-2 shows the expected sizes in bytes of the stiffness matrix for various values of maximum and minimum band values (in the case of skyline matrixes).

As Table 8-2 shows, using the skyline techniques makes it possible for the solver to achieve a solution “all in core” for problems of the order of tens of thousands of degrees of freedom. This dimension (with ten or twenty thousand degrees of freedom) is often amply sufficient, especially for studies of frame systems which don’t require meshes that are particularly dense.

However, it can happen that in spite of these reductions the stiffness matrix is still too big to be stored in memory. In this case the technique used by the solvers involves dividing the matrix into blocks.

The solution is loaded into RAM two blocks at a time: the block to be reduced, in triangular form, and the block of the previous equations, already reduced, which is required to reduce the current block. The two blocks can have different dimensions, using a single block “bordered” by only those previous equations that are strictly necessary because of pairing, making it possible to optimise the use of memory. Each block has to be read and written to the hard disk, and this causes interruptions in the solving process.

n **Degrees of Freedom**	**BAVE 3% *n***	**BAVE 6% *n***	**BAVE 10% *n***	**BAVE 15% *n***
100	2,4Kb	4,8Kb	8Kb	12Kb
500	60Kb	120Kb	200Kb	300Kb
1000	240Kb	480Kb	800Kb	1,2Mb
5000	6Mb	12Mb	20Mb	30Mb
10000	24Mb	48Mb	80Mb	120Mb
50000	600Mb	1,2Gb	2Gb	3Gb
100000	2,4Gb	4,8Gb	8Gb	12Gb
500000	60Gb	120Gb	200Gb	300Gb

Table 8-2: Space requirements in memory for stiffness matrix K using the skyline technique.

It is quite common for a matrix to be divided into 2, 4, 8, 12 or even more blocks, but it is incorrect – or at least, not optimal – to divide it into dozens or hundreds of blocks. In fact, in this case the frequent interruptions during the reduction phase, and the continual reading and writing to disk, constitute an inordinate waste of time, and generally speaking, lengthen the calculation time significantly, sometimes to an impossible extent. When this happens, it means that the system resources (that is, the RAM) are not sufficient to solve the problem in question.

To sum up, when the system does not have the resources necessary to carry out an optimal solution, we might have to accept a solution based on a subdivision of the initial matrix into quite large numbers of blocks, but this kind of solution is much less preferable than one that is all in core or just adequately subdivided, where this can be accomplished by increasing the memory resources of the system (see §8.1.4.1).

♣

Even though skyline techniques have resulted in a significant reduction of requirements in terms of memory and have been used for years, the growing number of required degrees of freedom have led to the need to develop techniques aimed at reducing still further the space required to store the stiffness matrix. These techniques, called *sparse matrix technology*, exploit the fact that even the zone inside the band of the matrix contains zeroes. It is thus possible to think of storing only the terms that are effectively non-null in a single, large vector, accompanied by a system intended to reconstruct the original position of a generic element of the vector itself, as well as identifying the position of the vector in a given element of the stiffness matrix (by means of the so-called *double linked list*). The advantages in terms of computation are considerable: only the non-null terms, wherever they might be found, are stored. The number used in these cases to determine the necessary amount of memory is not the bandwidth but rather the percentage of fill in, which grows as the coupling of the equations grows: the higher on average the number of finite elements connected to a node, and the more complete their contribution to stiffness, the greater the number of non-null terms in the stiffness matrix.

This results in a drastic decrease in calculation times and in the amount of RAM occupied, so much so that today sparse matrices are the most commonly used.

This topic could easily fill a whole volume by itself, so we cannot discuss it in depth, but in what follows we will clarify some of the aspects most helpful for managing the use of these solvers.

The most important observation is that, as we have seen, not all solving methods are equivalent. Of particular importance is whether or not the matrix can be positive semi-definite, that is, if it can contain singularities. As we have seen, the Cholesky method has the advantage of blocking the solving and making it clear when the matrix is only positive semi-definite, while the other methods limit themselves, in this case, to not guaranteeing convergence. It would thus be particularly interesting to be able to use the Cholesky method for sparse matrices, but not all sparse solvers adopt this method, because it requires a triangularisation that is far from simple when working with sparse matrices. It is by no means given that if the original stiffness matrix **K** is highly sparse, then so is matrix **L**, and it is even less certain that **L** is of the same identical type. It can in fact be that **K** is highly sparse and that **L** is not at all: in this case a preliminary treatment is required for matrix **K** in order to obtain a matrix **L** that is sparse enough to stay all in core. Even if the procedure used to optimise the problem is totally different than the one used for non-sparse matrix techniques (that is, the renumbering that will be discussed in the next section), it is clear that this preliminary phase also tends, as in the classic techniques, to interfere with the algorithm's efficiency, requiring additional time.

If the sparse matrix solver adopted by the program does not come with a feature that clearly signals when there are mechanisms or, more generally, singularities, then the user has to be aware of this and ready to guard against it in some way: the lack of this information can be dangerous, since a singularity in the coefficient matrix always indicates that a problem has been poorly formulated either structurally or numerically.

With the amounts of RAM currently available, it is probable that even a classic skyline solver is capable of solving systems with tens of thousands of linear equations in short spaces of time. Given this situation, it is preferable to choose solving methods that effectively guarantee that stiffness matrix **K** is positive definite, rather than methods which, although they are capable of solving the system a few seconds quicker, provide no such check.

The situation obviously changes if we are dealing with systems that have some hundreds of thousands of equations, or even more. In this case the skyline methods won't work, and sparse matrix solvers that are sufficiently stable and capable of identifying singularities are required.

8.1.3 Renumbering

In the preceding section we saw that the space in memory necessary for storing the matrix increases in direct proportion to either maximum bandwidth (the rectangularisation method) or mean bandwidth (skyline method).

The bandwidth is directly related to the way in which the nodes of a structure are numbered. Any given row of the stiffness matrix refers to a given degree of freedom of a certain node of the structure. When assembly was discussed in chapter 4, it was explained that the non-null terms located on a given row of the matrix referred to indirect stiffness, which were due in their turn to the presence of elements connected at different nodes.

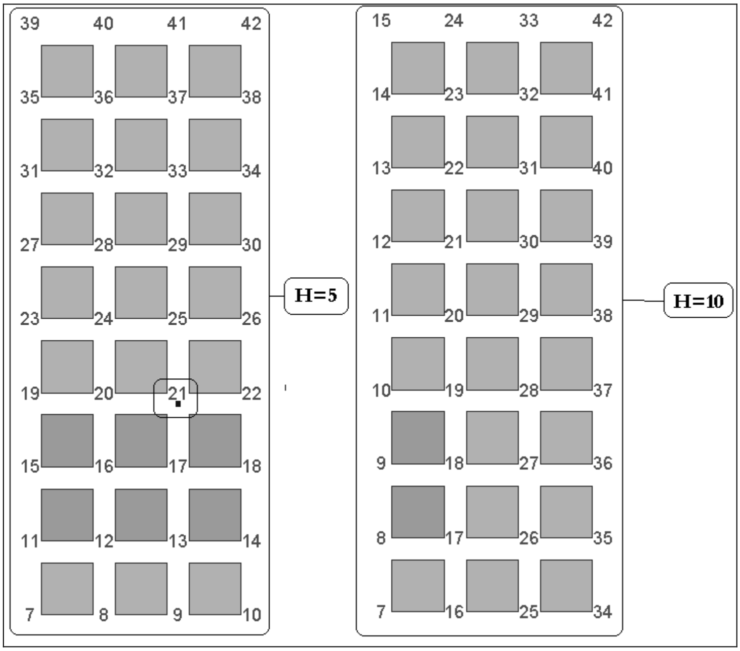


Figure 8-4: The effects of numeration on the normalised band.

To make this clear, let's look at the structure on the left in fig. 8-4. If, for example, we move node 21, keeping all the others fixed, we have to apply forces on some other nodes in order to ensure that they don't move. To be precise, we have to apply forces on nodes 16-17-18-20-22-24-25-26. This is because all the nodes listed are connected to node 21 by at least one element. No force is required to keep all of the other nodes fixed, and the indirect stiffnesses are identically null everywhere.

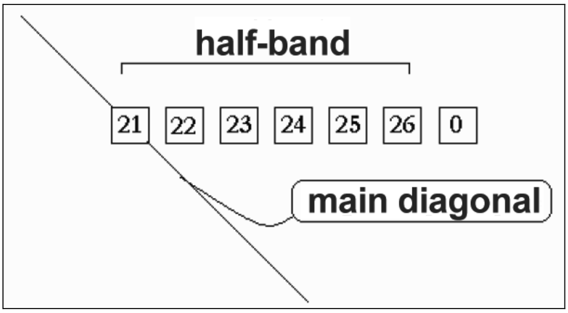


Figure 8-5: Blocks of direct and mixed stiffnesses on the row of node 21.

If we consider a stiffness matrix that is partitioned into small square blocks that refer to the degrees of freedom of each node, we can see that the presence of one mixed stiffness term 21-26 gives rise to the presence of one small non-null stiffness block that is five blocks away, starting from the diagonal term relative to node 21.

Let H be the maximum difference that can be obtained between the node number in question (21) and the number of any one of the nodes connected to the elements that are connected to node 21 (16-17-18-20-22-24-25-26). In our case

$$H=|21-16|=|21-26|=5.$$

If all the nodes have the same number of degrees of freedom, then a good approximation of the half-bandwidth can be obtained by multiplying H by the number of degrees of freedom per node. If the nodes do not all have the same number of degrees of freedom, we can expect to end up with remainders due to the nodes that are missing one of the degrees of freedom typical of the problem. In any case, H is a direct indicator of bandwidth: if H can be minimised, then the bandwidth is also minimised.

Looking at figure 8-4 we can see that a different numbering – all else remaining the same – is all that is needed to obtain a model where H is greater. The numbering of the structure shown on the right of 8-4 gives preference to the long side, and this leads to a value of H=10, that is, a normalised bandwidth that is the double of the structure shown on the left. What was node 21 in the structure on the left has become node 28, and since nodes 38 and 18 are connected to nodes 28 by elements, we can expect to have H=10, that is, double the previous value of H=5.

By choosing different node numbering it is possible to obtain different bandwidths, and thus increase or decrease problems of the amount of space required in memory during the solving phase.

Generally speaking the solvers begin their task by attempting to renumber the nodes of the structure so as to render bandwidth a minimum. From the very beginnings of the FEM, quite a number of studies have been undertaken to try to find systems that efficiently renumber the nodes of a structure.

Even if there is always an "optimum" numbering, it is not always possible to establish immediately just what this system is. If the structure has a side that is "long" (that is, one with more nodes, not necessarily the longest geometrically) with respect to other, shorter ones, it is usually helpful to renumber giving preference to the "short" sides, or better, to the "shortest" side (again, that with the fewest nodes). On structures that are highly regular (such as that shown in figure 8-4), it is easy to see that such numbering is also the "optimum" numbering. On irregular structures (those with steps or the like), it can be seen that this criterion, though still valid, does not necessarily lead to the optimum solution.

The algorithms for renumbering make use of graph theory, assigning a point value to each node on the basis of the number of elements or nodes that are connected to it. The nodes are thus arranged according to a hierarchy, and in general a path is constructed starting from an initial node, aimed at maximising or minimising the global point value of the "path". These algorithms are not always ideal, because the solution depends on the starting node, and it is practically impossible to try out each of the nodes to see which is the ideal starting node because this greatly increases calculation time.

Since the amount of time required for the renumberer to generate and try out a new numbering can sometimes be quite significant, the algorithms used to seek the best renumbering are not usually programmed to find the "optimum" numbering, but are rather limited to finding the best solution among a set of possible numberings, each of which is tested to determine bandwidth.

If the software program is equipped with the right commands, it is possible for the user to find on his own a numbering that is suitable for his purposes. In this case it is helpful to ask the solver to not execute any renumbering, so that no time is wasted in the useless search for a renumbering that is better than the one already in place.

It is quite clear that renumbering upsets the labelling of the nodes, reassigning a progressive number to them. Thus, when the solver gives messages about labilites or problems during solving, it gives them using the "new" numbering and not the "old". Some software programs limit themselves to giving cryptic messages such as

Error your model is not tied down enough-check dof 14326.

In such cases it is up to the analyst to find which node corresponds to DOF 14326 and what its original number was (for example, 2345), in order to locate it within the structure.

8.1.4 THE USER'S POINT OF VIEW

In this section we will look at the problem of solving from the point of view of the person using the software program, rather than that of the one who writes it or optimises it. We will briefly review some of the fundamental aspects that the user must bear in mind when working with the software programs currently in use.

8.1.4.1 PREDICTING HOW MUCH MEMORY IS OCCUPIED

We have seen that once he knows the method that his software program uses for storing stiffness matrix **K,** the user can arrive at a first estimate of the amount of space that will be taken up in memory by the stiffness matrix, and can estimate, at least roughly, the maximum or mean half-bandwidth, when skyline techniques are used.

Table 8-2 makes it possible to come up with a first estimate of the possible difficulty of the problem that has to be solved. If the computer being used has a certain number of megabytes of RAM, then it is possible to compare the amount of space required by the stiffness matrix to available RAM and understand what solving strategies are possible.

We can say right away that for a computer with one or two gigabytes of RAM, today widely available, problems of up to 50,000 to 100,000 degrees of freedom do not usually constitute a problem, as long as the computer is set up to work correctly.

For problems of a greater magnitude, the requirements in terms of RAM have to be evaluated and compared to the available memory so that it is possible to foresee – should it be necessary – how the computer has to be set up. If the stiffness matrix comes to take up 80 – 90% of the system's available RAM, then unnecessary multitasking should be avoided during solving, at least if optimal solving times are desired.

A good rule of thumb for determining *a priori* the size of the stiffness matrix is

$$\text{Mb}= 0.4\ \text{kdof}^2$$

where Mb are the megabytes required for the stiffness matrix, kdof the thousands of degrees of freedom, and where a mean band (for the skyline method) or a maximum band (for the ordinary method) has been implicitly presumed to be equal to 5% of the order of the problem. For example, for a problem with 100,000 degrees of freedom, a rule of thumb estimate of the space required for the stiffness matrix would be $0.4\text{x}100^2$=4000Mb. If our computer has, for instance, 2Gb of RAM, we can be certain that the system will be subdivided into blocks. Presuming that 1.5Gb of the available 2Gb of RAM are destined for the solver (that is, about three-quarters), we can expect that the stiffness matrix will be subdivided into two block. Of course, with a computer that has 4Gb of RAM the same problem could be solved *all in core* without any problem. It should be kept in mind that 4 Gb is the physical limit for addressing memory in 32bit systems, thus it is highly probable that 32bit systems will give way to 64bit systems in just a few years.

Can we use a computer with 500Mb of RAM to work on a problem that has 100,000 degrees of freedom? If we apply the formula given above, $0.4x100^2$=4000Mb, and allowing the use of almost all of the available RAM, of the order of 400 of the 500 total megabytes (which probably means that the run has to be launched at night), we can expect the stiffness matrix to be subdivided into ten blocks. Of course if a preliminary run guarantees us that the mean band is equal to 2% rather than 5% of the order of the problem, then the required amount of memory becomes 2/5 of that first estimated, that is, 2x4000/5=1600Mb, which would make only 4 blocks necessary. But this still constitutes a "big" problem for the machine we have at our disposal.

8.1.4.2 CORRECTLY SETTING UP THE COMPUTER

On the computers most widely used, that is, those based on the 32bit Windows platform, it is necessary take into account a series of factors that can significantly influence how solvers perform. Here we will list the most important of these.

8.1.4.2.1 MULTITASKING

Multitasking is the ability of an operating system to execute more than one program at the same time. This important capacity was achieved by mass-produced Microsoft operating systems for the first time with Windows 95. Running other programs at the same time as a solver is of course completely normal, as long as the solver does not require a percentage of the available RAM that is close to 100%. In such a case, multitasking can become impossible. If the user has decided to sacrifice some of the speed of solving in order to continue to perform other tasks at the same time, then it is helpful – in the programs where this is possible – to allocate less RAM for the functions related to the solver, so that it occupies less than 60% of available RAM.

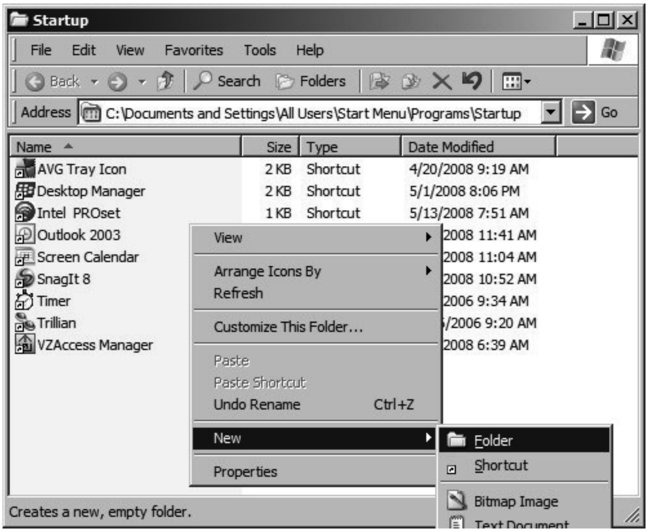
On a practical level, if the problem is "big" for the computer being used, then it makes sense to close the applications that are not strictly necessary.

8.1.4.2.2 VIRTUAL MEMORY

In simple terms, virtual memory is memory on the hard drive that is usually used by the operating system to transfer part of the data contained in RAM that is temporarily unnecessary. Given the enormous capacity of present-day hard drives, the increase of virtual memory would appear to be a valid system for getting around the limitations of available RAM. In reality this is not the case, since the availability of this kind of memory does not do away with the need for frequently swapping data[4] on the disk. In fact, since swaps are generated by the operating system in a way that is unpredictable, and not in ways that concord with the solver that partitions the stiffness matrix into blocks, it is possible that, by increasing the available virtual memory and the "RAM" requested of the operating system, solving will take even longer than it would have had less RAM been requested and an "honest" partitioning into blocks been performed. It thus makes sense to use virtual memory in the way it is normally used, without trying to "fool" the system.

8.1.4.2.3 AUTOMATIC EXECUTION

Some software programs allocate the memory that the operating system says is available at the moment that the procedure of solving is executed as a memory pool for the solution. It is as though the solver knocks at the door of the operating system and asks, "How much memory is free?" At this point very surprising things can occur.



Address: C:\Documents and Settings\All Users\Start Menu\Programs\Startup

Name	Size	Type	Date Modified
AVG Tray Icon	2 KB	Shortcut	4/20/2008 9:19 AM
Desktop Manager	2 KB	Shortcut	5/1/2008 8:06 PM
Intel PROset	1 KB	Shortcut	5/13/2008 7:51 AM
Outlook 2003			2008 11:41 AM
Screen Calendar			2008 11:04 AM
SnagIt 8			2008 10:52 AM
Timer			2006 9:34 AM
Trillian			/2006 9:20 AM
VZAccess Manager			2008 6:39 AM

Creates a new, empty folder.

Figure 8-6: Finding the contents of the "Start Menu" folder in Windows.

With Windows 9x operating systems, it can happen that in a computer with 500Mb of RAM and no programs running except the solver, there are only 20Kb of memory available, according to the operating system. It is clear that if the solver attempts to solve even a very banal problem with 500 or fewer degrees of

[4] By "swapping" we mean the moving of information between RAM and the hard drive; "swapping in" occurs when data is moved onto the hard disk; "swapping out" occurs when data is moved into RAM.

freedom, there is not even enough memory available to allocate the initial vectors, causing the solution procedure to lock up.

A search for the possible causes leads to the conclusion that ample amounts of RAM are reserved by some applications that are typically inserted in "automatic execution" even when they are not running. In order to overcome this problem, we need only eliminate all the applications that are not necessary (such as the Office toolbar) from the "automatic execution" menu. In this author's opinion, the only thing that needs to remain in automatic execution is the antivirus programs.

To find where programs in automatic execution are listed and eliminate them from the startup menu, follow the instructions that come with the computer's operating system (of which Microsoft alone has many in circulation today, such as NT™, XP™, Vista™, etc.).

8.1.4.2.4 THREAD PRIORITY

One method that can be used to increase the speed with which an analysis is performed is that of increasing the priority of the process (*thread*) connected to the solver. This method should be used carefully because increasing the priority of a thread can damage how other threads work. A plausible exception might be to increase the priority of a thread if the computer is to be dedicated entirely to that particular thread.

8.1.4.2.5 THREAD FREQUENCY

The amount of available memory that can be assigned to a process can vary as a function of the frequency with which that process is performed. When a program is run more than once you will notice that the situation in terms of memory can change. This is because the operating system "makes room" for applications according to how important these prove to be for the user.

8.1.4.2.6 DEFRAGMENTATION OF THE HARD DISK

In order to speed up the execution of solving (especially if it is necessary to do a run that requires writing a large amount of data to disk or the partitioning of the stiffness matrix into blocks), then it is absolutely worthwhile to make sure that the hard disk does not need to be defragmented. Defragmentation (or more commonly, defrag) of the disk usually leads to a noticeable improvement in performing reading and writing.

8.1.4.3 MAKING SOLVING FASTER

To summarise, in order to make solving faster the user can follow these steps as a guide:

- Free up RAM by eliminating the unnecessary automatic execution of unused programs;
- Free up RAM by reducing multitasking;
- Increase the thread priority connected to the solver;
- Defragment the hard disk.
- For exceptionally large problems it might be helpful to attempt renumbering based on the order of the nodes, and to seek the renumbering that results in the optimal bandwidth for the problem at hand;
- Eliminate from the beginning any inactive degrees of freedom, without asking that they be eliminated automatically;
- Make use of symmetries or antisymmetries to reduce the number of nodes;
- Create a mesh that is not homogeneous, that is, increase the mesh density only where significant variations in stress are expected;
- Reduce the load cases to those that are strictly indispensable, and use load combinations.

8.2 STRESS RECOVERY

8.2.1 HOW STRESS RECOVERY WORKS

Stress recovery is the phase in the FEM in which, the displacements of the nodes of the structure having been found, the stresses inside the elements are calculated. From a formal point of view, stress recovery is performed by the following steps:

1. The nodal displacement vectors $\mathbf{u}_{\mathbf{eg}}$ of each element are created in the global reference system, by disassembling the vector of overall nodal displacements $\mathbf{u}$. *Formally* we can write $\mathbf{u}_{\mathbf{eg}}=\mathbf{D}_{\mathrm{e}}\mathbf{u}$, *but in practice* the vector $\mathbf{u}_{\mathbf{eg}}$ is obtained simply by reading the relevant displacements and filling in vector $\mathbf{u}_{\mathbf{eg}}$ in a suitable way;
2. The nodal displacements of the global system are transformed to the local system: $\mathbf{u}_{\mathrm{e}} = \mathbf{T}\ \mathbf{u}_{\mathbf{eg}}$ (cfr. chapter 4). This phase involves calculating a matrix-vector product (or better, a series of matrix-vector products made in blocks; see §9-14);
3. The *astrological principle* is applied to determine the displacements within the elements ($\mathbf{U}=\mathbf{\Phi}\ \mathbf{u}_{\mathrm{e}}$), and from these displacements the congruence relation is applied: $\boldsymbol{\varepsilon}=\mathbf{CU}=(\mathbf{C}\ \mathbf{\Phi})\mathbf{u}_{\mathrm{e}} = \mathbf{B}\ \mathbf{u}_{\mathrm{e}}$ (cfr. chapter 4). This phase involves another matrix-vector product;

4. Finally, the relationship between the strains and the stresses is applied: $\boldsymbol{\sigma}=\mathbf{D}\boldsymbol{\varepsilon}$, that is, once the strains in a given point have been determined, the stresses are determined by means of another matrix-vector product.

During the third step it is necessary to understand that strain $\boldsymbol{\varepsilon}$ is evaluated for each pertinent point P by evaluating matrix **B** at that point; if we want to know the stresses in point P, we can write:

$$\boldsymbol{\sigma}(\mathbf{P})=\mathbf{D}\mathbf{B}(\mathbf{P})\mathbf{u}_e.$$

8.2.2 Interpolation, extrapolation (local smoothing)

Although in principle it is possible to apply the rules we have described to calculate the stresses in each point P, a different procedure is usually followed for reconstructing the set of stresses on an element; this is because the most rigorous and consistent approach is relatively costly in terms of the amount of time required for calculation. Rather than calculate the stresses in all the relevant points, normally the stresses are calculated in a few significant points, and then the trend for the entire element is reconstructed by means of techniques of interpolation or extrapolation (including the method of least squares). It should be noted that the interpolating field is not always consistent with the presumptions made about the displacement field, and thus the stresses predicted for a generic point P of the interpolation are usually different from those predicted in the previous section.

The relevant points that are usually used are the Gauss points (which were mentioned when the numerical calculation of the integrals was discussed) and the nodes of the element.

One possible technique consists in determining the stresses in the nodes and then interpolating them for the rest of the element. For example, on an element with three nodes, for the generic stress s in the local reference system of element (x,y) we can assume an interpolating function of the type

$$s= ax+by+c$$

which, since the stresses of the three nodes are known (by means of the most rigorous techniques), this makes it possible to determine (a, b, c) and thus provides us with a closed formula aimed at predicting the stresses in the other points. It is worthwhile to note once again that this kind of linear presumption might not be consistent with the presumed linearity *of the displacements*, since strictly speaking the stresses are proportional *to the strains*, which depend on the derivatives of the displacements. Thus we can expect stresses that are *constant*, not linear.

A second possible technique consists in determining the stresses in the Gauss points and then extrapolating the values of the stresses in the rest of the element with an estimate using the method of least squares. This is the technique used in the author's own FEM software program, SARGON. As Bathe says,

> ...when using the direct stress evaluation..., the stresses are frequently more accurate at the numerical integration points used to evaluate the element matrices [i.e., the Gauss points] than at the nodal points. Hence, for a least squares fit, it can be of value to use functions of order higher than that of the stress variations obtained from the assumed displacement functions because in this way improved values can be expected [Bathe 1996: 257].

What should be made clear here is that when the stress fields are reconstructed, new simplifying hypotheses are introduced, and these give rise to reliable results only if the mesh is sufficiently dense; if it is not, then the approximation is too rough. In order to generate the long awaited coloured map, the software program has to assume simplified laws of stress variations, which might not be justified, especially when there are marked variations between the stresses in the sample points.

8.2.3 INTERELEMENT JUMPS

Up to now we have discussed the reconstruction of the stresses in a single finite element, but what happens when we want to compare the stresses of two adjacent elements? Strictly speaking, if there are no material or geometrical discontinuities, then the stresses in a continuum are continuous. In practice, however, we find that along the common edges of two adjacent finite elements the predicted stress values are different. We find also that when we have nodes common to two finite elements, or we have several elements, the predicted stresses in each element differ from those predicted by the others. Thus it happens that the stress field calculated by a finite element software program is not continuous but "jumps" as we go from one element to another. These so-called *interelement jumps* can assume significant values, especially if the mesh is not sufficiently dense.

Even though the equilibrium equations take all of the elements into account, the stress field predicted by the FEM for a given element uses the nodes *of that particular element*, and only indirectly in correlation with the stress field predicted for another, adjacent element that has different nodes. The stress field of an element depends on *all* of its nodes, not only on those in common with the other elements adjacent to it.

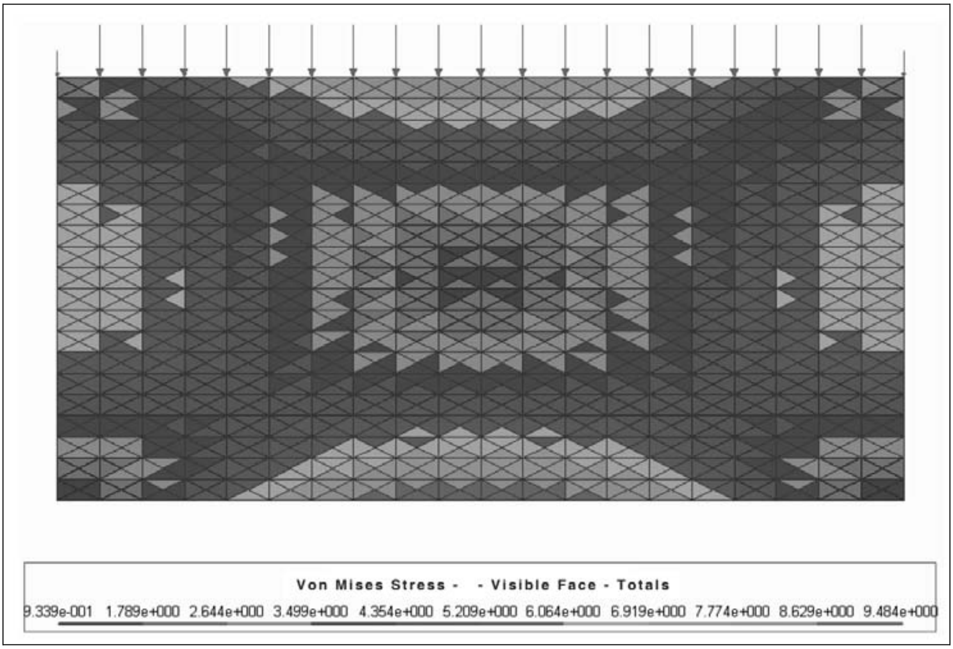

Von Mises Stress - - Visible Face - Totals
9.339e-001 1.789e+000 2.644e+000 3.499e+000 4.354e+000 5.209e+000 6.064e+000 6.919e+000 7.774e+000 8.629e+000 9.484e+000

Figure 8-7: Von Mises stress map in a plane stress problem (without smoothing): note that the stress in each triangular element with three nodes is constant (map generated with SARGON).

Quantifying interelement jumps and representing them graphically is one fundamental way to obtain information about the reliability of the analysis performed. Jumps that are too high, that is, high enough to make recognition of the isolines of the stress field difficult, are a sure sign that the analysis is unreliable; it should be redone with a more refined mesh. It should be noted that the convention of measuring the precision by taking the ratio of the maximum value predicted by the model and the maximum jump is absolutely unreliable, because due to the concentration of stresses, the maximum value predicted by the model is not at all realistic. Instead, it is necessary to take the ratio between the jump at a node and the value predicted *for that node*, not between the maximum jump and the maximum value, which are registered in different points, not in a single point.

An especially interesting graphic representation is the one that directly shows the amount of the interelement jumps, since this allows us to understand where the mesh has to be made more dense, and to quantify in real terms the error that we risk making.

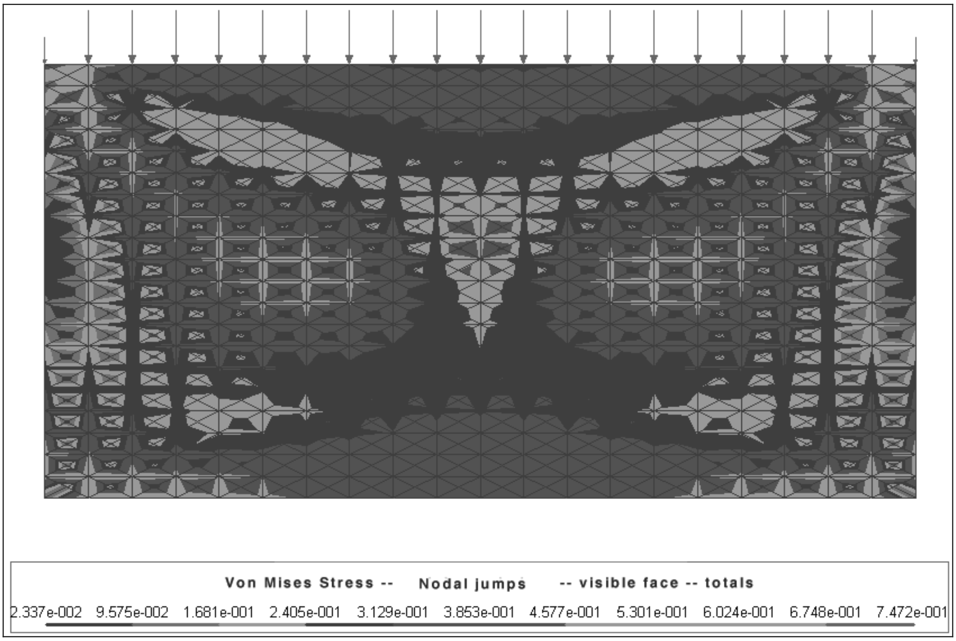


Figure 8-8: Graphic representation for the problem shown in fig. 8-7. The maximum jump is equal to 0.74N/mm^2 at the upper corners (map generated with SARGON).

The interelement jumps can be defined as max{s$_i$} – min{s$_i$}, where *i* varies between 1 and *n*, with *n* the number of elements that conjoin in the node. It can also be defined by other formulas, for example, as the average quadratic remainder of all {s$_i$} in the node in question; this is the technique used in the author's own FEM program, SARGON. In this case, if *m* is the average value of all {s$_i$}, then remainder *q* will be

$$q = \sqrt{\frac{\sum_{i=1}^{n}(s_i - m)^2}{n}} .$$

For example, figure 8-8 shows that the maximum jump, equal to 0.74N/mm^2 is found in the areas where a stress value of about 2 was predicted, and thus the error made there is large (30-40%). However, if in that area the stress is not determinant for our purposes, we might decide that that amount of jump is acceptable. At the extrados and intrados we find jumps of around 0.3 N/mm^2, but in that area the Von Mises stress is of the order of 9.4N/mm^2, with an error equal to "only" 8%. Obviously, this kind of analysis does not have the quality of a *precise* analysis; it could be refined so as to reduce the error to less than 1%, at least in the areas of greatest stress.

This discussion makes it clear that a finite element analysis with its pretty coloured maps is inherently a myth. It is up to the analyst to prove that his or her analysis is reliable by providing a measurement of the error committed by the model. As Bathe says authoritatively, "We emphasize that the calculation of an error measure and its display is a most important aspect of a finite element solution. The quality of the finite element stress solution ... should be known" [Bathe 1996: 258].

8.2.4 GLOBAL SMOOTHING: THE RISKS OF COSMETICS

The problem of interelement jumps is normally solved by means of techniques of *smoothing* (regularisation, elimination of jumps). If *n* elements of the same kind conjoin at a generic node, and given a certain stress measure *s*, then the finite element analysis will lead to *n* different values of *s* to be dealt with. The simplest technique for eliminating the problems related to jumps is that of substituting the *n* values of *s* predicted by the *n* elements with a single value $\underline{s}$ obtained by taking the simple average of the values of the individual elements:

$$\underline{s} = \frac{\sum_{i=1}^{n} s_i}{n} .$$

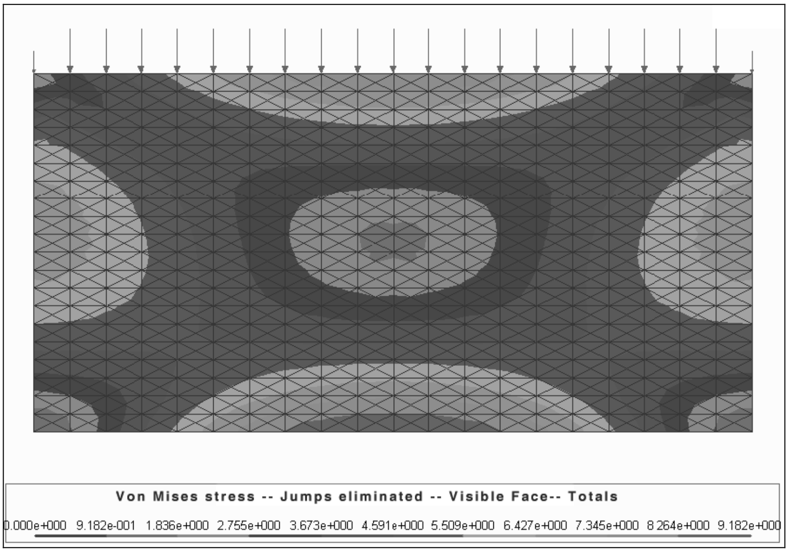

Von Mises stress -- Jumps eliminated -- Visible Face-- Totals
.000e+000 9.182e-001 1.836e+000 2.755e+000 3.673e+000 4.591e+000 5.509e+000 6.427e+000 7.345e+000 8.264e+000 9.182e+000

Figure 8-9: Graphic representation of the stresses with jumps eliminated (smoothed). Compare with the unsmoothed representation shown in figure 8-7 (map generated with SARGON).

If this is done, then the stress values become continuous between elements, and the interelement jumps disappear. By adopting identical interpolating functions for the various elements (often the same ones used for the displacement field) and starting from the unequivocal nodal values, the field has to turn out continuous.

There are two advantages to this kind of smoothing, called *global smoothing* because it takes place among the elements rather than within an individual element. First, the nodal values obtained by taking the average are more reliable than the values obtained from individual elements. Second, averaging the nodes eliminates the interelement jumps and regularises the stress field estimated by the calculations. It should be noted, however, that while the average is more reliable than individual values, absolute reliability still cannot be taken for granted: this depends on the interelement jumps.

It is so tempting to make any possible trace of discontinuity in the stress field disappear that we very rarely see any representations that are not smoothed. The analyst can show beautiful coloured maps, the software program proves its worth, and everyone is happy.

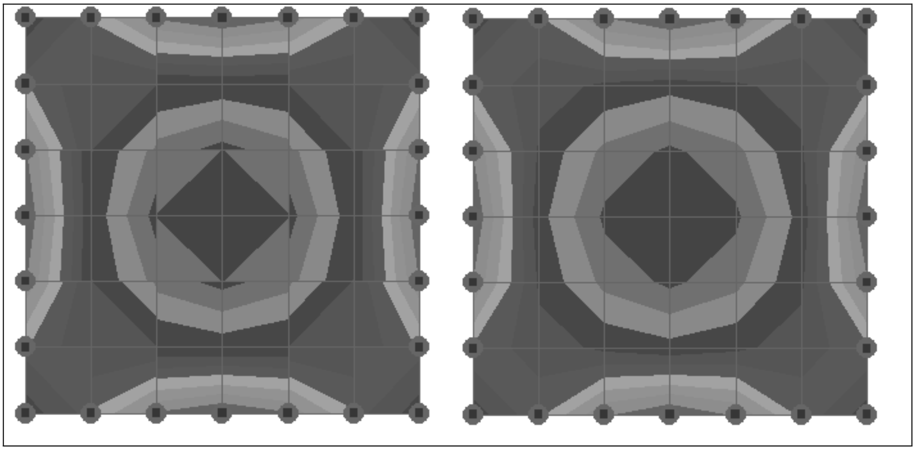


Figure 8-10: Comparison between unsmoothed (left) and smoothed (right).

In reality this disappearing act is extremely risky, because if it is applied to a problem where there are significant interelement jumps, the result is an image that is in fact beautiful and regular, but also completely unreliable. An serious and professional analysis has to include maps of interelement jumps as well, including maps without any global smoothing at all, in order to make it possible to determine at any moment how reliable the calculation is.

9 ERROR ASSESSMENT

9.1 THE ANALYST AND ERROR

This chapter is dedicated to error. Error is so common in finite element analyses that we might say that at least half of the task of a good analyst consists in searching for possible errors.

There is no doubt that an experienced analyst makes errors far less frequently than a beginner does, but even the most expert analyst has to accept the real possibility that errors will be found. In the best case, the analyst is a kind of detective and begins by assuming that his model is sure to contain errors unless proven otherwise: only after the model and its results have been submitted to an appropriate number of checks will he begin to believe that the results obtained are reliable. No finite elements software program – not even the most costly – can be seen as a kind of enormous machine capable of producing faultless results at the click of a key. Because no such program exists, the analyst has to be able to discern between reliable and unreliable results, and locate probable errors in the model and in the software.

After having produced any kind of analysis, the expert analyst will thus perform a series of checks in order to diminish the probability of there being any problems in the model.

There are essentially three main possible sources of problems: an insufficient density of the mesh and consequently the insufficient precision with which the solution was calculated; errors arising during the course of modelling; and errors in the software program.

In today's practice, all three of these sources are important. Even the last, which might seem to be academic, is actually quite significant, given the kind of technology currently used in the development of software and testing, *even in the cases of well-known and certified programs.*

9.2 DETERMINATION OF NUMERICAL ERRORS

9.2.1 CHARACTERISTICS OF CONVERGENCE IN THE FINITE ELEMENT METHOD

A first possible source of errors lies in the insufficient precision with which the differential equations used to represent the problem have been solved. There are many causes for a lack of precision, but the leading and most frequent one is related to the insufficient density of the mesh. We have seen that the FEM introduces various simplifying presumptions, in some of which the simplifications are quite significant indeed, and we noted that the propensity of

Structural analysis with finite elements
978-0-7277-4093-9

these hypotheses to generate errors tends to decrease as the dimensions of the finite elements used to model the problem decrease.

The speed and the type of convergence of course depend on the type of problem (the differential equations that have to be solved), on the type of finite elements used (for example, the higher order elements generally converge more rapidly), on the problem's particular geometry and on the particular loads (abrupt gradients of stress are usually caused by an insufficient discretisation, at least locally). Although the range of variation is rather broad, there are still general rules that regard convergence in the FEM that should be known in order to model structures correctly.

In concise terms, we can say that the convergence of the numerical solution towards the "exact" solution is monotonic (that is, always increasing) if the finite elements are *compatible*, if the formulation is *complete*, and if successive meshes *constitute a regular succession.*

The requisite of compatibility requires that there be no areas in the model where gaps or overlaps are present. This imposes certain requirements on how the mesh is made and on how the finite elements are formulated. It should be noted that there are certain elements (such as Bernoulli beams or shell elements) in which not only the displacement but also its derivatives must be continuous, given that part of the displacements depend on the derivatives of other displacements (for example, $u=-zw'$ in Bernoulli beams: w' as well as w must be continuous).

The requisite of completeness means that the finite elements have to be capable of simulating both rigid displacements and constant strains. The formulation of the elements has to be made in such a way that to a displacement field consistent with a rigid translation or a rotation there are no corresponding strains.

The requisite that the various meshes form a regular succession means that all the nodes contained in the previous mesh must also be present in the successive ones, and that the same kinds of finite elements are used (albeit smaller) in accordance with the same rules for numerical integration.

When these conditions are respected, it is possible to prove that the convergence is monotonic, that is, that each successive, denser mesh more closely approaches the final solution than the preceding one did. However, although the monotonicity of the convergence is useful, it is not absolute and indispensable: elements that are incompatible can still lead to excellent results as long as the meshes used are sufficiently dense.

The rapidity of convergence is fostered when the elements are regular, that is, when the ratios between the dimensions of their various sides are close to 1 and their angles are close to 90° (or close to 60° for triangular elements). The isotropy of the shape functions (which should not privilege one direction over another) is also very important, but the analyst does not have access to this since he is using finite elements that have been formulated by someone else.

One check that it is always a good idea to make when analysing continua (plane stress or plane strain, plates, etc.) is a series of at least two or three analyses with regular meshes of increasing density in order to estimate how much the stress fields involved vary as the discretisation of the mesh changes. Such meshes can often be obtained by simply subdividing, for example, a one-dimensional element into two, or a plane element into four parts, and so forth.

Recalling that the solution corresponds to a minimum of total potential energy, and in a few easily understood steps, we can see that the approximate solution given by the FEM is always stiffer than the real solution, and thus that the displacements calculated are always smaller than the exact ones. For total potential energy we can write

$$\Pi=0.5\mathbf{u}^{\mathbf{T}}\mathbf{K}\mathbf{u}\ \mathbf{-}\ \mathbf{p}^{\mathbf{T}}\mathbf{u}.$$

Since the "real" displacement field is such that for $\mathbf{K}\underline{\mathbf{u}}\mathbf{=}\mathbf{p}$, we can write

$$\Pi_v=0.5\underline{\mathbf{u}}^{\mathbf{T}}\mathbf{p}\ \mathbf{-}\ \mathbf{p}^{\mathbf{T}}\underline{\mathbf{u}}\ \mathbf{=}\ -0.5\mathbf{p}^{\mathbf{T}}\underline{\mathbf{u}}.$$

Since Π_v is minimum, in approximate conditions we will have

$$\Pi_a = -0.5\mathbf{p}^{\mathbf{T}}\mathbf{u}_a > \Pi_v=-0.5\mathbf{p}^{\mathbf{T}}\underline{\mathbf{u}}$$

and thus

$$\mathbf{p}^{T}\mathbf{u}_a < \mathbf{p}^{T}\underline{\mathbf{u}}.$$

If the convergence is monotonic (and thus if all the corresponding hypotheses are true) then the approximate displacements must be smaller than the real ones, and thus the structure appears to be stiffer than it actually is.

To clarify this with an example, let's take the problem of plane stress shown in fig. 9-1.

Here we begin with a rather coarse mesh (6x6 elements, and only 252 degrees of freedom) and then we proceed to increase the density of the mesh, leading to increasing accuracy. As sample data we take one displacement and the smoothed value of the Von Mises stress on the middle point of the upper horizontal side (node 13). Note that the stresses and the displacements of the node when loaded are significantly influenced by the presence of what is, strictly speaking, a singularity, and thus they cannot be taken as reliable indicators of the validity of the analysis (the peak values continue to grow and while the peak area becomes increasingly smaller as the analysis proceeds).

An examination of the tables and the figures allows us to confirm that while the displacements converge rather well almost from the very beginning, in order for the stresses to converge a rather dense mesh is necessary, and the rate of convergence is in any case slower. This is related to the fact that the stresses depend on the derivatives of the interpolating functions, derivatives that are evidently predicted less accurately than the displacements themselves.

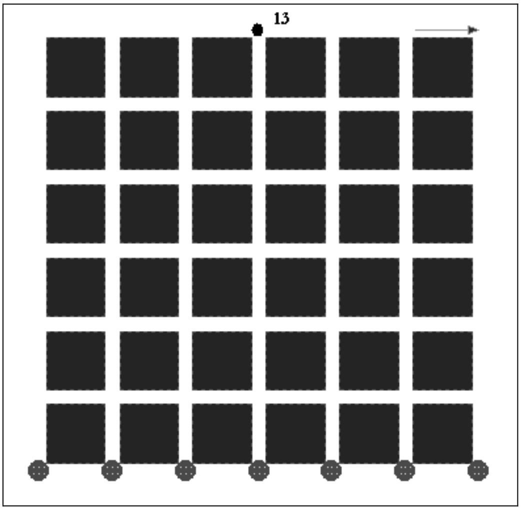


Figure 9-1: The starting mesh (6x6 grid).

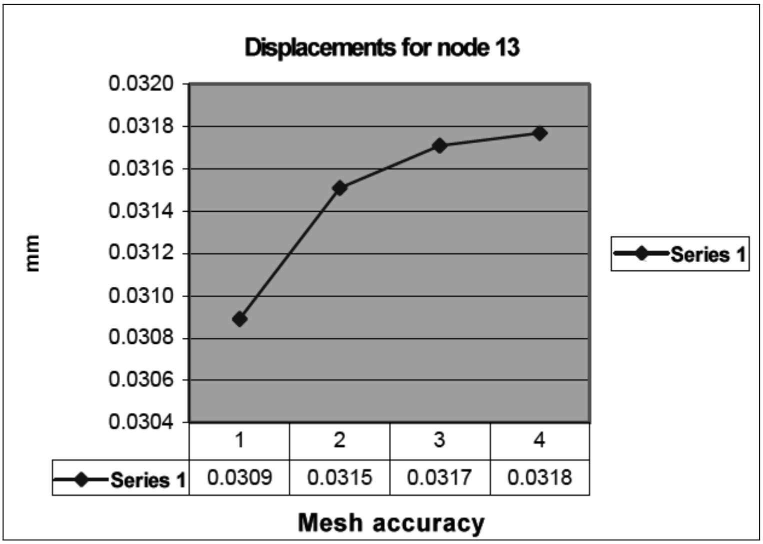


Figure 9-2: The tendency of a displacement towards convergence.

Today it is quite easy to produce this kind of data obtained from a series of successively denser meshes, because a model with 3,600 degrees of freedom requires very little time to "run" on low- to medium-capacity computers. Further, the subdivision of elements is accomplished by a simple command, and the modifications that have to be made to the model are often minimal: in this example the new nodes on the lower horizontal border have had to be constrained as they occurred.

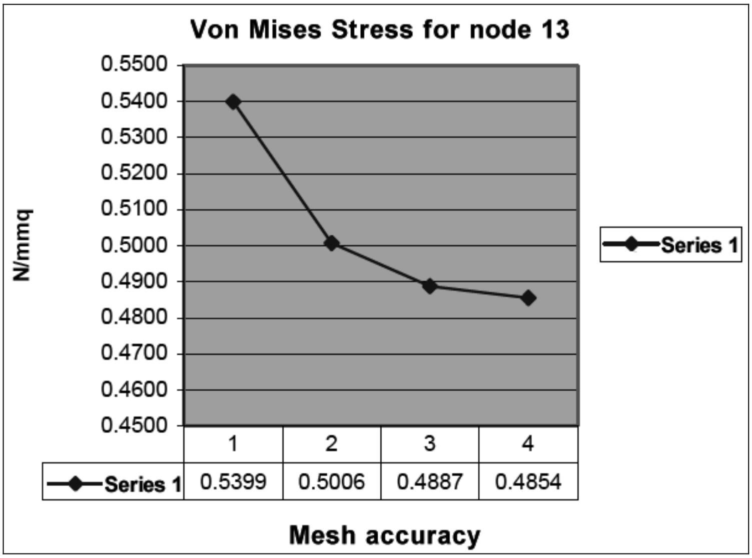


Figure 9-3: The tendency of a Von Mises stress towards convergence.

Mesh accuracy	Elements grid	No. DOF	% error displacement	% error stress
1	6x6 (1)	252	2.76	11.21
2	12x12 (2)	936	0.82	3.13
3	24x24 (3)	3600	0.18	0.67
4	48x48 (4)	14112	=	=

Table 9-1: Percentages of error assuming that a 48 x 48 mesh is "exact".

It is extremely useful to examine the contour lines as discretisation proceeds, because they provide a visual indicator of the progressively increasing precision of the stress field of the model (fig. 9-4).

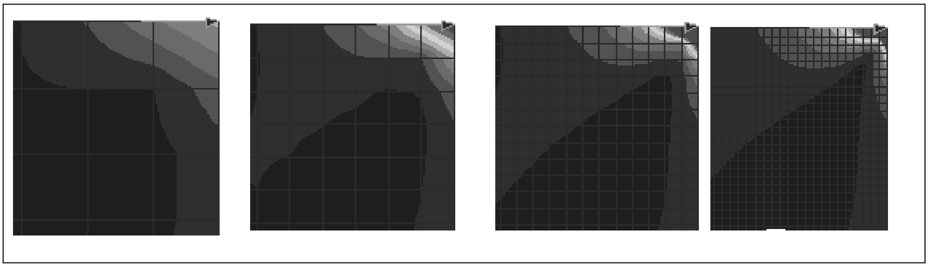


Figure 9-4: Increasing precise contour lines of stress (maximum value predetermined as 5 N/mmq): the upper right-hand quadrant of the model. Note that the dimensions of the elements are successively halved.

9.2.2 Measurement of error: Residuals and constraint reactions

Once the system of equations that gives the nodal displacements has been solved, it is a good idea to ask ourselves if any significant errors were made during the course of solving. Rounding errors, poorly conditioned matrices (generally related to quasi-mechanisms), and particular imbalances between stiffnesses all can make a solver give an answer that is less than satisfactory numerically, even though it has worked perfectly well on other occasions.

From a strictly numerical point of view the consequence of this is that the calculated solution **u*** does not rigorously satisfy the initial system of equations, and thus

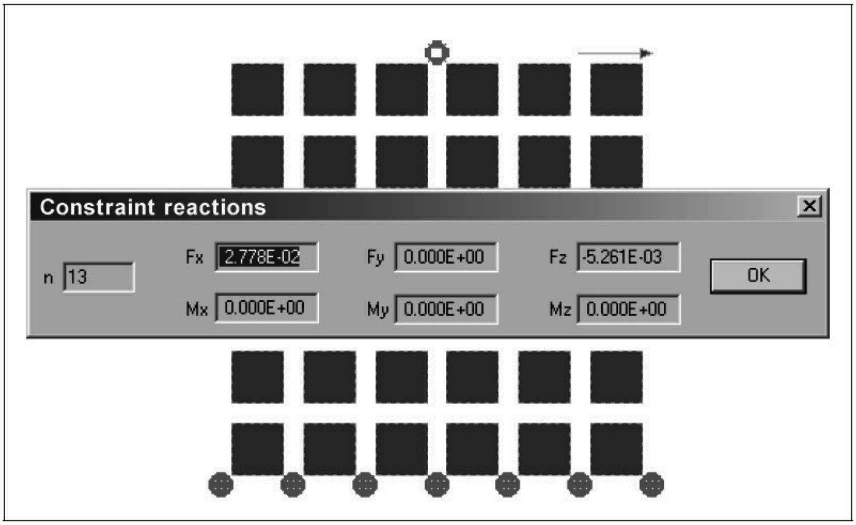
$$\mathbf{Ku^*} \neq \mathbf{p}$$

The difference between vector **Ku*** and the vector of equivalent nodal forces **p** is called the vector of residuals **r**, which, as we shall see, has an important physical interpretation. The vector of residuals is calculated by the definition

$$\mathbf{r=Ku^* - p}.$$

From a physical point of view the vector of residuals has the same dimensions as the vector of equivalent nodal forces, and it can effectively be interpreted as the set of unbalanced residual forces in the structure's various nodes, and according to their various degrees of freedom. It is interesting to observe that the calculation of the constraint reactions is performed with a very similar system in those nodes in which the displacements are known *a priori*, while the forces exerted by the constraints are unknown. Therefore, calculating the "constraint reactions" on free components leads, practically speaking, to calculating the residuals.

We thus see that by applying the formula for residuals in all the nodes of the structure it is possible to calculate with a fair degree of precision the size of the imbalanced forces, and it is thus also indirectly possible to evaluate the reliability of the numerical solution of the initial problem. The reliability of the problem's numerical solution should not be confused with the accuracy of the model. For example, fig. 9-5 shows the size of the residuals on node 13.



Constraint reactions

n 13

Fx 2.778E-02 Fy 0.000E+00 Fz -5.261E-03

Mx 0.000E+00 My 0.000E+00 Mz 0.000E+00

Figure 9-5: Size of the residuals on node 13 of the model in plane stress shown in fig. 9-1 (6x6 mesh, to be considered rather coarse). Note that the program, in this instance SARGON, evaluates these residuals with the same system used to evaluate the constraint reactions.

There are imbalanced forces of the order of, respectively, 3e-2N in direction *x* and 5e3-3N in direction *z*. The applied force is equal to 20,000N and thus the residuals can be considered extremely small, even if the solution is not at all refined as far as the model is concerned.

9.2.3 UNSMOOTHED STRESS MAPS

An extremely fruitful method for evaluating the degree of refinement of an analysis, at least in cases dealing with continua, is the analysis of the interelement jumps, both by direct means (with maps of the jumps), and by indirect means (unsmoothed maps). We have seen that the presence of significant interelement jumps where we instead expect the stresses to be continuous is a direct measure of how imprecise an analysis is. Although it is not possible to establish a direct relationship between the error and the interelement jumps, certainly the order of magnitude is an important indicator.

Let's look again at the example shown in fig. 9-1, and make a table of the maximum interelement jumps at node 13 for the various meshes used (in reference to the Von Mises stress of node 13).

Accuracy of the mesh	Maximum jump	Smoothed value	% Difference
1	0.1475	0.5399	27.3%
2	0.0821	0.5006	16.4%
3	0.0430	0.4887	8.8%
4	0.0219	0.4854	4.5%

Table 9-2: Examination of the interelement jumps.

An examination of table 9-2 reveals that in this case the jumps tend to be halved with each successive refinement of the mesh. The estimate of the error committed when we evaluate the percentage amount of the jump with respect to the (smoothed) value predicted by the corresponding mesh is also about halved. This kind of estimate is important because it can be made without requiring ulterior calculations or meshes that are further refined: it is an "internal" estimate of the reliability of the analysis. The final analysis has jumps equal to 4.5% of the predicted value: it is up to the analyst to determine whether or not this kind of approximation is sufficient, in part as a function of the actual final amount of the stresses with respect to the limits of reference.

For further evidence of this, consider what is shown in figs. 9-6 and 9-7. These represent the unsmoothed maps of Von Mises stresses (the jumps have not been eliminated) for, respectively, 6x6 and 48x48 meshes, that is, the two extremes on the mesh's refinement scale.

In both of these cases a maximum value of 5N/mm^2 was arbitrarily set in order to avoid contour lines that were too diverse being generated by the singularity at the point where the loads were applied. Hence, since the scale is fixed, the contour lines refer to identical values. An examination of the maps makes the coarseness of the 6x6 solution immediately clear, which, if not smoothed, makes it almost completely impossible to recognise how the contour lines go (which can instead be easily seen in the smoothed version shown in fig. 9-4). In contrast, in spite of the jaggedness that is still evident, the 48x48 solution shows a good level of regularity (in these examples a "basic" finite element with 4 nodes was used, which was not found to converge particularly quickly; much better results in terms of speed of convergence would have been obtained with isoparametric elements with 8 nodes, or with non-compatible elements QM6).

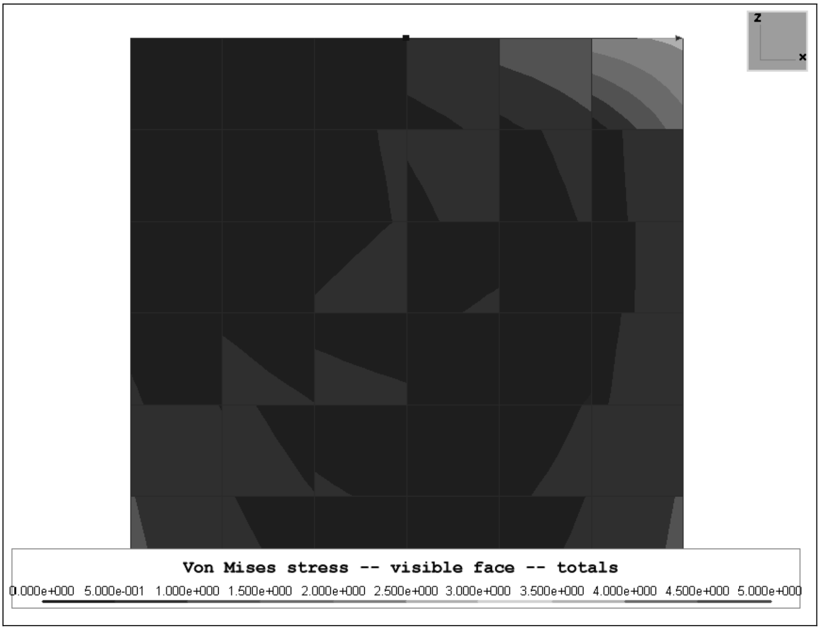


Figure 9-6: Unsmoothed map of the Van Mises stresses in a 6x6 mesh.

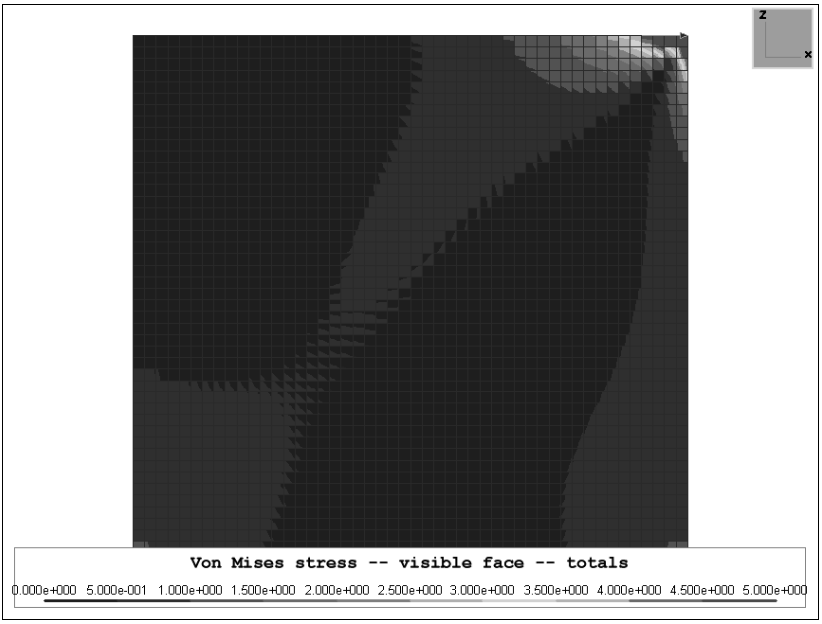


Figure 9-7: Unsmoothed map of the Van Mises stresses in a 48x48 mesh.

9.2.4 SIGNIFICANT DIGITS: TRUTH AND MYTH

In the search for precise solutions and reliable results we shouldn't lose sight of the way in which those solutions and results were obtained. Hypotheses that are simplified, sometimes to quite a great extent, have been applied at every single step leading from the physical reality to the eagerly desired colour map.

If we retrace the logical sequence that we followed, we will see that we have imagined the materials to be linearly elastic when in reality this is only vaguely true: think of reinforced concrete, masonry, and soils. We have imagined that we know the data regarding the materials being used – the modulus of elasticity, Poisson's ratio – while limiting ourselves to isotropic materials. Then we have imagined that the structure can be likened to parts that follow the various simplified theories; we simplified the constraints (often far from ideal); we have imagined that we can know the actions that take place there, including those that are random. Finally, on the basis of all that we have imagined, we have created a numerical model.

We cannot and should not forget that all of the approximations made have just as much influence on the results as the *rounding errors* do; in fact, they usually have a greater influence. It is thus illusory and academic to try to obtain stresses to the fifth or sixth decimal place in N/mm^2, displacements to the micron (μm), and so forth. The software programs give answers with such degrees of precision because they are furnished with all the available information, but the analyst should be thoroughly aware that this kind of precision has nothing to do whatsoever with *real* precision.

The fact that apparently precise instruments are available should not lead us to forget that all of the initial data are affects by errors and omissions, and thus that the results have to be taken as reliable only within certain reasonable limits, beyond which it is useless to go.

9.3 LOOKING FOR MODELLING ERRORS

9.3.1 OVERVIEW

In addition to numerical or modelling errors tied to insufficient discretisation of the mesh, a frequent cause of problems are *modelling errors*, that is, evident and not so evident oversights leading to erroneous results which are generally camouflaged by results that are more or less "correct". There is no set logical path to follow in order to flush out modelling errors, due as they are to oversight, conceptual errors and errors in entering the data; in short, this set of possible causes doesn't lend itself to a single rational course in searching for possible errors. Given that a finite elements software program cannot validate its initial data,[1] it is clear that the program will *in any case* arrive at *a* solution, and that it is only the skill of the analyst and his diligence in checking that prevent an erroneous solution from being adopted.

Errors must be looked for: not only when we have results that are obviously, glaringly wrong, but also when we have results that are apparently correct, since only a model that has been verified by all possible checks can be considered truly reliable. In practice today it is not uncommon that a model is accepted at face value, without any particular checks, and it is even more common that the person who uses the software program is not able to perform any checks because he or she is without the basic knowledge. Obviously this increases the burden of responsibility on the software house or producer, which must then try to keep its programs from being so deceptively seductive, and consider as its target group only those professionals who are sufficiently competent. It ought to be illegal to advertise programs as capable of covering the role of the engineer (in Italy a few years ago an advertisement with the slogan "Relax. Auto-engineer running" showed a complacent gentleman with his feet propped up on his desk admiring a wire frame model of a building on his computer monitor).

In the following sections we will provide a list of possible checks, or at least a minimum of checks, for the most frequent kinds of errors, in order to facilitate both the modelling process and the search for errors. The topic is so vast and complex that it really requires a volume in itself, which is in preparation.

9.3.2 FORMULATION

One typical source of errors in results is an error in formulation. This occurs when a more general problem is related back to a theoretical framework in which important parts are lost. Obviously this *always* happens, strictly speaking: there are no Kirchhoff or Mindlin plates in nature; there are only generic solids. The fact is that only in certain circumstances is it possible to relate the dominant part of the stress of a given problem back to what is obtained by means of a calculation performed according to one of the simplifying theories. In the great majority of cases, this cannot be done.

Let's review some rules that might be useful:

- in Kirchhoff or Mindlin plates there are no normal stresses acting perpendicular to the median plane; if such forces are fundamental, then solid modelling is necessary;
- the Kirchhoff plate does not take into consideration the strain energy associated with shear *but does not imply that the shears are null*;

[1] Nor do current attempts to check the data, which try to trace the part of the initial input data back to predetermined categories, appear to be sufficient.

these shear forces have to be determined if the objective is to study the plate;

- the Euler-Bernoulli beam and the Timoshenko beam exclude all of the normal stresses acting in a direction perpendicular to the axis of the element; if the beam is excessively chunky (say L/H < 8), then the Timoshenko theory may not be adequate either;
- the use of plane stress in structures whose thickness is significant must be adopted with care;
- models using homogeneous and isotropic materials for heterogeneous, anisotropic or orthotropic structures might give rise to significant errors;
- the use of membrane schemes is useful when we don't want to take bending stiffnesses into consideration, but we have to be sure that bending stresses induced in the "membrane" will not provoke failure;

It is worthwhile to emphasise once again that no matter how much the discretisation of the mesh in the model is increased, if the formulation is not suitable then the result will always be incomplete or incorrect.

9.3.3 The mesh

The most typical and frequent causes of errors in a model are related to problems with the mesh. It is always good practice to check the mesh carefully in order to avoid problems of disconnection or violations of congruity that might compromise the validity of the entire analysis. The use of shrunken representations can facilitate checking the mesh to a considerable degree, since this makes it easier to identify elements that are facing or unconnected.

As we make the mesh we should also be wary of operations that lead to the creation of many new nodes near or on top of existing ones, unless the software program is equipped with dependable instruments for checking whether or not there are double nodes or double elements at the moment the operation is performed and not after the fact.

The instruments for automatic intersections among elements, although certainly quite convenient in specific contexts, should not be used indiscriminately, because they can generate elements which are collapsed or distorted, and which are not always correctly evidenced by the programs that make the mesh. These commands are more properly used for clearly defined parts of the model, for which we already have a final mesh scheme in mind.

Errors during the meshing phase can be made evident during the post-processing phase by studying the deformed configurations or the stress fields (both smoothed and unsmoothed: unexpected discontinuities in the smoothed stress fields indicate that there are disconnections, as do the detachments of congruent parts). Many a time these errors are manifested even earlier during the solving phase, since they are responsible for mechanisms or pseudo-mechanisms that cause the solving to block. A typical case is that of nodes that are not referenced, that is, nodes in the structure that have not been attached to an element.

Also typical is the quite frequent case of structures which contain elements that are facing or unconnected, so that entire parts of the structure turn out to be completely free to move, since there is no structural continuity in the model. An example of such a case (which has actually occurred more than once and is thus of some significance) is shown in fig. 9-8, where the vertical elements are only facing the longitudinal members rather than being connected: the longitudinal members thus turn out to be made of only two finite elements. This is possible to see, for example, when the element numbering is shown in the middle of each element: a single element per longitudinal member is not enough.

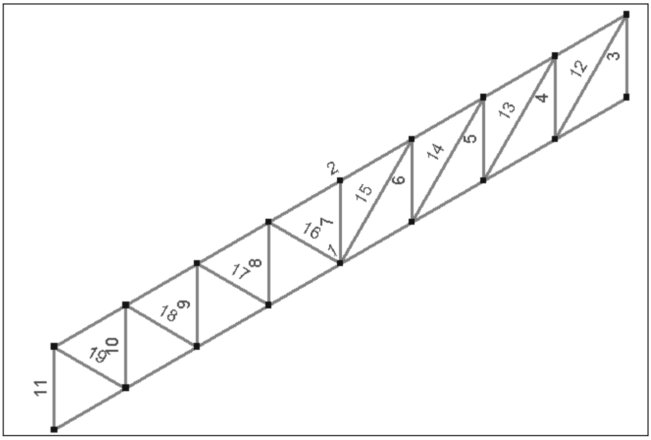


Figure 9-8: Erroneous mesh of a truss (element numbering).

Practically speaking, meshes such as this one are not meshes at all, but rather drawings. This is in fact one of the classic reasons why the beginner is unable to make proper models: because he or she hasn't understood that *the mesh is not a drawing.*

For use in specific circumstances, and with the aim of producing software programs that are simple to use, it is possible to conceive tools for meshing that systematically intersect all the one-dimensional elements: I myself successfully performed an experiment of this kind with the software C.E.S.CO, which in effect did drastically reduce the kind of problems that are linked to disconnections discerned by the user. The other side of the coin was that the programs like this do not make it possible to create certain kinds of modelling which, for instance, require elements that pass near each other but are not connected (for example, St. Andrew's cross-bracing with independent diagonals).

9.3.4 DOUBLE ELEMENTS AND DOUBLE NODES

Double elements and double nodes warrant a separate discussion. These rather quite common errors are rather deceptive because they are not glaringly evident.

Double nodes can lead to disconnections that are not immediately discernible and thus, as a consequence, to mechanisms. In general the meshing programs are equipped with specific tools for checking that are intended to prevent the generation of double nodes, which tend to be generated almost at every turn (think of the simple duplication of a block placed after another one: if there were no checks, then double nodes would be generated and the two blocks would not be attached to each other). Sometimes the software program is equipped with an appropriate command that makes it possible to look for double nodes in the mesh; having established a minimum distance under which two nodes are considered to be coincident, such a tool investigates to see if there are double nodes and if so "repairs" the mesh (cf. Sargon).

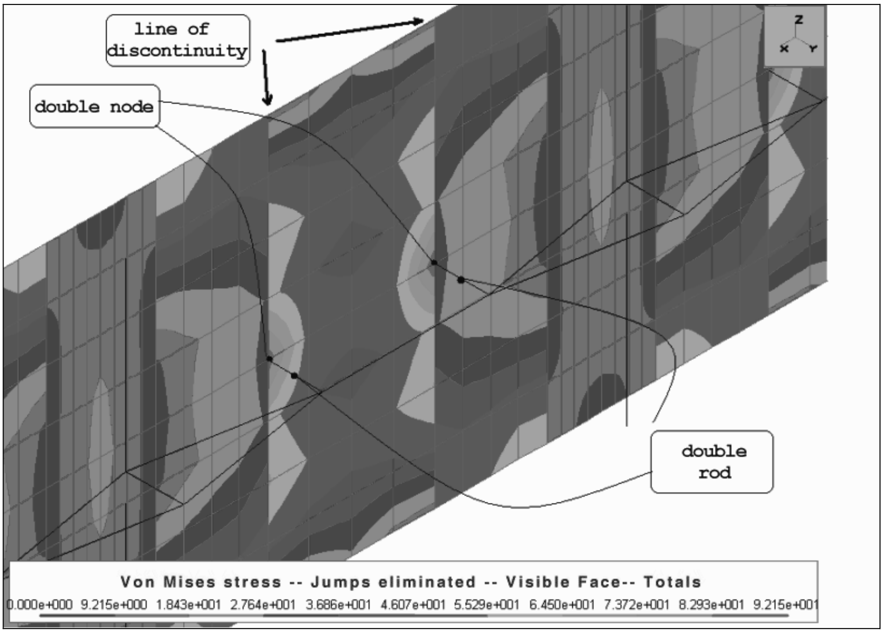

line of discontinuity
double node
double rod
Z
X Y
Von Mises stress -- Jumps eliminated -- Visible Face-- Totals
0.000e+000 9.215e+000 1.843e+001 2.764e+001 3.686e+001 4.607e+001 5.529e+001 6.450e+001 7.372e+001 8.293e+001 9.215e+001

Figure 9-9: Study of facing glass sheets: construction discontinuities.

Sometimes, however, the analyst wants there to be double nodes, as when he or she has to model construction joints or actual gaps, and thus needs nodes that occupy the same position but are kinematically independent. This is shown, for example, in fig. 9-9, where it was necessary to study glass sheets that are facing but independent, held in place by single hangers at specific points. This situation was modelled by means of two independent rod elements which had a common node at one end, while at the other end they were attached at two nodes – double – of the glass: one that appertained to the glass sheet on the right, the other to that on the left.

9.3.5 CROSS SECTION ORIENTATION AND END RELEASES

In cases of structures where there are beam elements, it is necessary to check the orientation of the elements carefully, because an incorrect orientation is certain to lead to incorrect results. The representation of the local axes or the solid representation of the elements is helpful in checking that all of the elements of the model are correctly oriented. Except for this particular function, this kind of "solid" representation is of no use to the analyst, who will find it to be contradictory and misleading in beam and truss models which are “wire frame” by definition. Those who rely on such images for their scenographic effects will have a different point of view; in these cases solid representations are desirable, but mainly for giving an idea of what the structure will look like once it is complete. It is not by chance that such images are pervasive, for example, in advertising.

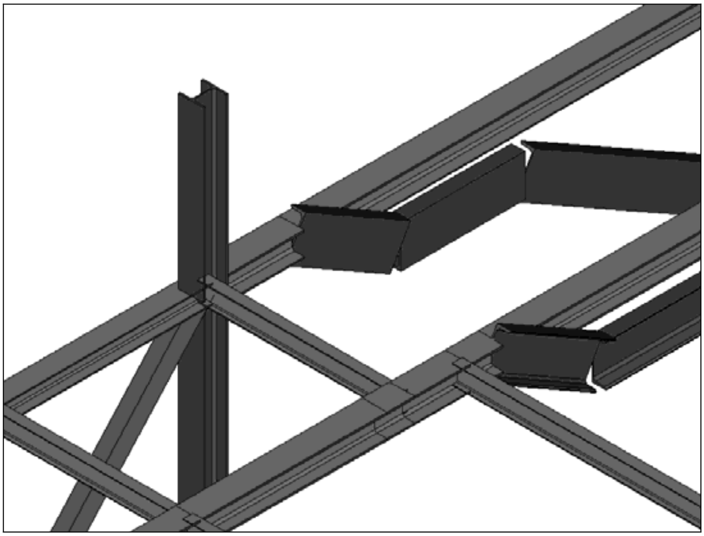


Figure 9-10: “Solid” representation of a beam and truss model.

It should be kept in mind that the orientation is determined by the choice of the local axes of the element with respect to the global axes. An analogous consideration is made when the end releases are applied to the ends of beam elements, releases that have to be set up on the basis of the local axes of the element and not with reference to the global axes of the structure.

9.3.6 SYMMETRIES AND ANTISYMMETRIES

If there are symmetries or antisymmetries that are not taken advantage of during the modelling phase, these must be used in order to validate the structural solution. The presence of unjustified asymmetries in the solution will naturally lead us to consider the model unreliable, and look for the causes of the asymmetry.

It should be recalled that in order to call an answer symmetrical (or antisymmetrical) there has to be symmetry in:

1. the formulation of the elements;
2. the geometry;
3. the mesh;
4. the loads;
5. the constraints and end releases;
6. the applied actions.

If even one of the factors listed is not present, or has been violated by mistake, then the answer will not be symmetrical. Further, even when we have a "symmetrical" model, it is possible that in reality there are small asymmetries related to calculation. In certain cases the model's collocation in space (for example, the fact that a symmetry plane does not pass through the origin) can give rise to (small) asymmetries in calculation.

9.3.7 GLOBAL AND PARTIAL RESULTANTS

Before performing any analysis it is always a good idea to check the distribution and the magnitude of the applied loads carefully. In order to do this, if the software program is equipped for it, it is very convenient to be able to determine the resultant of the loads before solving, in terms of both the global resultant and the partial resultant on parts of the structure. If, for example, a load equivalent to 4kN/m^2 has been applied to a floor slab modelled non-uniformly with primary and secondary elements, then it was probably necessary to make a series of approximations and roundings of the magnitude of the loads, given that the distances between two axes will turn out not to be constant and expressed in "clean" numbers.

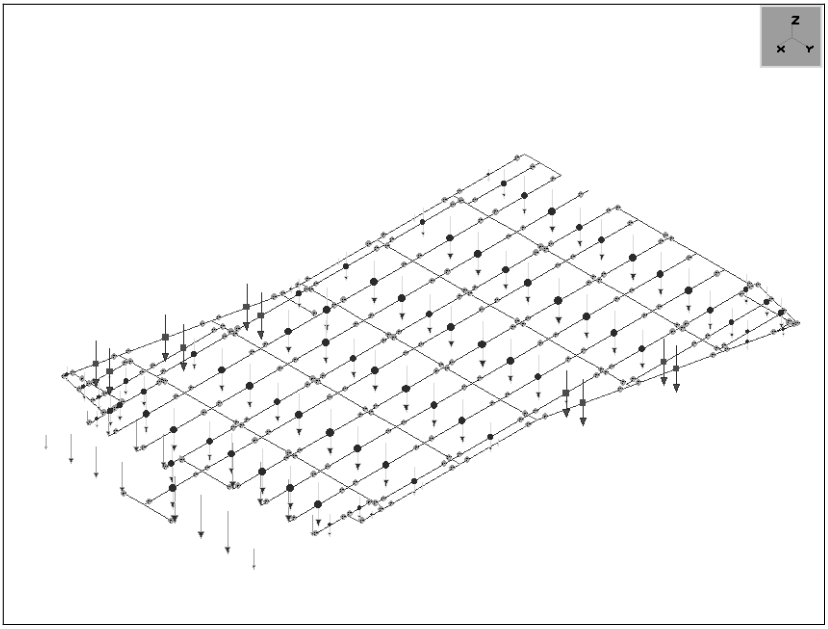


Figure 9-11: Loads applied on beams of a floor slab: the resultant?

In the end, the situation might be that shown in fig. 9-11: a series of loads that are all different. Checking that the global resultant of the loads *on that floor* of the building are as expected (surface per unit load) provides strong evidence of the fact that the loads were correctly modelled, and is independent of the check that follows of the constraint reactions, which is performed after solving.

9.3.8 Deformed configurations

The analysis of deformed configurations is one of the key tools for recognising errors in the model. The analyst has to try to establish if the expected congruity has been everywhere respected, in order to identify disconnections or errors in the mesh that could compromise the analysis.

The displacements of the main structural elements also provide important information about the reliability of a first order approach, since if there are significant added effects on the stresses related to the configuration (for example, the P-Δ effect), this must be taken into account.

The precision obtained for the displacement field is usually much greater than that, given equal meshes, obtained for the stresses, which generally depend on the derivatives of the displacements: thus we mustn't be misled by the convergence of the displacements if what we are (also) interested in are the stresses.

9.3.9 THE FUNDAMENTAL ROLE OF ENGINEERING ESTIMATES

The most important check of all is the engineering check. Here the analyst has to ascertain if the calculated structural response meets all of the requisite conditions for being considered reliable. It is impossible to provide a complete list of the factors to be checked, but it is in any case worthwhile to provide a partial one.

Factors to be checked include:

- The relationship of the applied loads and those expected from the design;
- The correspondence between loads and constraint reactions;
- The correspondence between the magnitude of the internal actions and the applied loads (for example, in the axial actions of the columns, in the bending moments of the beams, in the normal and shear stresses);
- The magnitude of the displacements, which have to be likely;
- The shape of the displacements and of the disconnections;
- The stress field, by means of schemes (including those that are extremely simplified and independent of the calculation model) aimed at verifying the results obtained (the role of the fundamental analogies, in a critical sense as well, is essential);
- Respect for the symmetries or regularities predicted by the theory, when pertinent;
- Respect for the morphology of the theoretical stress bending moments at the nodes, torque, the shape of the shear and moment diagrams with respect to the applied distributed loads, that is, with respect to the indefinite equilibrium equations;
- The residuals;
- Interelement jumps;
- Congruities between the deformed configuration of the model and the structurally kinematic constraints of the design;
- Consistency between the schemes adopted in the model and those adopted in the design, with particular attention to constraints and end releases;
- The correctness of the dimensions of the resisting elements, calculated spans of members, thicknesses, moments of inertia and areas, shear factors;

- The correctness of the values adopted for the material constants in terms of Young's modulus, Poisson's ratio, limits for yield and ultimate stresses;
- Correctness and thoroughness of the calculation combinations;
- Consistency between stresses and solicitations, with checks performed at significant sample points.

There is no such thing as a self-verifying model. It is up to the analyst to verify it, by submitting the model to all possible checks that appertain to the case at hand. *Numbers and coloured maps without such checks having been performed are worse than useless.*

For some years now we have been aware of the risks connected to an over-confident and inappropriate use of computer software, and many have denounced the dangers inherent in using computer programs without performing the necessary checks. A professional who is an expert *in structures* is irreplaceable in the role of verifying every possible numerical analysis produced by the FEM or any other kind of method.

It is worthwhile to quote a particularly strict point of view with regard to this point, that of Leroy Z. Emkin, who is a harsh critic of the widespread use of computers for structural analysis, even though he is co-director of one of the largest centres for research and development in computer software for structural analysis, GTSTRUDL. According to Emkin, it is urgently necessary:

1. To recognize the extreme dangers of computers.
2. To understand the basic principles of mathematics, science, mechanics, material behavior, system behavior, modeling techniques, analysis methods, design procedures and codes, error assessment, risk analysis, codes of ethics, and ethical engineering practice.
3. To understand the absolute requirement for engineers to be able to engineer without computers.
4. To always be skeptical of computers, to never ever use computer results without extensive validation, and to always assume computer displayed results are wrong until proven correct by the engineer.
5. To "know" the answer and merely use the computer to fine tune the solution.
6. To deglorify the computer, and to glorify knowledge and experience and the need to be thoroughly familiar with all the details of engineering system behavior, modeling, theory, and practice.
7. To avoid taking educational courses from engineering faculties that only provide opportunities to learn through computer use rather than by intensive instruction in the principles of engineering by

highly knowledgeable engineering educators who have extensive real world practice experience.

8. To avoid working for employers whose only available opportunities to learn are through computer use rather than through intensive training by experienced and knowledgeable engineers.
9. To recognize that less experienced engineers must develop strong engineering skills without the aid of computers before using computers as powerful engineering modeling, analysis, and design tools.
10. To recognize that only the most experienced and knowledgeable engineers are qualified to use computers as a tool for engineering modeling, analysis, and design.
11. To recognize that *only* engineers engineer, and that computers *do not* [Emkin 1998].

We have quoted these recommendations in their entirety because they are characteristic of an important mentality regarding the current problem of what constitutes the correct use of a computer. My own point of view is largely in agreement with that of Emkin, though I differ on some points.

There are cases in which the use of the computer is an irreplaceable tool for understanding because it makes it possible to determine effects that would otherwise be impossible to determine: unfortunately the simplified schemes are not always able to explain behaviours that are intrinsically complex. It is not always possible to "know" the solution beforehand. If used in an appropriate way, a software program can prove to be the means to know things that cannot be learned in any other way.

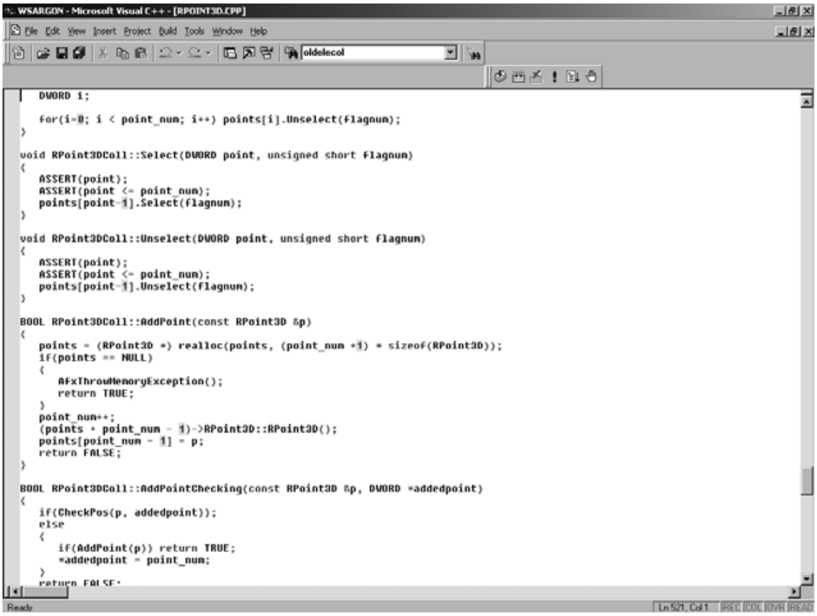
Although formal training is certainly indispensable, and time spent in performing calculations manually is irreplaceable, I don't believe it is necessary to forbid the use of the computer until complete knowledge of its functions is achieved. It would perhaps be better to flank "hand" calculations (or those made with other systems) with those performed by software in order to promote familiarity with the correct and supervised use of the computer. Thus I cannot share in the proposal to limit the use of computer programs to only those who are thoroughly expert.

It should also be said that time spent in performing calculations by hand needn't be overly romanticised, because even manual calculations were and are still plagued by errors due to oversight, fatigue, and numerical confusion.

However, having said this, the moral of the story is that every method of calculation, without exceptions, must always be carefully overseen and verified by competent professionals.

9.4 SOFTWARE ERRORS

Software errors are perhaps the most deceptive of all errors that the analyst has to deal with. Even though the software programs have been tested and certified, let's make it clear at the very beginning that software programs, all of them, are sure to contain errors. Those producers of software programs who claim that their products are free of bugs are merely talking spin because it is *impossible to prove definitively and with absolute certainty that any computer program is error-free.* Experience has proven that bugs are found in even the largest and most expensive programs, created with sophisticated systems of quality controls. Naturally, this does not mean that the testing and certification procedures are useless: to the contrary, even if they should be thought of in a probabilistic sense, as instruments for confining the possibility of errors within decreasing limits of probability, they are indispensable.



```
   DWORD i;

   for(i=0; i < point_num; i++) points[i].Unselect(flagnum);
}

void RPoint3DColl::Select(DWORD point, unsigned short flagnum)
{
   ASSERT(point);
   ASSERT(point <= point_num);
   points[point-1].Select(flagnum);
}

void RPoint3DColl::Unselect(DWORD point, unsigned short flagnum)
{
   ASSERT(point);
   ASSERT(point <= point_num);
   points[point-1].Unselect(flagnum);
}

BOOL RPoint3DColl::AddPoint(const RPoint3D &p)
{
   points = (RPoint3D *) realloc(points, (point_num +1) * sizeof(RPoint3D));
   if(points == NULL)
   {
      AfxThrowMemoryException();
      return TRUE;
   }
   point_num++;
   (points + point_num - 1)->RPoint3D::RPoint3D();
   points[point_num - 1] = p;
   return FALSE;
}

BOOL RPoint3DColl::AddPointChecking(const RPoint3D &p, DWORD *addedpoint)
{
   if(CheckPos(p, addedpoint));
   else
   {
      if(AddPoint(p)) return TRUE;
      *addedpoint = point_num;
   }
   return FALSE;
```

Figure 9-12: C ++ source codes

The impossibility of proving that a program is error-free is due to the fact that every program contains thousands of "ifs" and thus there is such an enormously large number of possible execution paths that it is literally impossible to test them all. Further, since at present there are no automatic tools for performing these checks, software tests are still largely performed manually (which is time-consuming and in any case involves another possibility of error).

Performing sets of standard tests before each new release helps limit the probability of errors but certainly does not protect us from them altogether.

The practical impossibility of creating programs that are free of defects is intimately related to the way software itself is structured, composed as it is of tens and tens of thousands of interdependent instructions written by hand (albeit by the hand of programmers).

To this discomforting conclusion is added the authoritative voice of *The Economist*, which in June 2003 reported on a study that calculated the damage caused annually in the United States by errors in software to amount to sixty billion dollars, and concluded that "People who write it are human first and programmers only second" [*Economist* 2003]. The fact that a magazine such as *The Economist* dedicates space to the economic consequences of bugs and the systems aimed at controlling them says much about the state of the art: with varying probability and frequency, bugs exist in all software programs large and small, and software users have to be aware of this in order to be able to manoeuvre when they encounter one.

In effect, much work has been done over the last twenty years to make available programming techniques that are increasingly secure. At the forefront of computer languages, the object-oriented approach has definitely supplanted the procedural approach (and C++ has in fact sounded the death knell for FORTRAN in the most important and widespread applications). Once programs were composed of vast numbers of "functions" or "routines" organised by programmers in ways that were substantially arbitrary, often making use of ad hoc assumptions that were valid in limited circumstances and then perhaps totally forgotten, but this is no longer the case. Today's programs are created with a series of "objects" (which can be either data or instructions), with very precise tasks, procedures and information, organised according to a logic that protects them from undesired manipulation. The increasing logical clarity that has grown out of this has improved data security enormously and made it possible to keep programs up-to-date.

As far as the user is concerned, the availability and widespread use of graphic interfaces has made it much easier than before to check the models as well as the results produced by the software, so that violations or software errors are often made glaringly evident.

However, there are still cases in which errors in a program are not obvious and easily verified. This occurs when the results are only quantitatively or locally damaged by the error, and it is thus particularly difficult to perceive them, even for analysts with the most experience in structures and calculation. If the results are wrong quantitatively, the engineering checks are often able to recognise that an error has occurred, although there are cases where the quantitative difference is not large enough to lead to the unquestionable conclusion that the error lies in the software program in use.

If the model does not represent a structure already described or studied in the literature, the only way that we can determine the reliability of the calculation indirectly is to compare the results obtained by one or more different programs (*analogue proof*). "While it is in fact possible that the results obtained by program A are wrong, it is highly improbable that program B, which gives results that are practically identical, also errs *in the same way*. It can thus be seen that only by an analogy between two or more potentially erroneous systems that we can reach the (probabilistic) certainty that there is no error" [Rugarli 2003].

This technique has been adopted in chapter 10 of the Italian code NTC2008 as the main tool for assessing the effectiveness of model computation.

It is thanks to precisely this technique that I personally have been able to prove that an application used for testing my own software produced different results because *it was itself erroneous.*[2]

This consideration might lead to the conclusion that programs should be compatible with each other in order to favour systematic reciprocal testing aimed at providing a probabilistic proof of the correctness of the results achieved. With regard to this, it is my opinion that we should encourage compatibility between programs that serve the same purposes rather than the creation of computational barriers intended to limit commercial competition but thwart such comparisons.

[2] This was an analysis of frequency response. The application used as a benchmark (known throughout the world) had been subjected to a process of quality control that was intended to prove its reliability. The correction arrived many months later, with the obligation that I sign a paper that imposed its use in non-professional situations only because the correction had not been subjected to quality controls (!).

10 STRUCTURAL PROBLEMS

10.1 Plane stress

Problems of *plane stress* are used to study thin membranes subject to forces that are equal throughout the thickness and lie in the plane of the plate. In these conditions the stresses do not vary along the thickness of the plate, and are reduced to two normal stresses and one shear stress.

The plane stress model is also often used to determine the situation of walls or shear walls, with or without openings, subject to vertical and horizontal forces, even if the model isn't completely correct (in part because of the difficulty of determining the elastic characteristics of the masonry or concrete, in part because the thickness of a wall isn't all that negligible with respect to its dimensions).

It is also possible that a part of a more complex structure is modelled with membrane elements (that is, in plane stress) which, even without imposing plane stress on the structure as a whole, makes it possible to model efficiently elements such as the shear walls just mentioned, or a bracing core, where modest effects of bending can be expected (in the case of a bracing core, the bending of the overall core mustn't be confused with the local bending of the walls that make it up; only the local bending is neglected when the core is modelled with membrane elements).

The plane stress model is also useful in the study of the effect of the distribution of stress in a plate in tension or compression. In the case of the web of a beam, this might contain holes, and the plane stress model can make the corresponding concentration of stresses evident.

Generally speaking, if there are holes or notches, or a marked variation in geometry, these can lead to concentrations of stresses which, in order to be correctly grasped, require an adequate local increase in density of the mesh and a careful evaluation of the interelement jumps.

In plane stress problems there are usually only two translations in the plane in question. Some types of finite elements also use the rotation of the nodes about an axis that is perpendicular to the plane of the structure (cfr. § 11.5.3).

10.2 Plane strain

Problems of plane strain are used in the study of structures that are elongated in a direction normal to the plane of the model, and subject to forces lying in the plane of the model and equal at all sections of the structure. Although in principle the structure should be elongated indefinitely, in practice even a structure that shows a marked elongation in one direction (for example, a retaining wall or a dam) can be studied in this way. Typical problems of *plane*

Structural analysis with finite elements
978-0-7277-4093-9

strain are those connected to underground sewer conduits, pressurised tubes, and tunnels. In examining the situation of underground structures below roadways it must be borne in mind that only some of the load situations can be likened to those of plane strain, since the presence of loads that are not repeated equally on all sections of the structure automatically renders the basic presumption invalid.

The study of a structure in plane strain cannot be made by mixing plane strain elements with other, different types of elements (for example, plate, beam, truss).

In a plane strain problem the only unknowns are the two translations in the plane in question (with the exception of elements with drilling DOF; cfr. § 11.5.3).

10.3 AXISYMMETRIC PROBLEMS

Axisymmetric problems are found when the structure can be likened to a solid of revolution subjected to forces that show axial symmetry, that is, those that are essentially independent of the angle θ[1]. Examples of typical axisymmetric structures are pressurised tanks (including the pressure vessels in power stations) or tanks full of liquids or gas. Due to the increase of the available degrees of freedom, axisymmetric modelling is less frequently used today than it was twenty years ago, although it should be noted that, given an equal number of degrees of freedom, the axisymmetric schemetisation can easily lead to much more precise results. In general, the bending part of the response of the tank or shell can be grasped with axisymmetric bending elements or through modelling with an adequate number of elements in the thickness (certainly more than one or two).

Problems of point impact on tank structures are not standard axisymmetric problems, because there is no axial symmetry in the loads.

As for structures in plane strain, structures studied using axial symmetry cannot contain any elements other than those of the axisymmetric formulation.

10.4 PLATE PROBLEMS

Plate problems occur quite frequently. The plate models behaviour in bending of a wide variety of two-dimensional continuums: slabs, walls, flanges or webs of beams subject to loads normal to their planes, foundation beds, cylinders or pressure tanks studied when axial symmetry is not taken advantage of, plane roofs subject to concentrated or distributed loads, etc.

As a rule the model used most often is the thin plate, rather than the thick plate. This is due to the fact that the strain energy related to shear can often be neglected, since the plates are of relatively modest thicknesses. As for beams, the

[1] It is also possible to take advantage of the axial symmetry of the structure even when the loads are not axisymmetric. These applications are, however, less frequent.

relationship between the deformed configuration due to shear and that due purely to bending is on the order of (E/G)*(t/L)^2, and thus for reasonable thicknesses it becomes quite small. As a rule of thumb, we can say that with a ratio L/t = 10 the plate can be considered thick; with a ratio of L/t=100 it can be considered thin. In the intermediate zone there is a brief interval in which the effects of shear tend to increase as L/t decreases.

The sandwich plate is a different story. This is obtained by superimposing various layers, especially when the material of the central layer has a high level of deformability under shear. The use of thin plate theory can sometimes lead to neglecting the shears normal to the plane of the plate (generally missing in the formulation of the elements); this is not correct, since these shears can assume significant values, very much like what occurs in the case of thin beams. If the finite element does not make it possible to determine the value of these shears, then these must be estimated at least approximately by taking equilibrium into account.

However, in general the predominant stress factor is tied to the presence of plate moments, and thus the study concentrates on the effects that these components of stress provoke in the various points of the structure.

Normally plate elements incorporate membranal behaviour as well, and so once a plate element has been placed, it is not necessary to add a corresponding membrane element.

Plate problems generally require a rather pronounced discretisation, since the elements most commonly used by the low- to medium-range solvers do not converge very rapidly. "Plate" problems can easily appear in parts of larger structures, such as buildings. It thus happens quite often that we find plate elements in the most general and complex models.

The bending component of plate elements is governed by the translation of the nodes in a direction normal to the plane of the plate, and by the rotations of the nodes about two orthogonal axes lying on the plane of the plate. The membranal component is governed by the two remaining translations and, in the case of elements with drilling DOF, by rotation about an axis normal to the plane of the plate. If the membranal part does not use drilling DOF, this degree of freedom remains inactive.

10.5 Shells

Shells are two-dimensional continua that are curved in at least one direction. In contrast to what takes place for plates, in shells the membrane problem is generally not disconnected from that of bending, and thus elements that have bending stiffness are necessarily used for modelling shells.

In order to model the geometry of shells adequately, curved elements are needed. These generally have a number of nodes equal to at least 6 for triangular elements and 8 for quadrilateral elements.

Although the modelling of shells with plate elements is not precise, because it introduces cusps that do not appear in the original continuum and because it uses elements that are planar rather than curved, the practice of modelling shells with plane elements is nevertheless widespread. This is due in part to the fact that the management of curves usually requires modelling to be done with complex primitives, such as cubic splines, which are more difficult to use. In these case it is clearly of the greatest importance to use a very dense discretisation, so that the parasite effects related to the cusps just mentioned are reduced to a minimum.

An example that sheds much light on how to calculate a shell with planar plate shell elements with a limited number of nodes can be found in chapter 12, showing a tank without reinforcing rings subject to hydrostatic loads.

10.6 Solid problems

Recourse to solid modelling is limited to those cases in which none of the available simplified formulations is applicable to the problem at hand. The generation of solid models is intrinsically more complex, both because it is more difficult to create the mesh, and because in general a very large number of degrees of freedom is necessary. The examination of the results is also more difficult, since the analyst cannot limit himself to a superficial examination but has to study what takes place inside the solid: it is therefore necessary to “slice” the solid with an adequate number of planes in order to understand thoroughly what takes place in terms of stresses and strains.

Typical solid problems are those structural problems relative to very thick mechanical pieces loaded generically. Examples of these are cogwheels, gears, transmissions, engine blocks, etc.

Solid modelling requires software programs of the medium to high range, since, with the exception of the banal case in which the mesh can be obtained by means of the extrusion of a plane mesh, the correct modelling of the geometry requires that curved elements be available, and thus modelling software must offer complex primitives such as, for example, splines. In any case, solid modelling is usually performed only by those who have the most expertise, in CAD if in nothing else.

In solid problems the unknowns are the three translations about each node, while the rotations are usually inactive.

10.7 RETICULAR STRUCTURES

Plane and space grid structures are very widespread and thus this kind of modelling – at least in the context of civil engineering– is very frequently encountered. The basic presumption behind this kind of model is that the bars that make up the structure are like rods, that is, they show the presence of only axial actions. In theory this would call for constraints like hinges at the ends. In reality, the presence of constraints other than hinges leads to the creation of a stress in addition to the nominally axial stress; in some circumstances this additional stress can be quite significant. In practice, it is often sufficient that the bending stiffness of the elements be very low, or that the joint not be able to transfer bending and twisting moments adequately, without resulting in very large displacements. Since the structure usually finds an equilibrium configuration in correspondence to very small displacements, the parasite stresses that are created remain quite low.

When we are dealing with structures made of elastic-plastic materials (steel), then the theorem of plastic limit analysis applies; this states that the true limit multiplier of the applied loads is always greater than that corresponding to a fictitious stress distribution in conformance with the plastic crtiterion adopted. Therefore modifying the constraints leads to safe solutions if all is kept consistent with the change adopted.

In order for a structure to be modelled with only rod elements (or *truss* elements), there must not be any loads perpendicular to the axis line of the elements applied at points other than the nodes of the structure.

The orientation of the rod elements has no influence whatsoever on the results of the analysis, but it can be useful in any case to orient these elements suitably for design purposes or where it might be desirable to transform the rod elements into beam elements (in the software programs that allow this).

The axis line of the rod elements corresponds to the location of the centroid of the sections of the structural element. If there are eccentricities with respect to this location, the element is no longer strictly speaking a rod but must be modelled with a beam.

In trusses modelled with rod elements (and only with rod elements), all the rotations of the nodes must be eliminated, either by explicit constraints, or by a suitable formulation in the software programs that allow for this (cfr. chapter 7). If the structure is flat, then translations in a direction normal to the plane of the structure have to be eliminated as well, while of course no forces normal to the plane of the structure are to be applied.

Trusses can be used to schematise structures such as pipe racks, towers and masts, beams with parallel chords variously connected to side struts, cranes, and so forth.

10.8 FRAMES

In plane frames as well as space frames, the elements are modelled with beams, usually with reference to the Euler-Bernoulli theory for slender beams. The stresses are thus general enough to include the two bending moments about the main axes, the two shears, the torsion, and the axial force.

Although modelling with beam elements is quite intuitive, it is necessary to be aware of some possible pitfalls.

A first problem is related to the significance of the energy of shear deformation, which is anything but negligible when the beam tends to be short but thick. In this case there is a simplified method for keeping track of the shear, either in the context of the usual beam element, or making use of Timoshenko finite elements (in the software programs where they are available). This helps avoid missing effects that are significant.

A second important point regards the effective existing constraints between one element and another, constraints that are often, strictly speaking, neither perfectly fixed nor perfectly hinged. The beam elements are clamped to each other from the very beginning, and it is only by means of suitable releases applied to the ends of one or another that they can simulate hinges. The most typical formulations do not provide for the use of either semi-continuous nodes or of rotational springs at the end of the elements, and so it is necessary to determine carefully the effective constraints between one element and another, in order to arrive at the most correct modelling. The whole set of releases at each of the two ends of each element with respect to the corresponding node, including the torsional component, has to be carefully determined.

A third point that can become quite important regards the possible eccentricity between the axes of the elements, including the presence of nodes whose dimensions are significant. This problem can be solved by providing rigid links, or in the absense of these, fictitious beam elements with a high degree of stiffness that will be exempted from successive checks. There also exists the possibility of defining beam elements that have a rigid eccentricity in them, which actually incorporate the rigid links just mentioned.

A fourth, fundamental aspect whose details needs to be looked after is that of the elements' orientation. The orientation is fundamental for the calculations. It is always necessary to check – if possible, with a graphic tool which gives an immediate picture – that the elements are correctly oriented (generally, as will be explained later, the "third node" is checked, but there are software programs that use other systems to orient the elements).

If it has been decided to use the energy of shear deformation, then the shear areas have to be determined quite accurately, keeping in mind that setting them equal to 0 does not usually mean that there are singular equations, but rather implies a return to the theory of bending without the contributions of shear: for this reason, the use of extremely small areas of shear to simulate a null value are to be avoided.

These schemes are laid out with regard to the axes of the element, and not their external edges.

In designing structures using the beam formulation, it should be borne in mind that the model can never take into account the normal stresses acting perpendicular to the element's axis, and thus if significant normal stresses of this kind are expected, they must be kept track of in a separate way.

A beam that is hinged at its ends is not modelled with a rod element but with a beam element with double end releases.

Whether a beam element or a rod element is used depends on whether there are moments of bending, torsion and shear. If there are moments of bending (even a single one), torsion or shear, then a beam element has to be used. If there is only axial force, then it is possible to use a rod element.

10.9 MIXED PROBLEMS

Without a doubt "mixed" problems are those that occur most frequently in engineering.

Mixed problems are those requiring the inclusion in a single model of finite elements that use different simplifying theories. Normally this is due to the fact that various structural elements can be indentified in the structure, and these can be modelled by means of different theories. Examples are the columns in a building and its bracing core in reinforced concrete, its floor slabs, and its foundation bed.

However, there are also cases in which the use of one or another type of element depends on the presumptions made about the kind of model that the analyst wishes to make.

For example, in an ordinary truss structure with parallel chords, we might want to model the two stringers with beam elements and the vertical struts with rod elements, so that the parasite bending related to the continuous construction of the stringers is not unaccounted for. Another example might be when we want to make a model entirely of rod elements, including the stringers, in order to create a situation where the calculation is immediately more clean, perhaps to facilitate pre-dimensioning.

Still another example might occur when, in the modelling of a shear wall, we must decide whether or not we want to take into account the bending inertia of the walls in a direction normal to its plane, because this determines whether plate elements or membrane elements are used.

In addition to these problems are added others related to the connection of radically different finite elements at the same node (plate-rod, membrane-beam, beam-rod, and so forth): what do such choices lead to?

The answer needs to be looked for case by case, bearing clearly in mind the formulation of each individual element and the "sensitive" degrees of freedom of each node. Taking each of the six degrees of freedom of the common node, we have to ask ourselves what this degree of freedom means for each element that is attached to the node, so that if there are components of the stress that "don't pass", these can be properly taken into account.

We will see better in chapter 12 how to deal with mixed problems. First it is necessary to understand the basics of how the most commonly used finite elements work.

11 THE MOST COMMON FINITE ELEMENTS

11.1 INTRODUCTION

This chapter will provide fundamental information relative to some of the finite elements used most widely in modelling by mid-range software programs today. It cannot, however, cover the entire range of the field, which is immense.

The majority of studies in the FEM are aimed at looking for formulations for finite elements that are reliable and efficient, and thus by now for any given problem there exists a vast amount of literature. Although there are numerous finite elements available, only some of them are frequently used: many finite elements have been found to entail difficulties of varying degrees, and others, while precise and reliable, are used in contexts that are so specialised that they are beyond the limits of this present work.

Here we shall briefly discuss the finite elements that have been so successfully applied that they are implemented in almost every solver available today. In what follows we will always be referring to the local tern of the element, without dealing with the element's collocation in space, unless explicitly stated otherwise. To indicate the variables of the local reference system we will use lower case letters (x, y, z) or numbers (1, 2, 3), while upper case letters will be used to refer to the global reference system (X, Y, Z). Greek letters will be used to indicate the local reference system of the master element (ξ, η, ζ).

11.2 TRUSS ELEMENTS

The truss element is used to model the axial behaviour of prisms, that is, *one-dimensional elements that are only subject to tension or compression, and not to bending, torsion or shear.* The tension or compression is assumed to be centred. The axis line of the truss element coincides with the location of the centroid of the structural element.

The truss element is by its nature incapable of stiffness in bending or shear, and thus the only stiffness that it can provide is that which opposes any shortening or lengthening of the element. Consequently, the node to which a truss is attached is not prevented from translation in a direction normal to the axis of the truss nor from any kind of rotation.

We can define the behaviour of a truss element once we have its area A, the elasticity modulus of the material E, the coefficient of thermal dilatation of the material α, and the specific weight of the material g. The behaviour of the truss doesn't depend on element's orientation with respect to its axis (hence no third node is necessary, except for purposes of interfacing with CAD software programs for drawing).

11.2.1 2-NODE TRUSS ELEMENTS

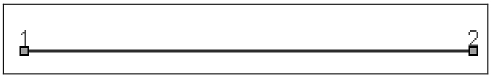


Figure 11-1: Truss element with two nodes

The element with two nodes is by far the most common kind. The element is rectilinear and has two nodes at the ends of the axis (node 1 and node 2[1]). There are no end releases, nor are end releases possible. The only unknown displacement function is u(x), which defines the displacements of the points on the element's axis in the direction of the axis itself (lengthening/shortening).

The essential character of the element lies in the polynomial approximation of the displacement function:

$$u(x) = u_1 + x(u_2-u_1)/L = (1-x/L)\, u_1 + x/L\, u_2 = u_1\, \mathbf{\Phi}_1 + u_2\, \mathbf{\Phi}_2 = \mathbf{\Phi u_e}$$

Thus the two shape functions $\mathbf{\Phi}_1$ and $\mathbf{\Phi}_2$ are:

$$\mathbf{\Phi}_1 = (1-x/L)$$

$$\mathbf{\Phi}_2 = x/L.$$

This approximation assumes that *the displacement varies linearly along the element*, and thus that *the strain and the axial force are constant along the points on the axis*:

$$\varepsilon(x) = \varepsilon = (u_2-u_1)/L$$

$$N = EA(u_2-u_1)/L.$$

From this we can see that if we have to look for a variation of axial force (such as that occurring, for example, when there are axially distributed loads), a single finite element is certainly not sufficient, but instead a series is needed, so that the various elements, with their constant axial forces, follow the variation of the axial force with a "stepped" diagram.

The axial stiffness of a truss has the well-known value

$$k = EA/L,$$

so the stiffness matrix **K** and the vector of nodal displacements $\mathbf{u_e}$ can be written as:

$$\mathbf{K} = \left|\begin{array}{c:c} \dfrac{EA}{L} & -\dfrac{EA}{L} \\ \hdashline -\dfrac{EA}{L} & \dfrac{EA}{L} \end{array}\right|$$

[1] The nodes can just as easily be designated node I and node J: what is important is that the concept is clear.

$$\mathbf{u_e} = \{u_{1,}\ u_2\}^T.$$

When there are no distributed axial loads (non-axial loads are not relevant to this element) the modelled response of the truss element is "exact", in the sense that it is identical to that obtained from the exact solution of the differential equations (always assuming linearly elastic material in first-order theory).

The truss element can be used in exactly the same way as a translational spring of stiffness k given by EA/L. When there are no springs it is thus possible to use this element as a spring; we need only regulate E, A and L suitably to obtain the desired stiffness.

The weight of the two-node truss element itself is usually transformed into merely translationally equivalent nodal forces parallel to the direction of the force of gravity and of an intensity equal to – at each node – half the overall weight of the truss.

11.2.2 3-NODE TRUSS ELEMENTS

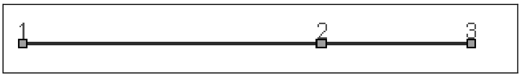


Figure 11-2: Truss element with three nodes

Programs with a richer library of finite elements offer truss elements with three nodes (1, 2, 3), in case the approximation of displacement u(x) is not linear but parabolic, where we can write

$$u(x) = \Phi_1 u_1 + \Phi_2 u_2 + \Phi_3 u_3$$

$$\Phi_1 = \frac{(x_2 - x)(x_3 - x)}{(x_2 - x_1)(x_3 - x_1)}$$

$$\Phi_2 = \frac{(x_1 - x)(x_3 - x)}{(x_1 - x_2)(x_3 - x_2)}$$

$$\Phi_3 = \frac{(x_2 - x)(x_1 - x)}{(x_2 - x_3)(x_1 - x_3)}.$$

Note that the dependence on x in each shape function is parabolic and that $\Phi_i(P_j)=0$, $\Phi_i(P_i)=1$. The 3-node element has a field of linear axial action in x, since linear functions are obtained when the shape functions are derived with respect to x.

11.3 Beam elements

11.3.1 Overview

Beam elements (and especially those with two nodes with a Euler-Bernoulli formulation) are widely used in technical practice for modelling the many kinds of frame structures. Since these kind of structures are found in civil, mechanical, naval and aeronautical engineering, we can see that beam elements are particularly important. They are undoubtedly the most commonly used elements by professionals who are not experts in the FEM, given that structures modelled with beams are practically ubiquitous.

Beam elements are used to model one-dimensional elements subject to forces *including* bending, shear and torsion. *The axial behaviour (that is, the behaviour of a truss element) is automatically included in the beam element, which thus also serves to model elements in tension or compression.* It is also possible to use beam elements to model elements that are only in tension or compression and are not subject to bending or torsion, but this is not especially efficient (except in very particular cases) because it makes calculations and meshing more difficult without adding any benefits.

It is possible to define beam elements that are curved (in which torsion and shear-bending are coupled) but here we will only discuss linear elements.

From the point of view of geometry, the beam element has an axis that corresponds to the location of the centroid of the element's cross-section. In some formulations it is possible to establish an eccentricity (or offset) between the axis line of the elements and its end nodes. This makes it possible to take into account the eccentricity of the elements and the carrying moments without explicitly modelling them (rigid elements).

The most widely-used finite elements are Saint Venant prisms, and do not model elements whose height, and thus whose inertia, is variable. Modelling elements with variable inertia requires the suitable subdivision of the structural element into blocks (which will then become the finite elements) of constant height and inertia. The formulation of finite elements with variable inertia generally assumes that the moments of inertia and the area of the sections vary along the axis in a simple way (for example, according to laws of linear or parabolic variation), a circumstance not found, however, in the majority of real elements with variable inertia.

In order for a beam element to be defined, it is necessary to know the Young's modulus for the material it is made of, presuming that we are considering the strains acting along the element's axis. In the case of corrections related to the shear strain, it is also necessary to define, for the given material, the shear modulus of elasticity G. As for the case of truss elements, the coefficient of thermal dilatation α and specific weight g are also required.

However, it is not sufficient to define only the material properties, because the axial and bending behaviours depend on the geometric characteristics of the section as well.

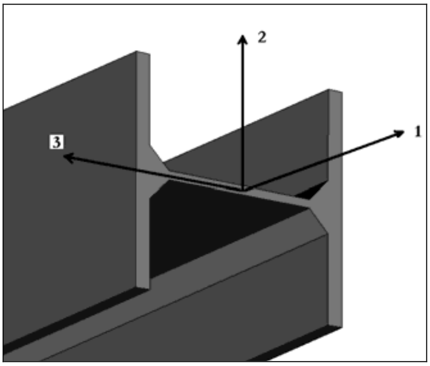


Figure 11-3: The local tern of a beam element represented in 3-D.

In defining the element we assume that the local tern of reference possesses these properties: local axis 1 corresponds to the axis of the element; local axis 2 corresponds to the first of the two principal axes of the section; local axis 3 corresponds to the second principal axis of the section (fig. 11-3). Due to these conditions and to the linearity of the element, we have the following fundamental properties:

1. The axial behaviour is decoupled from that of bending or torsion;
2. Behaviour in torsion is decoupled from axial and bending behaviours;
3. Bending about local axis 2 is decoupled with respect to that about local axis 3 and from axial and torsional behaviours;
4. Bending about local axis 3 is decoupled with respect to that of local axis 2 and from axial and torsional behaviours;
5. The shear acting in local direction 3 (V_3) is coupled to the bending moment action along local axis 2 (M_2);
6. The shear acting in local direction 2 (V_2) is coupled to the bending moment acting along local axis 3 (M_3).

It is necessary to know the following characteristics of the section: area A, moment of torsional inertia I_t, moments of bending inertia about the main axes I_2 and I_3. If shear strain energy is present, then it is also necessary to know the two shear factors χ_2 and χ_3 or, equivalently (and more frequently), the two shear areas A_2 and A_3 that derive from these:

$$A_2 = A/\chi_2$$

$$A_3 = A/\chi_3.$$

Given these characteristics, and on the basis of the hypotheses made, we need only model the following three behaviours separately and independently:

- axial behaviour;
- torsional behaviour;
- bending behaviour (and an analogous model for bending about axis 2 and bending about axis 3).

All three of these behaviours are found in beam elements and are modelled differently according to the element's formulation and number of nodes.

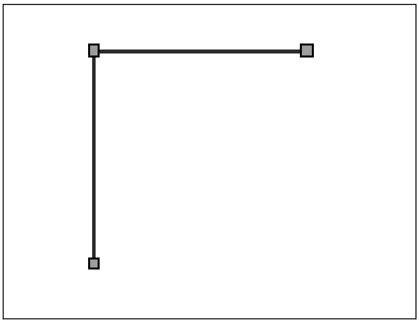


Figure 11-4: Beam elements are created connected to one another

A beam element is usually created completely connected at its end nodes. Thus if two beam elements have a common node and there are no end releases, they turn out to be connected to each other. In fact, the movement of the node gives rise to some quantity of strain in all the elements connected at that node.

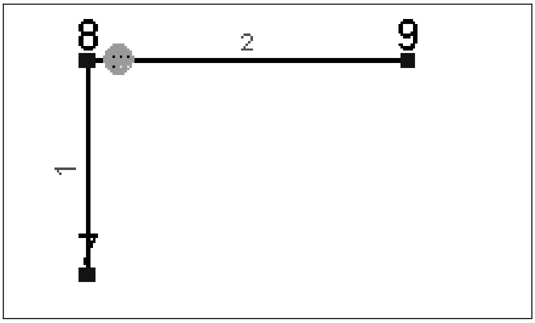


Figure 11-5: Beam elements. Element 2 is end released, and because of this the rotation of node 8 is not transmitted to element 2. This is equivalent to there being a hinge between elements 1 and 2.

If we want two elements to be hinged, we have consider the following (cf. also chap. 12): one of the two elements remains attached to its node without disconnections; the other element is disconnected with respect to the end node such that the rotation of the node is not transmitted by congruence to the end of the element. In order to do this, *end releases* are applied to the disconnected element.

If there are no end releases, a beam element allows stiffnesses to pass to its nodes and they are capable of limiting any possible degrees of freedom, that is, any translation and rotation. Thus if a beam element is attached to a node and is not released, then that node cannot give rise to any pseudo-mechanism. Naturally, the situation is different where releases have been applied or if the characteristics of the material or the section are special (as long as the software program being used permits this).

11.3.2 2-NODE EULER-BERNOULLI ELEMENTS

Elements with two nodes based on the Euler-Bernoulli formulation for slender beams are by far the most widely used. The element comprises axial behaviour (in exactly the same way as for truss elements already discussed) as well as torsional and bending components. In this section we will discuss the various aspects.

11.3.2.1 THE TORSIONAL COMPONENT, SHAPE FUNCTIONS

For the torsional component the function θ(x) is unknown. This function gives the rotation of a face on abscissa x with respect to the undeformed configuration, and is a rotation of the section about the element's axis. Since it is possible to consider the two rotations at the ends of the elements as unknown nodal displacements, not necessarily oriented according to the global axis, the interpolation formula for the rotation field is:

$$\theta(x) = \theta_1 + x(\theta_2 - \theta_1)/L = (1-x/L)\,\theta_1 + x/L\,\theta_2 = \theta_1\,\Phi_1 + \theta_2\,\Phi_2$$

Thus the two shape functions Φ_1 and Φ_2 are:

$$\Phi_1=(1-x/L)$$

$$\Phi_2= x/L.$$

This approximation assumes that *the rotation varies linearly along the element* and thus *that the torsional moment is constant along the points of the axis line:*

$$\theta' = (\theta_2 - \theta_1)/L$$

$$M_t = GI_t\,(\theta_2 - \theta_1)/L.$$

11.3.2.2 THE BENDING COMPONENT, SHAPE FUNCTIONS

The bending component is modelled by considering the bending in the two main planes individually.

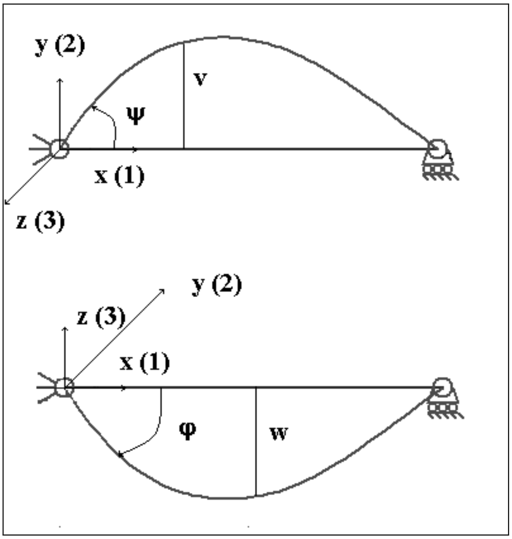


Figure 11-6: Coupling of the variables of displacement in bending.

In the local reference system this means (assuming that u is the translation in the direction of axis x (or 1), v in that of the local axis y (or 2, as the case may be), w in the direction of z (or 3) modelling the two unknown displacement functions: v(x) for the bending about axis 3; and w(x), for the bending about axis 2.

The displacement function has to be continuous with its first derivative (or as we say technically, it has to belong to C^1) so that it respects congruence: cusps in the axis lines are not permitted. Thus it is necessary to consider as nodal displacements not only the two translations, but also the rotations of the end nodes.

This furnishes us with four nodal displacements for interpolating each of the unknown functions v(x) and w(x), according to the following correspondence (recalling that φ is the rotation of the node about axis 2, and ψ is the rotation about axis 3):

Bending about axis 3	v(x)	depends on	v_1, v_2, ψ_1, ψ_2
Bending about axis 2	w(x)	depends on	$w_1, w_2, \varphi_1, \varphi_2$.

Since four parameters define a cubic, the interpolation function for bending displacement is a cubic, as are the shape functions. If for example we concentrate on the interpolation of v(x) associated to moment M_3, we can write

$$v(x)= v_1\Phi_1+\psi_1\ \Phi_2+v_2\ \Phi_3+\psi_2\ \Phi_4$$

where the shape functions are expressed as follows:

$$\Phi_1 = 1 - \frac{3x^2}{L^2} + 2\frac{x^3}{L^3}$$

$$\Phi_2 = x - 2\frac{x^2}{L} + \frac{x^3}{L^2}$$

$$\Phi_3 = 3\frac{x^2}{L^2} - 2\frac{x^3}{L^3}$$

$$\Phi_4 = -\frac{x^2}{L} + \frac{x^3}{L^2}.$$

The shapes are shown in fig. 11-7.

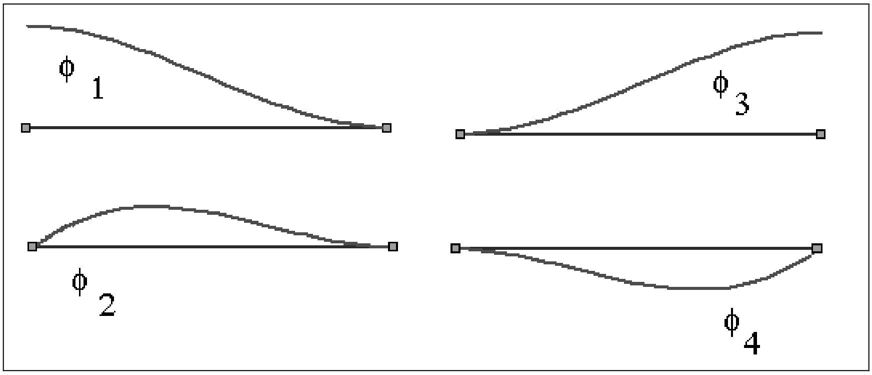


Figure 11-7: Shape functions for a beam element (they refer to v translation and ψ rotation: shape functions 2 and 4 associated to φ rotation should have sign changed, due to right hand rule).

Recalling that M=-EIv″, V=M′(x) and taking, for example, the first of the shape functions for the deformed configuration of the beam, we obtain:

$$M(x) = -EI(-6/L^2+12x/L^3)v_1$$

$$V(x) = -12(EI/L^3)v_1.$$

Thus the couples C and forces F to be applied to the nodes in order to obtain the unit distortion with respect to the first of the shape functions (v_1=1 and $v_2=\psi_1=\psi_2=0$) are the following:

at the first end: $C=6EI/L^2$ $F=12EI/L^3$

at the second end: $C=6EI/L^2$ $F=-12EI/L^3$

Note that these forces balance each other out.

If we want to see what happens with the second shape function (ψ_1=1 and $v_1=v_2=\psi_2=0$), we obtain

$$M(x) = -EI(-4/L^2+6x/L^3)\ \psi_1$$

$$V(x) = -6(EI/L^2)\ \psi_1,$$

and so

at the first end: $C=4EI/L^2$ $F=6EI/L^2$

at the second end: $C=2EI/L^2$ $F=-6EI/L^2$

The forces and the couples that we have found are the direct and indirect stiffnesses with which the beam opposes the movement of the nodes. In the next section we will find these terms and others when we discuss how to write the complete stiffness matrix.

11.3.2.3 The complete model

By combining all the various formulations (axial, torsional, bending about axis 2 and bending about axis 3) it is possible to arrive at a single representation of the beam element in matrix form which follows the process discussed in chapter 4. In what follows we give the various vectors and matrices (see also the review already presented in chapter 3).

$$\mathbf{U}=\{u, v, w, \theta\}^T$$

$$\mathbf{U}=\mathbf{\Phi u_e}$$

$$\Phi = \left|\begin{array}{ccc:ccc:ccc:ccc}
1-\frac{x}{L} & 0 & 0 & 0 & 0 & 0 & \frac{x}{L} & 0 & 0 & 0 & 0 & 0 \\
\hdashline
0 & 1-\frac{3x^2}{L^2}+2\frac{x^3}{L^3} & 0 & 0 & 0 & x-2\frac{x^2}{L}+\frac{x^3}{L^2} & 0 & 3\frac{x^2}{L^2}-2\frac{x^3}{L^3} & 0 & 0 & 0 & -\frac{x^2}{L}+\frac{x^3}{L^2} \\
\hdashline
0 & 0 & 1-\frac{3x^2}{L^2}+2\frac{x^3}{L^3} & 0 & -\left(x-2\frac{x^2}{L}+\frac{x^3}{L^2}\right) & 0 & 0 & 0 & 3\frac{x^2}{L^2}-2\frac{x^3}{L^3} & 0 & -\left(-\frac{x^2}{L}+\frac{x^3}{L^2}\right) & 0 \\
\hdashline
0 & 0 & 0 & 1-\frac{x}{L} & 0 & 0 & 0 & 0 & 0 & \frac{x}{L} & 0 & 0
\end{array}\right|$$

$$\mathbf{u_e}=\{u_1, v_1, w_1, \theta_1, \varphi_1, \psi_1, u_2, v_2, w_2, \theta_2, \varphi_2, \psi_2\}^T.$$

Now, recalling what was found in chapter 3, we can write the law relating strains to displacements:

$$\boldsymbol{\varepsilon}=\{u', \theta', -w'', -v''\}^T=\mathbf{Bu_e}$$

$$\mathbf{B}=\mathbf{C\Phi}$$

$$\mathbf{C}=\begin{vmatrix} \frac{\partial}{\partial x} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\partial}{\partial x} \\ 0 & 0 & -\frac{\partial^2}{\partial x^2} & 0 \\ 0 & -\frac{\partial^2}{\partial x^2} & 0 & 0 \end{vmatrix}$$

$$\mathbf{B}=\left|\begin{array}{ccc:ccc:ccc:ccc} -\frac{1}{L} & 0 & 0 & 0 & 0 & 0 & \frac{1}{L} & 0 & 0 & 0 & 0 & 0 \\ \hdashline 0 & 0 & 0 & -\frac{1}{L} & 0 & 0 & 0 & 0 & 0 & \frac{1}{L} & 0 & 0 \\ \hdashline 0 & 0 & \frac{6}{L^2}-\frac{12x}{L^3} & 0 & -\left(\frac{4}{L}-\frac{6x}{L^2}\right) & 0 & 0 & 0 & -\frac{6}{L^2}+\frac{12x}{L^3} & 0 & -\left(\frac{2}{L}-\frac{6x}{L^2}\right) & 0 \\ \hdashline 0 & \frac{6}{L^2}-\frac{12x}{L^3} & 0 & 0 & 0 & \frac{4}{L}-\frac{6x}{L^2} & 0 & -\frac{6}{L^2}+\frac{12x}{L^3} & 0 & 0 & 0 & \frac{2}{L}-\frac{6x}{L^2} \end{array}\right|$$

At this point we recall the relationship between strains and stresses:

$$\boldsymbol{\sigma}=\{N, M_1, M_2, M_3\}^T=\mathbf{D}\boldsymbol{\varepsilon}$$

$$\mathbf{D}=\begin{vmatrix} EA & 0 & 0 & 0 \\ 0 & GI_t & 0 & 0 \\ 0 & 0 & EI_y & 0 \\ 0 & 0 & 0 & EI_z \end{vmatrix}.$$

We have written the matrices in their extended form in order to make clear the relationship that was explained in chapter 4: we are following the same process already outlined. The stiffness matrix is obtained by determining:

$$\mathbf{K}=\int \mathbf{B}^{\mathbf{T}}\mathbf{D}\mathbf{B}\,\mathrm{d}L.$$

Practically speaking, by taking into account the decoupling of the axial, torsional and bending behaviours and the constancy of **D**, it can be seen that the calculation is reduced to just a few integrals, which can be obtained substantially by the following expressions:

Axial stiffness (as for the truss element): EA/L

Torsional stiffness: GI_t/L

Bending in terms of direct stiffness:

translation of the node: $$EI\int_0^L\left(\frac{6}{L^2}-\frac{12x}{L^3}\right)^2 dx=\frac{12EI}{L^3}$$

rotation of the node: $$EI\int_0^L\left(\frac{4}{L}-\frac{6x}{L^2}\right)^2 dx=\frac{4EI}{L}$$

rotation of the node: $$EI\int_0^L\left(\frac{2}{L}-\frac{6x}{L^2}\right)^2 dx=\frac{4EI}{L}$$

Bending in terms of indirect stiffness:

moment due to translation: $$EI\int_0^L\left(\frac{6}{L^2}-\frac{12x}{L^3}\right)\left(\frac{4}{L}-\frac{6x}{L^2}\right)dx=\frac{6EI}{L^2}$$

force due to rotation: $$EI\int_0^L\left(\frac{6}{L^2}-\frac{12x}{L^3}\right)\left(\frac{2}{L}-\frac{6x}{L^2}\right)dx=\frac{6EI}{L^2}$$

moment due to rotation: $$EI\int_0^L\left(\frac{4}{L}-\frac{6x}{L^2}\right)\left(\frac{2}{L}-\frac{6x}{L^2}\right)dx=\frac{2EI}{L}.$$

It is clear that we have once again found the terms that we had already calculated on the basis of the considerations regarding equilibrium in the previous section.

Hence, the final stiffness matrix (12x12) is as follows (using the right hand rule for positive rotations):

$$\mathbf{K}=\left|\begin{array}{ccc|ccc|ccc|ccc}
\frac{EA}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\
 & \frac{12EI_3}{L^3} & 0 & 0 & 0 & \frac{6EI_3}{L^2} & 0 & -\frac{12EI_3}{L^3} & 0 & 0 & 0 & \frac{6EI_3}{L^2} \\
 & & \frac{12EI_2}{L^3} & 0 & -\frac{6EI_2}{L^2} & 0 & 0 & 0 & -\frac{12EI_2}{L^3} & 0 & -\frac{6EI_2}{L^2} & 0 \\
\hline
 & & & \frac{GI_t}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{GI_t}{L} & 0 & 0 \\
 & & & & \frac{4EI_2}{L} & 0 & 0 & 0 & \frac{6EI_2}{L^2} & 0 & \frac{2EI_2}{L} & 0 \\
 & & & & & \frac{4EI_3}{L} & 0 & -\frac{6EI_3}{L^2} & 0 & 0 & 0 & \frac{2EI_3}{L} \\
\hline
 & & & & & & \frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\
\text{symmetrical} & & & & & & & \frac{12EI_3}{L^3} & 0 & 0 & 0 & -\frac{6EI_3}{L^2} \\
 & & & & & & & & \frac{12EI_2}{L^3} & 0 & \frac{6EI_2}{L^2} & 0 \\
\hline
 & & & & & & & & & \frac{GI_t}{L} & 0 & 0 \\
 & & & & & & & & & & \frac{4EI_2}{L} & 0 \\
 & & & & & & & & & & & \frac{4EI_3}{L}
\end{array}\right|$$

If we want to interpret these terms physically, we can say that each row represents in an ordered way the forces to be applied to the element's nodes (first and second) when they are assigned a distortion in which the degree of freedom corresponding to the row in question is set equal to 1, and all the others are null.

For example, let's look at the first row (fig. 11-8). We move the first node in direction u (axis 1) by a unitary quantity. To the first node we have to apply a direct force like axis 1 equal to EA/L, to the second node a force that is equal and opposite equal, –EA/L.

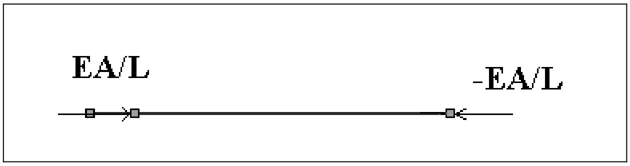


Figure 11-8: Unit distortion and associated stiffness.

Let's now look at the third row (fig. 11-9). We move the first node in direction w (axis 3) by a unitary quantity. To that node we have to apply: a force equal to $12EI_2/L^3$ in the direction of axis 3; and a couple equal to 6 EI_2/L^2 in a negative direction about axis 2. Meanwhile, to the second node we have to apply: a negative force equal to $-12EI_2/L^3$ in the direction of axis 3; and a couple equal to 6 EI_2/L^2 in a negative direction about axis 2.

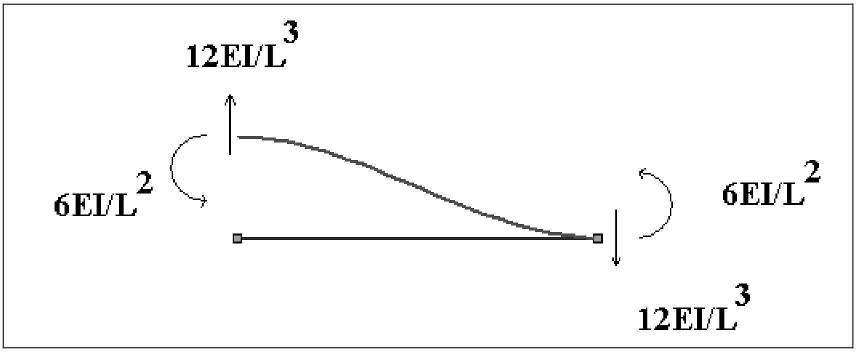


Figure 11-9: Unit distortion and associated stiffness.

Now let's look at the twelfth row (fig. 11-10). Here we rotate the second node about local axis 3 (rotation ψ=1). To the second node we have to apply: a positive couple equal to $4EI_3/L$ in the direction of axis 3; and a negative force in the direction of axis 2 equal to -6 EI_3/L^2. To the first node we have to apply: a positive couple equal to $2EI_3/L$ in the direction of axis 3; and a positive force in the direction of axis 2 equal to $6\ EI_3/L^2$.

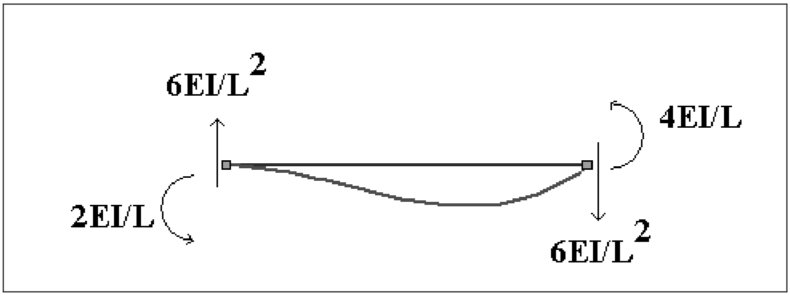


Figure 11-10: Unit distortion and the associated stiffness.

11.3.2.4 THE PROBLEM OF THE ORIENTATION IN SPACE

In contrast to the truss element, the beam element needs to be oriented in space, because the direction of the axis line is not sufficient to define the element's bending behaviour.

Various different methods have been proposed and adopted to orient the element in space, and each has its pros and cons. The most widely used method is probably that of the third node. Practically speaking, the finite element is oriented by providing, in addition to the first node N1 (or node I) and second node N2 (or node J) a third node N3 (or node K) which together make it possible to define a plane. Obviously the three nodes N1, N2 and N3 cannot lie on the same line, because this would make it impossible to define the plane unequivocally. By convention, axis 2 of the local tern lies in the plane defined by nodes N1, N2 and N3, and it is "on N3's side", that is, that the scalar product axis 2 – (N3-N1) is positive. In practice, it is necessary to bear in mind that the two vectors (N3-N1) and (N2-N1) do not necessarily have to be orthogonal, while local axis 2 must be perpendicular to local axis 1, defined by the versor of N2-N1.

Figure 11-11 gives a diagrammatic representation of the third node method of orientation. We can see that:

- The direction of vector (N2-N1) is the same as that of axis 1 of the local tern;
- Local axis 2 lies in the plane defined by N1, N2 and N3;
- Axis 3 makes a right tern with axes 1 and 2, at right angles to them;
- The scalar product of axis 2 and (N3-N1) is positive, that is, the two vectors form a angle equal to less than 180°.

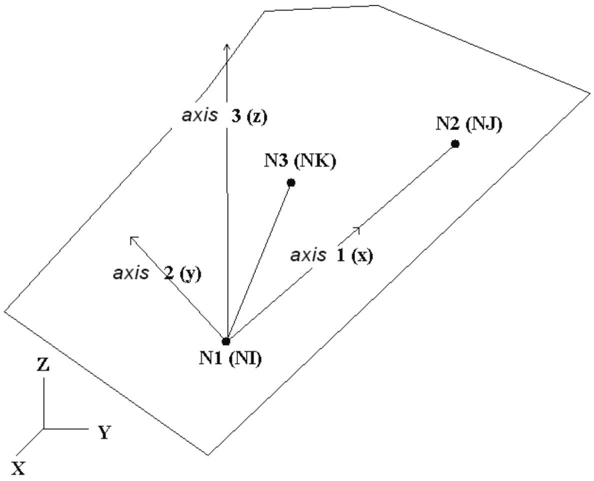


Figure 11-11: Orientation in space using the third node.

Any node can be used to orient the other elements, and if there is no suitable structural node, then it is possible to create an ad hoc node, bearing in mind that if this node only serves to orient one or more beam elements, then it has to be completely constrained.

Some software programs define certain nodes in suitable positions in order to establish standard orientations. The nodes involved are the first six nodes of each model, and their position is at infinity in the three directions as shown in table 11-1.

Node number	At infinity in direction:
1	+Y
2	+Z
3	+X
4	-Y
5	-Z
6	-X

Table 11-1: Predefined nodes.

When we say "infinity" we are obviously talking about a numerical infinity. These are predefined, pre-constrained nodes set at great distances from the origin in directions Y, Z and X in both positive and negative directions. A sample distance might be of the order of 10^{14}mm. Using the six predefined nodes means that only in rare cases is it necessary to define additional nodes. In order to explain this method clearly, table 11-2 shows which third node should be used for the most commonly used elements and positions:

Structural element	Third node number
Column axis 2 parallel to axis X	3
Column axis 2 parallel to axis Y	1
Beam parallel to X, axis 2 parallel to Y	1
Beam parallel to Y, axis 2 parallel to X	3
Beam parallel to X, axis 2 parallel to Z	2
Beam parallel to Y, axis 2 parallel to Z	2

Table 11-2: Third node to be used for typical structural elements.

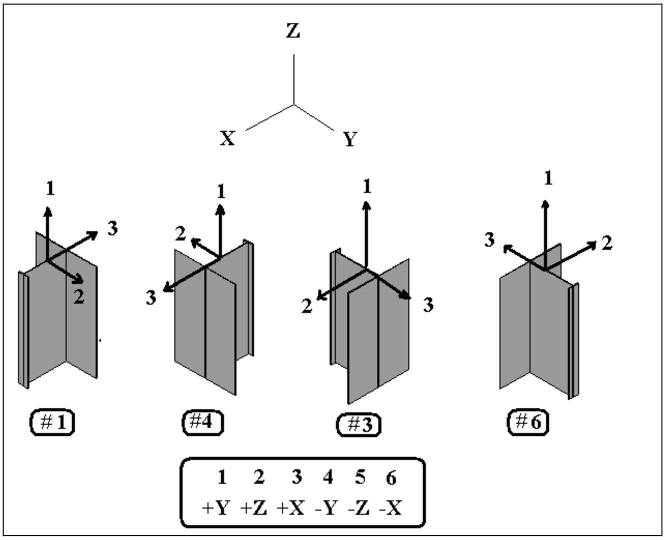


Figure 11-12: Examples of column orientations using the six predefined nodes (from the SARGON program).

It should be noted that an incorrect orientation in space can lead to results that are completely erroneous, since switching the moments of inertia or incorrectly orienting the tern gives rise to errors that can sometimes be quite significant. Making sure that the elements are correctly oriented is one of the fundamental checks that must always be performed on any model containing beam elements.

The system of using the third node is not the only method available. Some FEM software programs, such as SAP2000®, make use of a typical orientation and then, in special cases, make it possible to define an angle of deviation from that orientation. In the case of SAP2000® the typical orientation is defined so that:

- Plane 1-2 is always vertical;
- Axis 2 points in direction +Z except when the element is vertical, in which case axis 2 is assumed to point in direction +X;
- Local axis 3 is always horizontal, that is, it lies in plane XY.

Still other programs, such as ANSYS®, use the convention of the third node, but for defining plane 1-3 instead of 1-2, and make it possible to use more than one method for orienting the element.

A thorough study of the system adopted by the software program being used is essential for avoiding banal errors that might invalidate the analysis. The question of how orientation is established is one of the first things to understand about any software program, and information regarding this can be found in the program's instruction manual.

11.3.2.5 End releases

As we noted earlier, a beam element is created rigidly attached at its end nodes, which are connected to it in such a way that a rotation or translation of those nodes will be communicated unaltered to the end point of the axis line that corresponds to the node that moved. Naturally it might be necessary for the node to be able to rotate or translate without communicating an analogous movement to the axis line, which will thus turn out to be released with respect to it. In this case the behaviour of the beam will be equivalent to that of a hinge or a sliding block, depending on the kind of release adopted.

In applying the releases it must be kept in mind that in any case each degree of freedom of every node has to be associated with a stiffness, because if not, a pseudo-mechanism is introduced. Let's consider the examples shown in fig. 11-13, which is doubly hinged at the ends of the cross-member. A first way of modelling this hinge is shown in A: the ends of the cross-member are released with respect to the end nodes, which is a correct solution; B is also correct, and shows that the end of the column is released but the end of the beam is not; in contrast, C is correct only if the rotations of the two end nodes are constrained. If both of the elements attached at the right or left nodes are released, then these nodes are left without any stiffness associated to the rotation of the node itself, which thus has to be constrained.

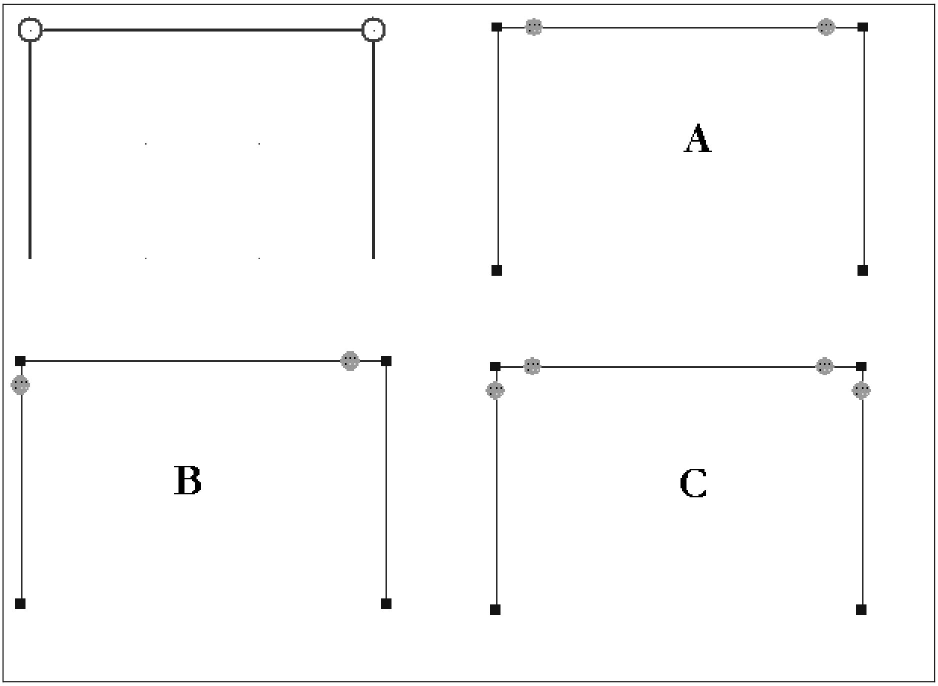


Figure 11-13: Examples of applications of releases, and the pseudo-mechanism that can arise from them: A correct, B correct, C node pseudo-mechanism if not constrained for rotation.

When applying releases, care must be taken to avoid situations such as that shown in figure 11-14, because they give rise to local mechanism. Table 11-3 lists the specific release templates to avoid.

From an analytical point of view, the stiffness matrix of the released element can be obtained by a procedure known as *condensation*, for which we will now discuss the fundamental guidelines.

When there are no distributed or nodal loads, for each element it is possible to write the following matrix equation:

$$\mathbf{Ku=r}$$

where **r** is the vector of the nodal reactions. Using "c" to indicate the connected degrees of freedom, and "u" the disconnected degrees of freedom, and after having reordered the equations and the unknowns, the preceding equation can be rewritten in the following way:

$$\begin{vmatrix} \mathbf{K}_{cc} & \mathbf{K}_{cu} \\ \mathbf{K}_{uc} & \mathbf{K}_{uu} \end{vmatrix} \begin{vmatrix} \mathbf{u}_c \\ \mathbf{u}_u \end{vmatrix} = \begin{vmatrix} \mathbf{r}_c \\ \mathbf{r}_u \end{vmatrix}.$$

Now we should observe that the reactive forces $\mathbf{r}_u$ on the disconnected degrees of freedom have to be null, hence the system becomes

$$\begin{vmatrix} \mathbf{K}_{cc} & \mathbf{K}_{cu} \\ \mathbf{K}_{uc} & \mathbf{K}_{uu} \end{vmatrix} \begin{vmatrix} \mathbf{u}_c \\ \mathbf{u}_u \end{vmatrix} = \begin{vmatrix} \mathbf{r}_c \\ \mathbf{0} \end{vmatrix}.$$

Solving the second matrix equation with respect to $\mathbf{u}_u$ and substituting the resulting value in the first equation, we obtain the following matrix relationship:

$$(\mathbf{K}_{cc}-\mathbf{K}_{cu}\mathbf{K}_{uu}^{-1}\mathbf{K}_{uc})\mathbf{u}_c=\mathbf{r}_c$$

which gives in parentheses the expression of the new stiffness matrix reduced to the order of $\mathbf{K}_{cc}$.

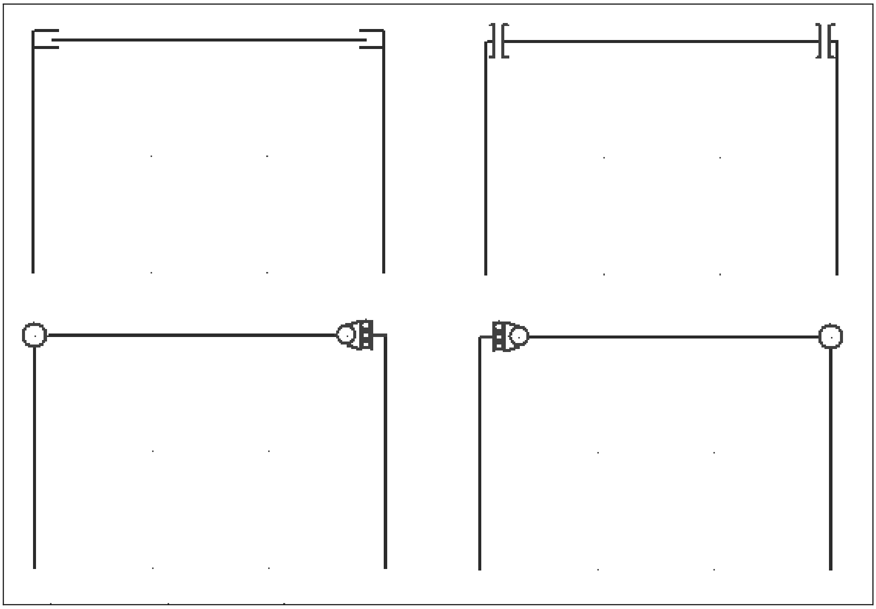


Figure 11-14: Releases that are not permitted.

This matrix can then be expanded to the original dimension by setting as null all the values corresponding to the disconnected (and condensed) degrees of freedom. It should be noted that in order for the procedure to work, the square matrix $\mathbf{K}_{uu}$ has to be invertible, and thus the releases – as we have already said – must not give rise to any situations where there are mechanisms.

Description	First end	Second end
Double sleeve	T1	T1
Double sliding block	T2	T2
Double sliding block	T3	T3
Double torsional release	R1	R1
Ill-disposed roller	R2	T3, R2
Ill-disposed roller	R2, T3	R2
Ill-disposed roller	R3	T2, R3
Ill-disposed roller	R3, T2	R3

Table 11-3: Combinations of releases that should not be applied (T=translation, R=rotation).

11.3.2.6 Corrections related to the shear factor

In chapter 3 we saw that completely modelling a Timoshenko beam would require holding two displacement functions as unknowns rather than only one. Thus the element that we described is not suitable for modelling the effects of shear in the general case. However, *when there are no distributed loads*, the exact solution for the deflection w coincides with a cubic, and the shape function adopted for this element is also cubic. In this case the shear is constant (as is the correction α(x)) and it is relatively easy to calculate the exact solution for the stiffnesses. In the Timoshenko theory the displacement does not coincide with that of the shape functions (although these are also cubics), because the rotation of the node does not coincide with w′ (in fact, w′ cannot be null where φ=0).

Even though the rotation of the node does not coincide with the first derivative of w, and this precisely because of the contribution of the shear factor, it is possible to exploit the same kind of shape function to determine the stiffness matrix with the Timoshenko theory, providing that the terms of stiffness have been corrected in a suitable way.

Practically speaking, the stiffnesses already found for the slender element are multiplied by a series of dimensionless corrective factors that tend to 1 as the contribution of shear to the strain energy tends to 0. This means that we can define six corrective factors, three for bending about axis 2 and three (homologous) for bending about axis 3.

$$A_2 = \frac{3K_2 + 1}{12K_2 + 1}$$

$$B_2 = \frac{1}{12K_2 + 1}$$

$$C_2 = \frac{1 - 6K_2}{12K_2 + 1}$$

$$A_3 = \frac{3K_3 + 1}{12K_3 + 1}$$

$$B_3 = \frac{1}{12K_3 + 1}$$

$$C_3 = \frac{1 - 6K_3}{12K_3 + 1},$$

having set K_2 for bending about axis 2 as:

$$K_2 = \frac{\chi_3 EI_2}{GAL^2} = \frac{EI_2}{GA_3L^2}$$

and K_3 for bending about axis 3 as:

$$K_3 = \frac{\chi_2 EI_3}{GAL^2} = \frac{EI_3}{GA_2L^2}$$

where A_2 and A_3 are the two shear areas associated with shear V_2 and shear V_3, respectively.

When these corrective factors are introduced, the stiffness matrix **K** for the element becomes:

$$\mathbf{K} = \left[\begin{array}{ccc:ccc|ccc:ccc}
\frac{EA}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\
 & \frac{12EI_3}{L^3}B_3 & 0 & 0 & 0 & \frac{6EI_3}{L^2}B_3 & 0 & -\frac{12EI_3}{L^3}B_3 & 0 & 0 & 0 & \frac{6EI_3}{L^2}B_3 \\
 & & \frac{12EI_2}{L^3}B_2 & 0 & -\frac{6EI_2}{L^2}B_2 & 0 & 0 & 0 & -\frac{12EI_2}{L^3}B_2 & 0 & -\frac{6EI_2}{L^2}B_2 & 0 \\
\hdashline
 & & & \frac{GI_t}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{GI_t}{L} & 0 & 0 \\
 & & & & \frac{4EI_2}{L}A_2 & 0 & 0 & 0 & \frac{6EI_2}{L^2}B_2 & 0 & \frac{2EI_2}{L}C_2 & 0 \\
 & & & & & \frac{4EI_3}{L}A_3 & 0 & -\frac{6EI_3}{L^2}B_3 & 0 & 0 & 0 & \frac{2EI_3}{L}C_3 \\
\hline
 & & & & & & \frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\
\text{symmetrical} & & & & & & & \frac{12EI_3}{L^3}B_3 & 0 & 0 & 0 & -\frac{6EI_3}{L^2}B_3 \\
 & & & & & & & & \frac{12EI_2}{L^3}B_2 & 0 & \frac{6EI_2}{L^2}B_2 & 0 \\
\hdashline
 & & & & & & & & & \frac{GI_t}{L} & 0 & 0 \\
 & & & & & & & & & & \frac{4EI_2}{L}A_2 & 0 \\
 & & & & & & & & & & & \frac{4EI_3}{L}A_3
\end{array}\right]$$

This corrected stiffness matrix is widely used in practice. It has to be borne in mind that the results obtained with this kind of stiffness matrix are not the same as those obtained with the original stiffness matrix: it contains a correction that can be quite significant where the beams are thick and short.

The corrected terms of stiffness descend directly from the Timoshenko beam theory, and can be calculated using it. For example, if we want to calculate the stiffness terms relative to a unit displacement of the second end (shape function Φ_3), we perform the following calculations:

Deflection v solution to the differential equation (a, b, c, d, to be determined):

$$v = ax^3 + bx^2 + cx + d.$$

Boundary conditions

$$v(0)=0$$

$$\psi(0)=0$$

$$v(L)=1$$

$$\psi(L)=0$$

Note that a condition has been imposed on rotation ψ which does not coincide with v'. Recalling that

$$\psi = v' - \beta$$

we obtain the following solutions for a, b, c, d:

$$a=2(\beta L-1)/L^3$$

$$b=-3(\beta L-1)/L^2$$

$$c=\beta$$

$$d=0.$$

Now if we recall (cf. chapter 3) that $-EIv''=M(x)$ and that $-EIv'''=V(x)$ (given that $M'=V$), and that $V=GA\beta/\chi$, we find that

$$\beta L=12K/(12K+1)$$

since

$$K = \frac{\chi EI}{GAL^2}.$$

The moment at the first and second end is thus:

$$M = \mp\frac{6EI}{L^2}\left(\frac{1}{12K+1}\right) = \mp\frac{6EI}{L^2}B$$

while the shear is

$$V = \frac{12EI}{L^3}\left(\frac{1}{12K+1}\right) = \frac{12EI}{L^3}B .$$

Except for the sign, which is different by convention, the terms that we have found coincide with the stiffness terms that can be found in the matrix, in correspondence to the unitary translation of one of the nodes (in the direction of either axis 2 or axis 3). For example, for the translation of the second node in the direction of 2, v (row 8), the term $K_{8,8}$ and the term $K_{8,12}$ coincide with those found.

Analogous calculations allow us to determine all the other stiffness terms.

If we examine the exact function v(x) for the case that we have just discussed, we will find that (substituting a, b, c, d), it has to be

$$v(x) = \frac{2(\beta L-1)}{L^3}x^3 - \frac{3(\beta L-1)}{L^2}x^2 + \beta x, \qquad [11\text{-}1]$$

while, in contrast, when the shape function of the element is not corrected for shear we have

$$\mathrm{v(x)} = 3\frac{x^2}{L^2} - 2\frac{x^3}{L^3}, \qquad [11\text{-}2]$$

and the two functions coincide if βL tends to zero, as expected. Thus, in terms of displacement, *the 2-node element with corrections for shear is not "exact" even when there are no distributed loads.* However, the error concerns only the displacement field, and only the points within the element, and not the ends (where the shape functions are equal to 1), because the terms of stiffness are correctly determined.

It should be noted that eliminating the effects of shear corresponds to a null shear factor, and thus to an infinite shear area. Since it is usually precisely the shear area that has to be provided as input data, we want to bear in mind that normally it makes sense to assume that the effect related to shear is null if the shear area is null, even if rigorously speaking this could have a K factor that tends to infinity (and thus A tending to 0.25, B tending to 0, and C tending to – 0.5). In other words, if we want to eliminate the effect of shear, we have to set the shear area as null in the input data, since it is usually a good idea to associate the elimination of shear effects to a null shear area.

11.3.2.7 BEAM ELEMENT PRECISION

11.3.2.7.1 THIN BEAMS (NO CORRECTION FOR THE EFFECTS OF SHEAR)

From what has been discussed, we can establish that the 2-node beam element is precise when there are no distributed loads. Thus we can be sure that all elements that are not directly loaded with concentrated or distributed forces can be modelled with a single finite element. In the case where there are concentrated loads, to arrive at the exact solution we need only divide the element so that we have a node under each load. In the case of a uniformly distributed load, we should subdivide the elements into a certain number of parts, so that we can arrive at a suitable approximation of the strains and stresses. With regard to this, it should be kept in mind that with a 2-node element we expect to find a linear bending moment and a constant shear, and thus as the moment diagram becomes more complex, the number of elements that the beam has to be divided into increases.

This line of reasoning is valid if the software program treats the beam element the same way it treats other finite elements, that is, the local contribution of stress, that related to distributed loads, is neglected. In actual fact, the programs most equipped to deal with frame structures, or programs that are simply more complete, take into account the fact that in the case of beam elements, it is easy to add the stress due to distributed loads at fixed nodes to the stress due to the movement of the nodes, thus arriving at the exact solution. The overall stress is in fact the sum of the stress due to distributed loads when the nodes are fixed and the stress due to distortions imposed by the nodes of the element. We have seen that this second part is modelled in an exact way, because the shape function of the beam element is a cubic, just like the exact solution of the elastic line when there are no distributed loads or effects due to shear; the first part can be arrived at simply by taking into consideration point by point the effects of the distributed load (whether it is concentrated or not).

A similar reasoning is also valid for the displacement, which can be considered the sum of those obtained with the shape function and those obtained from the element in conditions of fixed ends. Each individual user should make sure to understand the approach adopted by the software program in use, taking the simple example of a single element subject to a distributed load and then comparing the values found with those expected in theory. Sargon pre-release 9.0, for example, re-added the effects of distributed loads in terms of stresses but did not re-add them in terms of displacements.

The displacement of points within the finite element was predicted only by the shape function. Note that this leads to a null error in the displacements *at the ends of the element* (where in any case by congruence the correction has to be null). For example, figure 11-15 shows an HE300B cantilever with 3m of clear span, and a uniformly distributed load equal to 5kN/m, with a 1.063mm displacement at the ends (exact value 1.063mm), while in the centre we have 0.3685mm against an exact value of 0.3888mm (an error equal to 5.2%). After release 9.0 Sargon re-adds local displacements as well as local stresses to beam elements for the most common distributed loads.

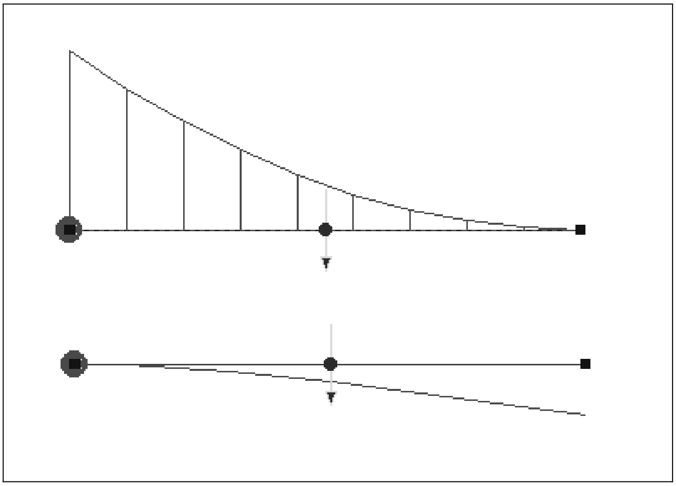


Figure 11-15: Cantilever under a load that is uniformly partitioned (represented by the arrow with the dot) modelled with a single finite element in SARGON: the moment diagram is parabolic, the deflection is cubic (shape function only). The maximum displacement is in one of the nodes and thus the predicted value is exact, even with a "normal" shape function).

11.3.2.7.2 THICK BEAM (CORRECTION FOR THE EFFECTS OF SHEAR)

What has already been discussed regarding beams with no correction is also valid for stresses: if the local effects are added, then the result is exact.

As far as displacements are concerned, the "normal" shape function is not exact even when there are no distributed loads, since, for example, for Φ_3 it takes the form of eq. 11-2 rather than eq. 11-1. However, the displacements at the ends of the element are exact since at the ends the translations of the two shape functions are identical. Instead, the values within the element change, and so it is necessary to bear in mind that the predicted cubic displacement of the "normal" shape function is not exact except at the element's ends (for a comparison, see for example § 12.2.4).

11.3.3 2-NODE TIMOSHENKO ELEMENTS

We have seen that the "slender" 2-node element can be corrected in order to partially simulate the behaviour of a thick beam. There is a 2-node finite element that is expressly intended for this purpose, one that has two unknown displacement functions instead of only one for each bending about each of the main axes.

The two functions are the deflection v and the rotation ψ, modelled independently of each other. Given that we have two available nodes, the shape function is linear, that is, with reference to bending in one of the two planes, and more precisely, bending about axis 3, we set

$$v = v_1\Phi_1+v_2\Phi_2$$

$$\psi= = \psi_1\Phi_1+ \psi_2\Phi_2$$

where

$$\Phi_1=(1-x/L)$$

$$\Phi_2= x/L.$$

Given that the rotation is explicitly modelled, the displacement function w has to be continuous, but its derivative does not have any continuities (that is, it belongs to C_0).

The bending moment is

$$M_3=-EI_3\,\psi'=(EI_3/L)(\ \psi_1 - \psi_2).$$

It is thus assumed to be constant along the element. In contrast, the shear is

$$V_2=GA\beta/\chi_2=GA_2(v'-\psi)=GA_2\left[\frac{v_2-v_1}{L}-\psi_1\Phi_1(x)+\psi_2\Phi_2(x)\right]$$

and it thus varies linearly along the element (note the incongruence with the indefinite equilibrium equations that call for V=M′). At the element's centre, the shear is

$$V_2=GA_2\left[\frac{v_2-v_1}{L}-\frac{\psi_1+\psi_2}{2}\right].$$

The stiffness matrix is obtained by the usual procedure, that is, by writing the elastic strain energy in function of the derivatives of the shape functions, of bending and shear stiffnesses and of nodal displacements. Here we will limit ourselves to expressing only the part of bending relative to local axis 3 (displacement vector $\mathbf{u}=\{v_1, \psi_1, v_2, \psi_2\}^T$):

$$\mathbf{K}=\left|\begin{array}{c|c|c|c}
\frac{GA_2}{L} & \frac{GA_2}{2} & -\frac{GA_2}{L} & \frac{GA_2}{2} \\ \hline
\frac{GA_2}{2} & \frac{EI_3}{L}+\frac{GA_2L}{3} & -\frac{GA_2}{2} & -\frac{EI_3}{L}+\frac{GA_2L}{6} \\ \hline
-\frac{GA_2}{L} & -\frac{GA_2}{2} & \frac{GA_2}{L} & -\frac{GA_2}{2} \\ \hline
\frac{GA_2}{2} & -\frac{EI_3}{L}+\frac{GA_2L}{6} & -\frac{GA_2}{2} & \frac{EI_3}{L}+\frac{GA_2L}{3}
\end{array}\right|$$

The remaining part of the overall stiffness matrix can be obtained by "blowing up" what is described here to the dimension 12x12, bearing in mind that analogous terms are associated to bending about local axis 3, while the axial and torsional components remain unchanged.

The element described here in brief tends to behave absolutely rigidly when the beams are slender (that is, as L increases, the terms of stiffness associated to shear tend to grow excessively with respect to those of bending). This phenomenon, called *locking*, is common to other finite elements in other contexts as well: it can lead to significant errors when this element is used to solve problems of slender beams. Hence the use of this kind of completely integrated element is not advisable when there are slender beams.

We mentioned that there are techniques for integrating the stiffness matrix called *selective* (where the parts of the matrix in bending and in shear are integrated numerically in different ways, with shear being integrated in a reduced way, while bending is integrated exactly) and *reduced* (where integration is done using fewer Gauss points than what is necessary to integrate exactly), aimed at reducing or annulling the effect of locking that was discussed above. Even though such techniques can lead to an improvement in the elements' response, they must be used with extreme care, because they can give rise to problems that are difficult to quantify, such as the onset of strains associated with a null strain energy (spurious modes).

In the particular case of the beam element, the use of these techniques does not lead to the onset of spurious modes and thus the 2-node Timoshenko element is usually calculated with their help (a single Gauss point of integration for the shear component of the matrix, two Gauss points for the bending component). The correct stiffness matrix that results is the following:

$$\mathbf{K} = \left| \begin{array}{c|c|c|c} \frac{GA_2}{L} & \frac{GA_2}{2} & -\frac{GA_2}{L} & \frac{GA_2}{2} \\ \hline \frac{GA_2}{2} & \frac{EI_3}{L}+\frac{GA_2L}{4} & -\frac{GA_2}{2} & -\frac{EI_3}{L}+\frac{GA_2L}{4} \\ \hline -\frac{GA_2}{L} & -\frac{GA_2}{2} & \frac{GA_2}{L} & -\frac{GA_2}{2} \\ \hline \frac{GA_2}{2} & -\frac{EI_3}{L}+\frac{GA_2L}{4} & -\frac{GA_2}{2} & \frac{EI_3}{L}+\frac{GA_2L}{4} \end{array} \right|$$

The 2-node Timoshenko-Mindlin element is only seldom used in practice, at least for linear problems: the 2-node element with the corrections described earlier is preferred.

11.3.4 ELEMENTS WITH THREE OR MORE NODES

In addition to elements with two nodes, elements with three or more nodes have also been theorised and formulated for beam elements, including formulation for both negligible shear strain and for Timoshenko beams. These elements lead to results that are more reliable and precise, especially with regards to Timoshenko elements (by permitting such elements to have linear or parabolic moments). As far as elements that are purely in bending are concerned, the availability of fourth order or higher shape functions makes it possible to have elements that are intrinsically capable of modelling parabolic or cubic bending moments, with an increased advantage in terms of precision and speed of convergence, at least regarding elements whose stresses and displacements are not corrected as discussed earlier.

However, in technical contexts, by far the most widely used element is that with two nodes whose shear strain energy is corrected as described in § 11.3.2.

11.4 SPRINGS

The spring element is used to connect a degree of freedom to a reference ("ground") by means of an elastic constraint of a predetermined stiffness. It should be noted that the spring element does not usually connect two degrees of freedom, but rather, it connects a degree of freedom to the reference, such that a generalised elastic recall force is developed that is equal to ku.

The desired effect is obtained by adding the term relative to the desired stiffness to the term of direct stiffness relative to the degree of freedom in question. For example, if before the application of the spring the equilibrium on the degree of freedom (i) is

$$\sum K_{ij}u_j - p_i = 0,$$

then after the spring with stiffness k is applied we will have

$$ku_i + \sum K_{ij}u_j - p_i = 0$$

such that the original stiffness K_{ii} becomes

$$K_{ii} + k.$$

Not all software is programmed for the use of springs that can connect two generic degrees of freedom (that is, structural springs that are not "grounded"). The stiffness matrix of this element is analogous to that of the truss element, provided that the stiffness k of the spring is substituted in place of EA/L, and provided that we consider the row and the column that correspond to the two connected degrees of freedom u_i and u_j. The result is

$$\mathbf{K}=\left|\begin{array}{c:c} k & -k \\ \hdashline -k & k \end{array}\right|$$

$$\mathbf{u_e}=\{u_{i,}\ u_j\}^T.$$

Even when the spring element is not available, it is often possible to simulate it by means of a truss element (for translational stiffnesses) and an suitably released beam element (for rotational stiffnesses). In the case of translational stiffnesses, we need only connect two nodes with a truss element arranged like the desired spring, and set

$$k=EA/L.$$

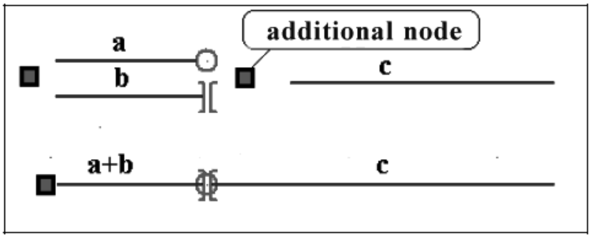


Figure 11-16: Modelling of a rotational spring.

The case of rotational stiffnesses is more complex. For example, in order to connect a beam element "c" to a node with a rotational spring, it is necessary to interpose two small superimposed elements ("a" and "b") with the same connectivity between an added node (at the left end in fig. 11-16) and the end of the element. Element "a" is rigid and serves to guarantee that the shear is continuous: it is released for rotation. Element "b" serves to introduce the desired spring and is released for shear. To this end, element "b", whose length L is, for example, equal to 1/1000 of the original length of the element, will have a moment of inertia I such that

$$K=EI/L$$

where K is the rotational stiffness to be matched.

11.5 PLANE STRESS AND PLANE STRAIN

In problems of plane stress or plane strain, only two displacement functions are unknown. Given plane XY where the structure is defined, the unknown functions are u(X, Y) and v(X, Y). Thus, at every node "i" the two translations u_i and v_i are generally used as unknown nodal displacements (although there are situations when this is not the case, as we shall see later). If we look at the formulation of the individual element, we see that it is defined in a local system (x, y) or (1, 2) and that the nodal displacements in direction x and y are unknown. Because none of the other components of displacement exert any influence, they must be (either automatically or explicitly) eliminated or constrained.

In some finite elements, rotations about the axis normal to the plane θ_i of the element are also used, and thus this degree of freedom is not constrained (see, for example, [Taylor and Simo 1985]). *In this case this degree of freedom should not be constrained, because if it is the results obtained are incorrect.*

For a complete definition of the element it is necessary to provide the characteristics of the material and – in plane stress – its thickness.

11.5.1 3-NODE TRIANGULAR ELEMENT (CONSTANT STRAIN TRIANGLE, CST)

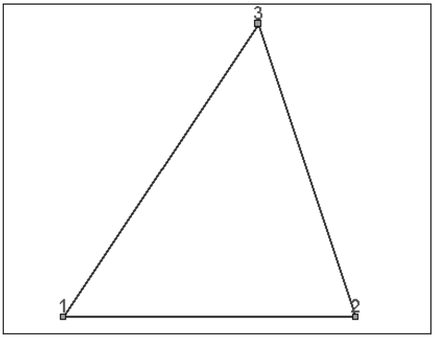


Figure 11-17: Constant strain triangle (CST) element.

The CST element, or constant strain triangle, is the most simple but also the least precise. Each of the two unknown functions u(x, y) and v(x, y) is modelled by means of the following polynomial approximations:

$$u(x) = ax+by+c$$

$$v(x)= dx+ey+f,$$

that is, by means of a linear displacement function. Setting boundary conditions on the nodal displacements u_i and v_i (i=1, 2, 3)[2] leads to obtaining the shape functions, such that u(x) and v(x) can be written in the following canonical way:

$$u(x)=u_1\Phi_1+ u_2\Phi_2+ u_3\Phi_3$$

$$v(x)=v_1\Phi_1+ v_2\Phi_2+ v_3\Phi_3$$

where

[2] See chap. 4, equation 4-3: here a, b, c, d, … are the a_1, a_2, a_3, a_4, etc. of eq. 4-3.

$$\Gamma_1 \equiv \Phi_1 = \frac{1}{2A}\left[(x_2 y_3 - x_3 y_2) + (y_2 - y_3)x + (x_3 - x_2)y\right]$$

$$\Gamma_2 \equiv \Phi_2 = \frac{1}{2A}\left[(x_3 y_1 - x_1 y_3) + (y_3 - y_1)x + (x_1 - x_3)y\right]$$

$$\Gamma_3 \equiv \Phi_3 = \frac{1}{2A}\left[(x_1 y_2 - x_2 y_1) + (y_1 - y_2)x + (x_2 - x_1)y\right].$$ [3]

Since the strains are obtained by deriving the displacement field, it is easy to see that in correspondence to the approximation introduced there is a strain field that is constant (hence the element's name, constant strain triangle or CST).

In matrix terms we can write

$$\mathbf{U} = \mathbf{\Phi u}$$

having set $\mathbf{U} = \{\mathrm{u}, \mathrm{v}\}^{\mathrm{T}}$, $\mathbf{u} = \{\mathrm{u}_1, \mathrm{v}_1, \mathrm{u}_2, \mathrm{v}_2, \mathrm{u}_3, \mathrm{v}_3\}^{\mathrm{T}}$ and

$$\mathbf{\Phi} = \left|\begin{array}{cc:cc:cc} \Phi_1 & 0 & \Phi_2 & 0 & \Phi_3 & 0 \\ \hdashline 0 & \Phi_1 & 0 & \Phi_2 & 0 & \Phi_3 \end{array}\right|$$

The strain vector $\boldsymbol{\varepsilon}$ can be determined in the following way (see also chap. 3):

$$\boldsymbol{\varepsilon} = \mathbf{CU} = \left|\begin{array}{cc} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{array}\right| \left|\begin{array}{cc:cc:cc} \Phi_1 & 0 & \Phi_2 & 0 & \Phi_3 & 0 \\ \hdashline 0 & \Phi_1 & 0 & \Phi_2 & 0 & \Phi_3 \end{array}\right| \mathbf{u}.$$

that is,

$$\boldsymbol{\varepsilon} = \mathbf{Bu} = \frac{1}{2A}\left|\begin{array}{cc:cc:cc} y_{23} & 0 & y_{31} & 0 & y_{12} & 0 \\ \hdashline 0 & x_{32} & 0 & x_{13} & 0 & x_{21} \\ \hdashline x_{32} & y_{23} & x_{13} & y_{31} & x_{21} & y_{12} \end{array}\right| \mathbf{u}.$$

The stiffness matrix **K** is obtained by the following integral

$$\mathbf{K} = \int_A \mathbf{B}^T \mathbf{DB}\, dA.$$

where **D** is the matrix that relates the stresses and the strains (different for plane stress and plane strain, see chap. 3). Since the terms concerned are constant, the previous expression becomes

$$\mathbf{K} = \mathrm{A}\mathbf{B}^{\mathbf{T}}\mathbf{DB}.$$

The CST element converges rather slowly: the stress within it is constant and thus it is necessary to use a high number of elements in order to obtain a

[3] The symbols Γ_1, Γ_2, Γ_3 were introduced because they will be useful in what follows.

sufficient approximation. However, the element is compatible and complete, since it is capable of correctly representing both rigid motions and constant strains.

11.5.2 THE 6-NODE TRIANGULAR ELEMENT (LINEAR STRAIN TRIANGLE)

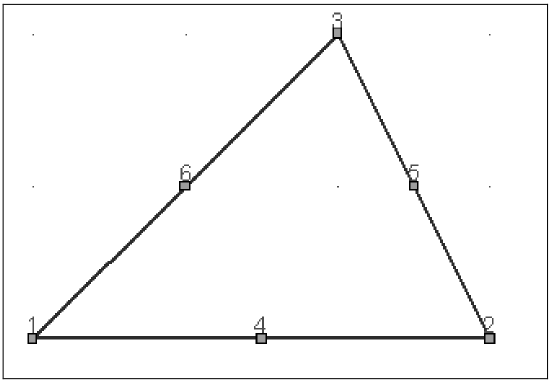


Figure 11-18: Linear strain triangle (LST) element.

In this element, which, like the CST element, is triangular with straight line sides, there are six nodes instead of three. These additional nodes are placed at the mid-points of the three sides, and are numbered starting from 4 for side 1-2 up to 6 for side 3-1. The basis expressions for the interpolation field are:

$$u(x) = ax^2+by^2+cxy+dx+ey+f$$

$$v(x)= gx^2+hy^2+lxy+mx+ny+p.$$

By making the displacements in the nodes congruent with the nodal displacements, we obtain the following matrix equation for the displacements:

$$\mathbf{U}=\mathbf{\Phi u},$$

having set $\mathbf{U} = \{u, v\}^T$, $\mathbf{u}=\{u_1, v_1, u_2, v_2, u_3, v_3, u_4, v_4, u_5, v_5, u_6, v_6\}^T$ and

$$\mathbf{\Phi} = \left|\begin{array}{cc:cc:cc:cc:cc:cc} \Phi_1 & 0 & \Phi_2 & 0 & \Phi_3 & 0 & \Phi_4 & 0 & \Phi_5 & 0 & \Phi_6 & 0 \\ \hdashline 0 & \Phi_1 & 0 & \Phi_2 & 0 & \Phi_3 & 0 & \Phi_4 & 0 & \Phi_5 & 0 & \Phi_6 \end{array}\right|$$

.

The shape functions are expressed as:

$$\Phi_1=\Gamma_1(2\Gamma_1\text{-}1)$$

$$\Phi_2=\Gamma_2(2\Gamma_2\text{-}1)$$

$$\Phi_3=\Gamma_3(2\Gamma_3\text{-}1)$$

$$\Phi_4=4\Gamma_1\,\Gamma_2$$

$$\Phi_5=4\Gamma_2\,\Gamma_3$$

$$\Phi_6=4\Gamma_3\,\Gamma_1$$

with the Γ_i as defined in the previous section (practically speaking, these are the shape functions of the CST element).

The next step towards obtaining the stiffness matrix follows the general rules already discussed for the CST element, with the difference that the integral is not immediate. However, the integration can in any case be made in closed form by using the following formula (where "m!" is m factorial):

$$\int_A \Gamma_1^{\,k}\Gamma_2^{\,l}\Gamma_3^{\,m}dA = 2A\frac{k!l!m!}{(2+k+l+m)!}$$

This element has a quadratic displacement function and linear stress and strain functions (hence its name, linear strain triangle, LST). From the point of view of convergence, its behaviour is clearly better than the CST, as can be seen from the analysis of the data in table 11-4, adapted from [Cook, Malkus, Plesha 2001] and checked again using SARGON for cases B and C shown in fig. 11-20. The problem is the classic case of a cantilever with a concentrated load at its free end.

An examination of table 11-4 leads us to conclude that, for almost equally precise results, the CST elements require a number of degrees of freedom that is not simply double, but 3.5 times higher than that of LST elements.

In general we can say that the CST element behaves rather poorly in situations where it is necessary to simulate bending, unless a high number of elements is used in the thickness.

It is certainly helpful to consider the unsmoothed maps of strain σ_x, shown in fig. 11-21 for cases B and C, that is, in the two cases where CST elements were used. As we can see, smoothing leads to significant improvement, even if – as we have said – it can be used only when the interelement jumps are modest.

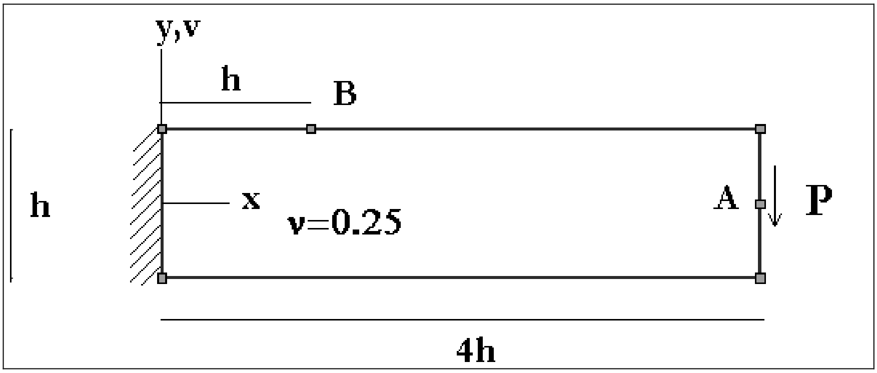


Figure 11-19: Test problem of plane stress: E=200kN/mm^2, t=10mm, P=40kN, h=12cm. [Carr 1967] cited in [Brebbia and Connor 1978] and in [Cook, Malkus, Plesha 1989] in a dimensionless form. "Exact" v_A = 5.3374mm, "exact" σ_B = 600 N/mm^2. The problem provides for a parabolic shear distribution at the end; here for brevity's sake the force is concentrated in a node (which leads to a limit value for the ratio between the calculated solution and the "exact" one equal to 1.008 instead of 1.000 for displacement).

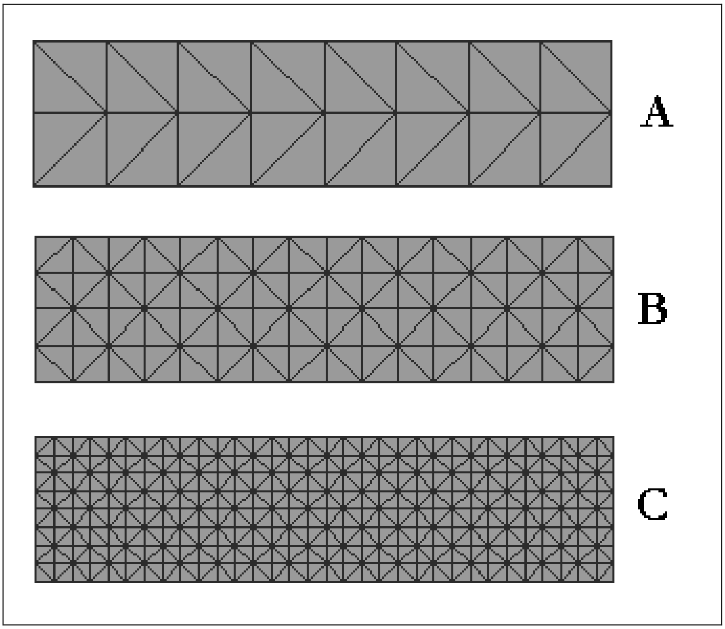


Figure 11-20: Study cases A, B and C of the problem shown in fig. 11-19.

Mesh	Elements	DOF	v_A	σ_{xB}
A	32LST	160	0.998	0.986
B	128CST	160	0.859	0.854
C	512CST	576	0.961	0.956

Table 11-4: Ratios of calculated to exact solutions for cases A, B and C shown in fig. 11-20, with comparisons of the number of elements and degrees of freedom. Strain σ_{xB} is calculated using the average of the elementary values.

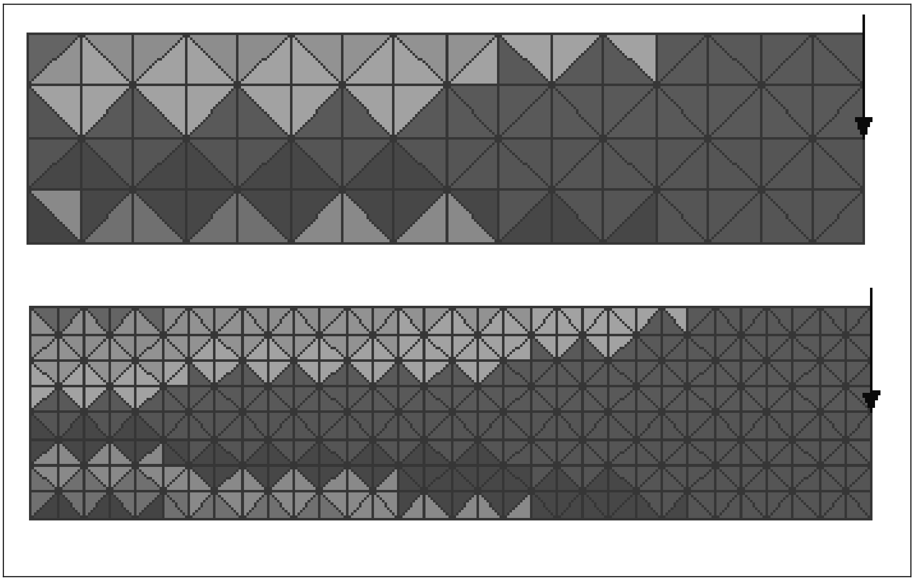


Figure 11-21: Unsmoothed maps of strain σ_x for CST elements in cases B and C (same scale –800-800 N/mm^2). The strain is constant on each element.

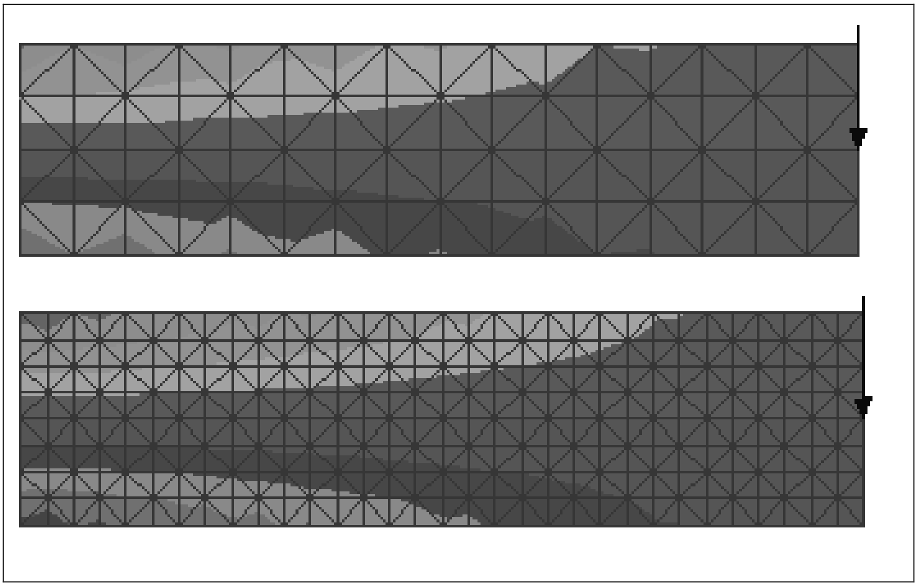


Figure 11-22: Smoothed maps of strain σ_x for CST elements in cases B and C (same scale –800-800 N/mm^2). The strain now varies linearly on each element due to the interpolation assumed beginning with the average strains at the nodes.

11.5.3 4-NODE ELEMENTS

11.5.3.1 BILINEAR ISOPARAMETRIC ELEMENT

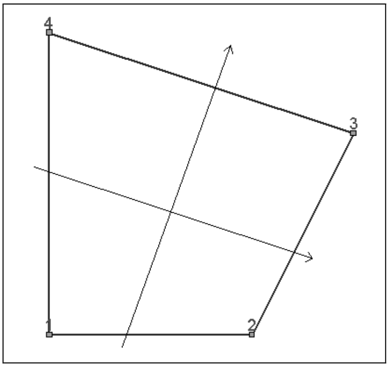


Figure 11-23: Planar bilinear element.

This element is the first of a family of very important and widely used elements: plane isoparametric elements with *n* nodes, with *n* variable from 4 to 9. The isoparametric formulation makes it possible to have irregular forms such as quadrilaterals with non-right angles or curved sides (where there are at least three nodes to a side). The four-node isoparametric element uses an interpolating function of the kind:

$$u(x, y)= axy+bx+cy+d$$

$$v(x, y)= exy+fx+gy+h$$

in which, as we can see, an incomplete second-order polynomial is used. Usually this element and those related to it are formulated by making use of the *master element* discussed in §4.6, such that the integration takes place in a system of normalised dimensionless coordinates.

Observing the interpolating polynomials we can see that the strains – and thus the stresses – depend on the variables in this way:

$$\varepsilon_x \rightarrow ay+b$$

$$\varepsilon_y \rightarrow ex+g$$

$$\gamma_{xy} \rightarrow ax+c+ey+f,$$

and thus we see that the normal strains depend linearly but incompletely on (x, y). In particular, we can see that, given a certain direction, x or y, the dependence of the stress acting in this direction is only constant in that direction, while the dependence in the orthogonal direction is linear.

The shape functions (in the reference system using ξ and η with ξ and η variable between –1 and 1) are the following:

$$\Phi_1=\tfrac{1}{4}(1-\xi)(1-\eta)$$

$$\Phi_2=\tfrac{1}{4}(1+\xi)(1-\eta)$$

$$\Phi_3=\tfrac{1}{4}(1+\xi)(1+\eta)$$

$$\Phi_4=\tfrac{1}{4}(1-\xi)(1+\eta)$$

and are equivalent to the following interpolating expressions for x and y (the element's geometry)

$$x(\xi, \eta)=x_1\Phi_1+ x_2\Phi_2+ x_3\Phi_3+ x_4\Phi_4$$

$$y(\xi, \eta)=y_1\Phi_1+ y_2\Phi_2+ y_3\Phi_3+ y_4\Phi_4$$

and those that follow for the unknown displacements

$$u(\xi, \eta)=u_1\Phi_1+ u_2\Phi_2+ u_3\Phi_3+ u_4\Phi_4$$

$$v(\xi, \eta)=v_1\Phi_1+ v_2\Phi_2+ v_3\Phi_3+ v_4\Phi_4.$$

The points of the element, whose shape can even be distorted as long as it is planar, correspond to the set of points that lie in the square comprised between (-1, 1) for both ξ and η. However, the distortion should not be taken as a licence to make degenerate elements. The angles have to remain close to 90° and the sides should all be more or less the same length. If this is not the case then the precision is diminished.

We can see that if F(x, y) is a function of x and y, and x and y depend on two other variables (ξ, η), then we can write

$$\frac{\partial F}{\partial x}=\frac{\partial F}{\partial \xi}\frac{\partial \xi}{\partial x}+\frac{\partial F}{\partial \eta}\frac{\partial \eta}{\partial x}$$

$$\frac{\partial F}{\partial y}=\frac{\partial F}{\partial \xi}\frac{\partial \xi}{\partial y}+\frac{\partial F}{\partial \eta}\frac{\partial \eta}{\partial y}$$

as well as the inverse of these

$$\frac{\partial F}{\partial \xi}=\frac{\partial F}{\partial x}\frac{\partial x}{\partial \xi}+\frac{\partial F}{\partial y}\frac{\partial y}{\partial \xi}$$

$$\frac{\partial F}{\partial \eta}=\frac{\partial F}{\partial x}\frac{\partial x}{\partial \eta}+\frac{\partial F}{\partial y}\frac{\partial y}{\partial \eta},$$

which are more convenient because we know the functions x(ξ, η) and y(ξ, η) and not the inverses.

The last two equations can be rewritten in matrix form in the following way, which serves to define the Jacobian matrix **J**:

$$\left|\begin{matrix} F,_{\xi} \\ F,_{\eta} \end{matrix}\right| = \left|\begin{matrix} x,_{\xi} & y,_{\xi} \\ x,_{\eta} & y,_{\eta} \end{matrix}\right| \left|\begin{matrix} F,_{x} \\ F,_{y} \end{matrix}\right| \equiv \mathbf{J} \left|\begin{matrix} F,_{x} \\ F,_{y} \end{matrix}\right|.$$

The previous relationships can be inverted, giving rise to the following:

$$\left|\begin{matrix} F,_{x} \\ F,_{y} \end{matrix}\right| = \mathbf{J}^{-1} \left|\begin{matrix} F,_{\xi} \\ F,_{\eta} \end{matrix}\right| = \frac{1}{J} \left|\begin{matrix} y,_{\eta} & -y,_{\xi} \\ -x,_{\eta} & x,_{\xi} \end{matrix}\right| \left|\begin{matrix} F,_{\xi} \\ F,_{\eta} \end{matrix}\right| \equiv \left|\begin{matrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{matrix}\right| \left|\begin{matrix} F,_{\xi} \\ F,_{\eta} \end{matrix}\right|$$

where the determinant of the Jacobian, J, is expressed as

$$J = x,_{\xi}\, y,_{\eta} - y,_{\xi}\, x,_{\eta}.$$

When these laws of transformation are known, then it is easy to calculate the strains starting from the displacement field. For example:

$$\varepsilon_x = u,_x = \sum_1^4 \Phi_{i,x} u_i = \frac{1}{J}\sum_1^4 (y,_{\eta}\, \Phi_{i,\xi} - y,_{\xi}\, \Phi_{i,\eta}) u_i\ ,$$

where

$$x,_{\xi} = \sum_1^4 \Phi_{i,\xi} x_i$$

$$x,_{\eta} = \sum_1^4 \Phi_{i,\eta} x_i$$

$$y,_{\xi} = \sum_1^4 \Phi_{i,\xi} y_i$$

$$y,_{\eta}=\sum_{1}^{4}\Phi_{i,\eta}y_i\ .$$

$$\begin{array}{ll}
\Phi_{1,\xi}=-\tfrac{1}{4}(1-\eta) & \Phi_{1,\eta}=-\tfrac{1}{4}(1-\xi) \\
\Phi_{2,\xi}=\tfrac{1}{4}(1-\eta) & \Phi_{2,\eta}=-\tfrac{1}{4}(1+\xi) \\
\Phi_{3,\xi}=\tfrac{1}{4}(1+\eta) & \Phi_{3,\eta}=\tfrac{1}{4}(1+\xi) \\
\Phi_{4,\xi}=-\tfrac{1}{4}(1+\eta) & \Phi_{4,\eta}=\tfrac{1}{4}(1-\xi)
\end{array}$$

In a few steps we obtain (see, for example [Cook, Malkus, Plesha 1989])

$$\boldsymbol{\varepsilon}=\mathbf{Bu}.$$

Having set $\boldsymbol{\varepsilon}=\{\ \varepsilon_x\ ,\ \varepsilon_y\ ,\ \gamma_{xy}\}^T$ $\mathbf{u}=\{u_1,\ v_1,\ \ldots,\ u_4,\ v_4\}^T$ and

$$\mathbf{B}=\begin{vmatrix}1&0&0&0\\0&0&0&1\\0&1&1&0\end{vmatrix}\begin{vmatrix}G_{11}&G_{12}&0&0\\G_{21}&G_{22}&0&0\\0&0&G_{11}&G_{12}\\0&0&G_{21}&G_{22}\end{vmatrix}\begin{vmatrix}\Phi_{1,\xi}&0&\Phi_{2,\xi}&0&\Phi_{3,\xi}&0&\Phi_{4,\xi}&0\\\Phi_{1,\eta}&0&\Phi_{2,\eta}&0&\Phi_{3,\eta}&0&\Phi_{4,\eta}&0\\0&\Phi_{1,\xi}&0&\Phi_{2,\xi}&0&\Phi_{3,\xi}&0&\Phi_{4,\xi}\\0&\Phi_{1,\eta}&0&\Phi_{2,\eta}&0&\Phi_{3,\eta}&0&\Phi_{4,\eta}\end{vmatrix}$$

the stiffness matrix is obtained from

$$\mathbf{K}=\int\mathbf{B}^T\mathbf{D}\mathbf{B}Jtd\xi d\eta$$

where t is the thickness (=1 in plane strain) and J the determinant of the Jacobian of transformation (it can be proven that dA is transformed into $Jd\xi d\eta$).

In the bilinear isoparametric element, the strain and thus the stress have an expression that varies linearly (although in an incomplete way: it is missing one term to complete the linear form). It is very important to determine the appropriate order of integration for the element, in order to determine the stiffness matrix numerically (it is not generally possible to integrate the expression of stiffness matrix **K** in closed form). Usually a scheme of integration is adopted that provides for 2x2 Gauss points, with the exceptions discussed below.

In order to improve the performance of this element, *which proves to be visibly stiffer than necessary for discretizations that are not dense* (see also chapter 12), first of all techniques of reduced integration, then techniques of selective integration, were proposed in the attempt to reduce the phenomenon of locking related to excessive stiffness (see, for example, [MacNeal 1994]). While the techniques of reduced integration (one Gauss point instead of four) have been found to lead to significant problems related to the onset of spurious modes (that is, the possibility that the underintegrated element will deform when the strain energy is null, such as, for example, the well-known “hourglass” spurious mode), techniques of selective integration, that is, those aimed at underintegrating only the shear part of the strain, while conserving the 2x2 integration for the bending part, have been found to not lead to spurious modes and to improve the solution considerably, although they do involve other problems, such as those sometimes related to the formal proof of their convergence.

The use of techniques for selective or reduced integration when there are no formal proofs of convergence (constituting what have been called "variational crimes") has been criticised by various authorities on the subject, and in particular by Bathe, who wrote:

> The reason for recommending the [full] numerical integration orders ... is that the *reliability of the finite element procedures is of utmost concern*[4] ... and if an integration order lower than the "full" order is used (for a displacement-based or a mixed formulation) the analysis is in general unreliable [1996: 469-470].
>
> For these reasons any element with a spurious zero energy mode should not be used in engineering practice, in linear or in nonlinear analysis, and we therefore do not discuss such elements in this book. However, we should mention that to prevent the deleterious effects of spurious modes, significant research efforts have been conducted to control their behaviour [1996: 473].
>
> The key question as to whether a reduced and/or selectively integrated element can be recommended for practical use is: Has the element formulation (using the specific integration procedure) been sufficiently tested and analyzed for its stability and convergence? If tractable, a mathematical stability and convergence analysis is of course most desirable [1996: 476].

It is interesting to note that the opinions of various authors differ greatly on this point. We have just seen Bathe's concerns about the use of elements whose convergence has not been formally demonstrated. Others, such as MacNeal [1994], for example, propose basing their use on the results effectively obtained by various elements rather than on the formal proof of their convergence, maintaining that the patch test is more important and that some elements which were expected to perform well in theory (proof of convergence, conformance to variational principles, etc.) have instead been found to perform poorly in practice.

It is difficult to say with certainty that one approach is better than another. However, it is worthwhile noting that elements for which we have no proof of convergence, and which do not conform to any known principle of variation, appear to be like a kind of time bomb, since only in some cases, and not in all, their convergence has been or can be proven.

Personally, we prefer Bathe's approach to that of MacNeal. Bathe recommends the full 2x2 integration for four-node isoparametric elements, and thus this was the element implemented in SARGON. It can be easily seen that the effects of locking disappear if the discretization is significantly increased, that is, if finite elements that are regular in shape and sufficiently small are made (see, for example, table 11-6). Further, it doesn't seem really necessary to be able to

[4] The italics are by Prof. Bathe

obtain results that are close to exact ones with meshes that are extremely coarse, if this leads to even a minimum of unpredictability regarding the element used.

Because of this, and also in light of the simplicity with which the four-node elements can be implemented with selective integration, and the opportunity of having initial estimates that are generally closer to the final results even with meshes that are rather coarse, it was decided that SARGON would also permit the use of four-node elements with selective integration.

In defining the element, it is very important to avoid angles that are too far from right, or even tend to be flat, in order to avoid that the determinant of the Jacobian is null. In the case of rectangular elements as well, it is a good idea to avoid having sides that are excessively different in length, keeping the elements as close as possible to square.

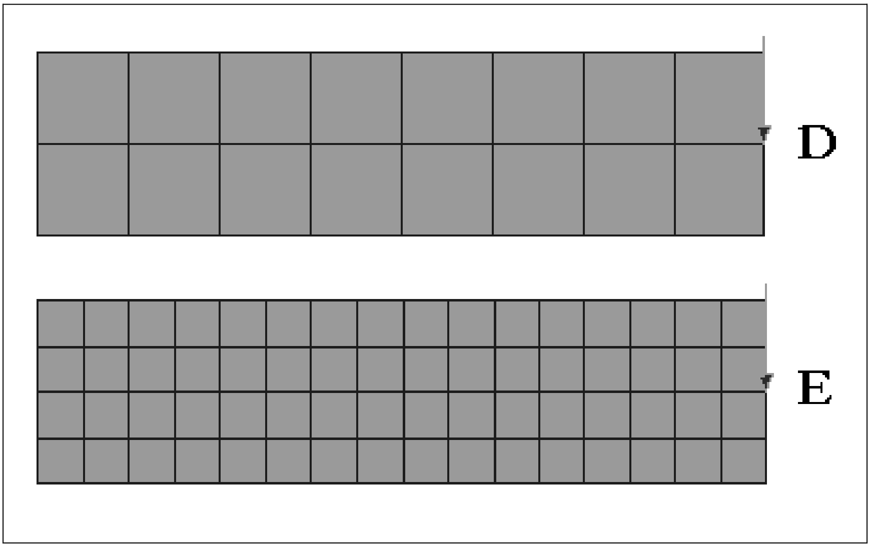


Figure 11-24: Test cases D and E of bilinear elements.

As far as precision is concerned, table 11-5 shows the results obtained for the problem shown in fig. 11-19 with the meshes shown in fig. 11-24, comparing results obtained with full integration QUAD4 and with selective integration QUAD4SRI for test cases D and E.

As we can see, the results are rather good even when a relatively coarse mesh (only 160 degrees of freedom) is used. Note however that in the problem in question there are no gaps, discontinuities or other factors that can generate concentrated stresses.

As in the case of CST elements, the smoothed and unsmoothed stress maps for σ_x allow us to compare the results obtained with various kinds of mesh.

Mesh	Elements	DOF	v_A	σ_{xB}
D	16QUAD4	48	0.890	0.928
E	64QUAD4	160	0.970	0.987
D	16QUAD4SRI	48	0.974	1.022
E	64QUAD4SRI	160	0.995	1.013

Table 11-5: Results for bilinear elements (QUAD4 and QUAD4SRI), ratios of the calculated to the "exact" solutions (SARGON).

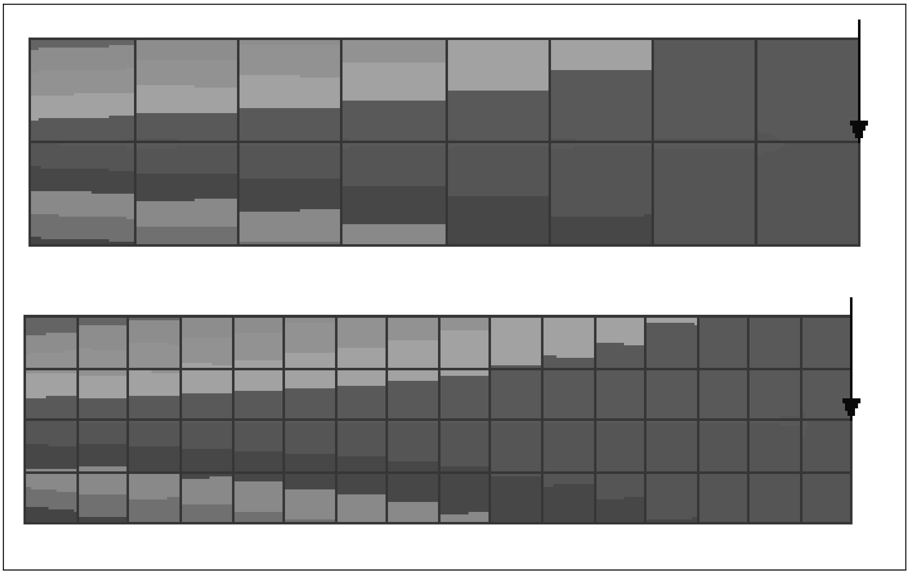


Figure 11-25: Unsmoothed stress maps for σ_x for test cases D and E for QUAD4 elements (same scale –800-800 N/mm^2). The stress varies linearly over each element.

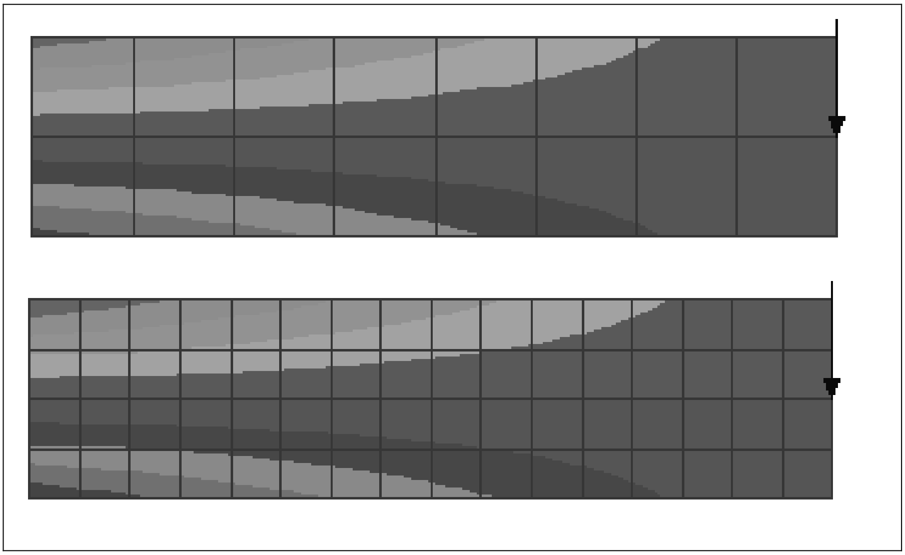


Figure 11-26: Smoothed stress maps for σ_x in test cases B and C for QUAD4 elements (same scale –800-800 N/mm^2).

11.5.3.2 TAYLOR-SIMO ELEMENTS (TAYL4)

There are other four-node elements that are used to study problems of plane stress and plane strain. One such element is that of Taylor-Simo [1985] which is implemented in the well-known software programs Sap80®, Sap90® and Sap2000®, and which is special in that, in addition to two translations per node, it also makes use of *the rotation about an axis normal to the plane of the element (drilling DOF).*[5] *Thus for this particular element normal rotation is not free, and thus it must not be constrained.* Adopting the same mesh used for test cases D and E, we have respectively cases F and G, the results for which are shown in table 11-6.

Mesh	Elements	DOF	v_A	σ_{xB}
D	16TAYL4	72	0.984	1.034
E	32TAYL4	240	0.997	1.016

Table 11-6: Results for the Taylor-Simo elements (TAYL4, [Taylor & Simo 1985]), and ratios of the calculated to the "exact" solutions.

[5] There are also triangular elements that use *drilling DOF.*

As we can see, the performance of this element is already excellent with a mesh that is not very dense (but note that the elements are still not distorted).

11.5.3.3 Non-compatible elements QM6WI

Similar excellent results are achieved with the use of higher-order elements (with a larger number of parameters) which are then reduced to only four nodes by means of a condensation mechanism (see §11.3.2.5). The resulting element is not compatible (because the displacements on the boundaries also depend on parameters that have been eliminated, which are different for the two contiguous elements) and thus it loses the characteristics of convergence that are proper to compatible elements. However, it has been found to perform very well for meshes that are rather coarse.

Practically speaking, to the four parameters that are associated to the nodal displacements are added two "nodeless" parameters, that is, parameters that are not associated to nodes, so as to complete the terms associated with a complete second-order polynomial (as happens for the LST element). For example, for displacement u we set:

$$\mathrm{u}=\sum_{1}^{4}\phi_i u_i+\left(1-\xi^2\right)a_1+\left(1-\eta^2\right)a_2\ .$$

As a consequence, the four new parameters introduced (two for u, two for v, a_1, a_2, a_3, a_4) are eliminated by condensation, and thus the matrix remains (8x8) instead of becoming (12x12). The four-node element formulated in this way, called Q6, can pose significant problems if it is not a rectangle or a parallelogram. In this case it will not pass a patch test.

Various methods have been proposed to overcome this problem, first using techniques of modified integration (element QM6), and then with different techniques by Wilson and Ibrahimbegovic [1990]. In essence, to the part of the stiffness matrix associated with shear strain energy is added a correction matrix, so that the effect of the incompatible modes is rendered null when a constant strain is set (which is precisely what takes place with the patch test).

The QM6WI element is considered one of the best four-node elements available, since it is also extremely precise regarding the stress field, thanks to a technique described by Wilson and Ibrahimbegovic (and which can also be extended to other contexts as well). Practically speaking, the stresses are not calculated with the standard procedure of stress recovery, but by means of techniques based on setting the internal equilibrium of the element.

This element was implemented in SARGON and, in the standard case used for our comparison, led to the results shown in table 11-7.

Mesh	Elements	DOF	v_A	σ_{xB}
D	16QM6WI	48	0.990	1.001
E	32 QM6WI	160	0.999	1.000

Table 11-7: Results for the non-compatible four-node Wilson-Ibrahimbegovic elements (QM6WI), and ratios of the calculated to the "exact" solutions (SARGON).

11.5.4 FOUR-SIDED ISOPARAMETRIC ELEMENTS WITH 8 NODES (QUAD8) AND 9 NODES (QUAD9)

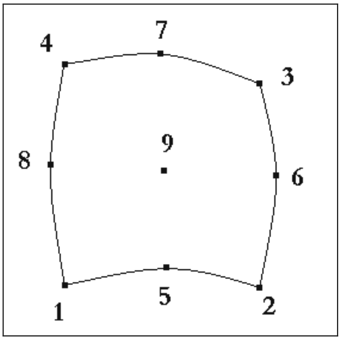


Figure 11-27: Plane isoparametric element with 8 or 9 nodes.

With respect to the previous element, the plane isoparametric element with eight nodes adds one node per side (at the midpoint of the side in the projected plane ξ–η). The interpolating formula for the displacements are

$$u(x)= ax^2y+bxy^2+cx^2+dy^2+exy+fx+gy+h$$

$$v(x)= ix^2y+lxy^2+mx^2+ny^2+oxy+px+qy+r$$

that is, incomplete third-degree polynomials. The shape functions that make it possible to write the usual formulas

$$u(\xi, \eta)=u_1\Phi_1+ u_2\Phi_2+ u_3\Phi_3+ u_4\Phi_4 +u_5\Phi_5+ u_6\Phi_6+ u_7\Phi_7+ u_8\Phi_8$$

$$v(\xi, \eta)=v_1\Phi_1+ v_2\Phi_2+ v_3\Phi_3+ v_4\Phi_4 +v_5\Phi_5+ v_6\Phi_6+ v_7\Phi_7+ v_8\Phi_8$$

are the following:

$$\Phi_1=¼(1- \xi)(1- \eta)-½\Phi_5-½\Phi_8 \qquad \Phi_5=½(1- \xi^2)(1- \eta)$$

$$\Phi_2=¼(1+ \xi)(1- \eta)-½\Phi_5-½\Phi_6 \qquad \Phi_6=½(1+ \xi)(1- \eta^2)$$

$$\Phi_3=¼(1+ \xi)(1+ \eta)-½\Phi_6-½\Phi_7 \qquad \Phi_7=½(1- \xi^2)(1+ \eta)$$

$$\Phi_4=¼(1- \xi)(1+\eta)-½\Phi_7-½\Phi_8 \qquad \Phi_8=½(1- \xi)(1- \eta^2).$$

This formulation allows a strain ε_x to vary parabolically in y and linearly in x, a strain ε_y to vary parabolically in x and linearly in y, and a shear γ_{xy} to vary according to a complete second-order polynomial in x and y.

Thanks to there being three nodes per side, this element can have curved (parabolic) sides while respecting the compatibility of the displacements as well as the congruence of the continuum. Of course the element can be used with straight sides as well. If an ulterior node is added to this element in a position that corresponds to the origin of the dimensionless plane (ξ, η), the resulting element performs even better than the previous one, since the approximating polynomials become

$$u(x)= ax^2y^2+bx^2y+cxy^2+dx^2+ey^2+fxy+gx+hy+i$$

$$v(x)= lx^2y^2+mx^2y+nxy^2+ox^2+py^2+qxy+rx+sy+t,$$

making it possible to achieve an especially improved behaviour when the element is distorted, with curved sides or with nodes that are not at the midpoints of the sides. It should be noted that the more distorted and irregular the elements are, the less accurate they become, a circumstance that should remind us to make the meshes as regular as possible. The additional shape function is called a *bubble function*, because it recalls the shape of a bubble blown on the element's four sides. The ninth shape function is

$$\Phi_9=(1-\xi^2)(1-\eta^2)$$

and the other shape functions (those of the eight-node element) have to be corrected by adding to each Φ_i with i=1,2,3,4, the quantity

$$-\tfrac{1}{4}\Phi_9$$

and to each Φ_i with i=5,6,7,8, the quantity

$$-\tfrac{1}{2}\Phi_9.$$

The eight- and nine-node isoparametric elements perform quite well and are the elements of choice for the analysis of bidimensional continua.

In what follows we will cite the integration order suggested by Bathe [1996] for this kind of element (sometimes the integration order can be decided on by the user):

8-node rectangular or distorted element	3x3 Gauss points
9-node rectangular or distorted element	3x3 Gauss points

Table 11-8 shows the results obtained with these higher-order elements for the same problem already examined with QUAD4 and CST elements.

Mesh	Elements	DOF	v_A	σ_{xB}
D	16QUAD8	128	0.999	0.999
E	64QUAD8	304	1.003	1.000
D	16QUAD9	256	1.001	1.001
E	64QUAD9	432	1.004	1.000

Table 11-8: Results QUAD8 and QUAD9 elements, and ratios of the calculated to the "exact" solutions (courtesy of engineer Fulvio Trudi; calculations performed with ABAQUS, version 6.2-7). The fact that the ratios of the displacement of the QUAD8 elements, which are compatible, are greater than 1 is because the load was concentrated in a single node rather than distributed along a parabolic arc.

11.5.5 Transition Elements

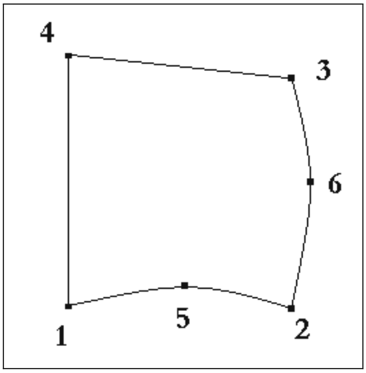


Figure 11-28: Isoparametric element with 6 nodes.

The family of isoparametric elements is also very important because it is possible to generate an entire series of finite elements that have a number of nodes ranging from 4 to 9, simply by suitably modifying the shape functions.

For example, figure 11-28 shows an element that has two parabolic sides and two straight sides, with a total of 6 nodes. Thanks to this kind of element – also called a transition element – it is possible to connect areas of the mesh where there are lower-order elements to areas where there are higher-order elements, without sacrificing the compatibility of the interelement displacements.

The shape functions are generated starting from those of the nine-node element, by simply eliminating from all the expressions the shape functions relative to nodes that are not present. In the example shown in fig. 11-28, we would eliminate all the terms that contain Φ_7, Φ_8 and Φ_9:

$$\Phi_1=\tfrac{1}{4}(1-\xi)(1-\eta)-\tfrac{1}{2}\Phi_5 \qquad \Phi_4=\tfrac{1}{4}(1-\xi)(1+\eta)$$
$$\Phi_2=\tfrac{1}{4}(1+\xi)(1-\eta)-\tfrac{1}{2}\Phi_5-\tfrac{1}{2}\Phi_6 \qquad \Phi_5=\tfrac{1}{2}(1-\xi^2)(1-\eta)$$
$$\Phi_3=\tfrac{1}{4}(1+\xi)(1+\eta)-\tfrac{1}{2}\Phi_6 \qquad \Phi_6=\tfrac{1}{2}(1+\xi)(1-\eta^2).$$

11.6 MEMBRANES

The membrane element is simply an element of plane stress that is placed in a three-dimensional context. The element maintains its characteristics of stiffness, which are "blown up" to fill all the pertinent areas of the stiffness matrix in the three-dimensional context into which the element has been introduced.

When considering an element of this type, it is necessary to bear two things in mind.

First, this element does not present any stiffnesses that limit the translation of its nodes in the direction normal to the plane of the element; thus, if there are no other elements attached to a given node of a membrane element, the translation normal to the plane of the element is free. If the plane of the element is parallel to one of the global planes (xy, yz, zx) then the node is the site of a pseudo-mechanism. Second, this element does not in any way limit the rotation of its nodes, which are free of stiffnesses due to the element itself (although there can be other stiffnesses due to other elements connected to the node).

"Membrane elements" therefore range from the three-node triangular element up to the nine-node four-sided isoparametric element. These elements are generally used in cases of buildings to model shear walls or wind bracing.

11.7 PLATE-SHELL

11.7.1 OVERVIEW

Plate-shell elements are used to model the behaviour of thick or thin plate and shells. Here we will briefly describe some of the flat elements that are suitable for modelling problems involving plates or shells. The membrane behaviour is decoupled from bending behaviour: plate elements generally comprise the behaviour typical of space membranes, so that a plate-shell element is actually always a membrane element (if this were not the case, it would be necessary to superimpose two elements, one purely membrane and the other purely bending). In what follows we will refer only to the elements' behaviour in bending, since the membrane part has already been discussed (cf. § 11.5).

From a practical point of view, a plate element is defined when the material it is made of and its thickness are defined (the thickness can be constant or variable, depending on the element's formulation; it is usually constant). Bearing in mind that the bending behaviour is decoupled from the membrane behaviour, it is also possible to determine the thickness independent of the unit moment of inertia (which is equal to $t^3/12$), so that the two behaviours are optimally calibrated (in the manner of ribbed plates, for instance).

Research and study of finite elements in problems of plates has been ongoing for forty years and shows no sign of having reached a conclusion. Numerous authors have grappled with the problem and this has led to a large number of formulations which, for various reasons, have been subjected to criticism that led to their demise.

The first problem to be addressed was that of thin plates, in the attempt to devise elements with three degrees of freedom per node (the translation normal to the plane of the element w, and the two rotations, φ and ψ, respectively about axes x and y) whose formulation required only a prediction about the displacement field. Unfortunately this approach proved to be untenable because the elements violated the properties of compatibility or completeness, or they tended to be unreliable when distorted. Recall that the function w(x, y) has to belong to C^1, that is, it has to be continuous with its first derivatives between element and element (the elastic surface cannot have any cusps).

In a second phase, research was aimed at devising elements that used derivatives of w of a higher order than the first, in order to have more parameters and thus complete polynomials, but this kind of element was never used very much, in part because it was difficult to set boundary conditions on the higher order derivatives, and in part because this kind of nodal unknown was more artificial.

In use at present are some finite elements that have proven to be thoroughly reliable for problems of either thick *or* thin plates, and others that can be used for both thick *and* thin plates. For thin plates we have the Discrete Kirchhoff Triangle (DKT) elements (set out in [Stricklin et al 1969] and reformulated and discussed in [Batoz et al. 1980]) and Discrete Kirchhoff Quadrangle (DKQ) elements (set out in [Batoz and Ben Tahar 1982]). For both thick and thin plates there is the Hughes element [Hughes and Tezduyar 1981] and the Dvorkin-Bathe element [Dvorkin and Bathe 1984]. All of these elements have a "complex" formulation that requires an expertise beyond the simple application of the formulas that we have discussed.

Because an in-depth discussion of the techniques used to develop these elements is beyond the scope of this book, we will limit ourselves to giving only basic information and refer the reader to the works cited in the bibliography for further study. For example, [MacNeal 1994] contains an interesting survey of the available elements and a history of their development.

11.7.2 DKT AND DKQ ELEMENTS (THIN PLATES)

DKT elements have three nodes, while DKQ elements have four [Stricklin et al. 1969; Batoz et al. 1980; Batoz and Ben Tahar 1982]. In both cases, there are three degrees of freedom per node: the translations outside of plane w and the rotations φ and ψ. The hypotheses used for Kirchhoff plates are set on a discrete number of points starting from an interpolation that treats w, φ and ψ independently. These added conditions lead to the elimination of the parameters that are in excess of those available. DKT and DKQ elements do not show a monotone convergence and are not compatible.

At present, DKT and DKQ elements are the preferred elements for the study of thin plates. [Batoz et al. 1980] proved that these elements possess a good capacity for (monotone) convergence (especially with regard to the triangular elements) and are preferable to others based on, for example, selective reduced integration (SRI).

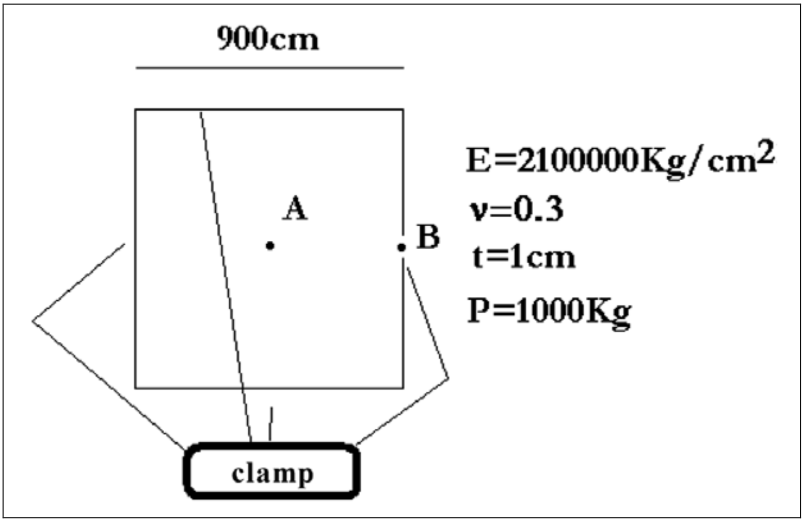


Figure 11-29: Problem of a square thin plate clamped on all four sides with concentrated load P at the centre.

For example, for the DKT and other elements, we will show the process towards convergence for the case of a square plate clamped on all four edges and subject to a concentrated load. This is a problem in which the elements are put to the test, given that in this case they are usually less precise than in other classic cases. This particular example was studied with CLEVER, the SARGON solver (which uses DKT elements for triangular plates, and an assembly of four DKT elements for plates with four nodes), as well as with other solvers for the other elements.

The "exact" results were not obtained simply by taking those listed in the classic text of Timoshenko and Woinowsky-Krieger [1959] for the following reasons. Timoshenko and Woinowsky-Krieger adopted a solving procedure based on a development in a series. Within the limits of the possibilities of calculating that existed fifty years ago, they used a certain number (four) of harmonics to determine the "exact" solution, but it is clear that by increasing the number of harmonics the solution can be made even more precise. In my opinion, it does not make much sense to compare the results of programs that may use thousands of equations with tables produced painstakingly fifty years ago with methods that were themselves approximate. For this reason, the results given here were obtained by significantly increasing the number of harmonics so as to stabilise the results, at least as regards the number of significant digits used here for the comparison (for example, Timoshenko obtains 23.5872 for the displacement instead of 23.638 and between the two numbers there is a difference of 0.2% that can't be attributed to the solvers). Thus, the results given here are taken from [Timoshenko and Woinowsky-Krieger 1959], in the sense that the method described there was used, but bearing in mind that fifty years have passed, and that it is possible to produce numbers that are much more accurate.

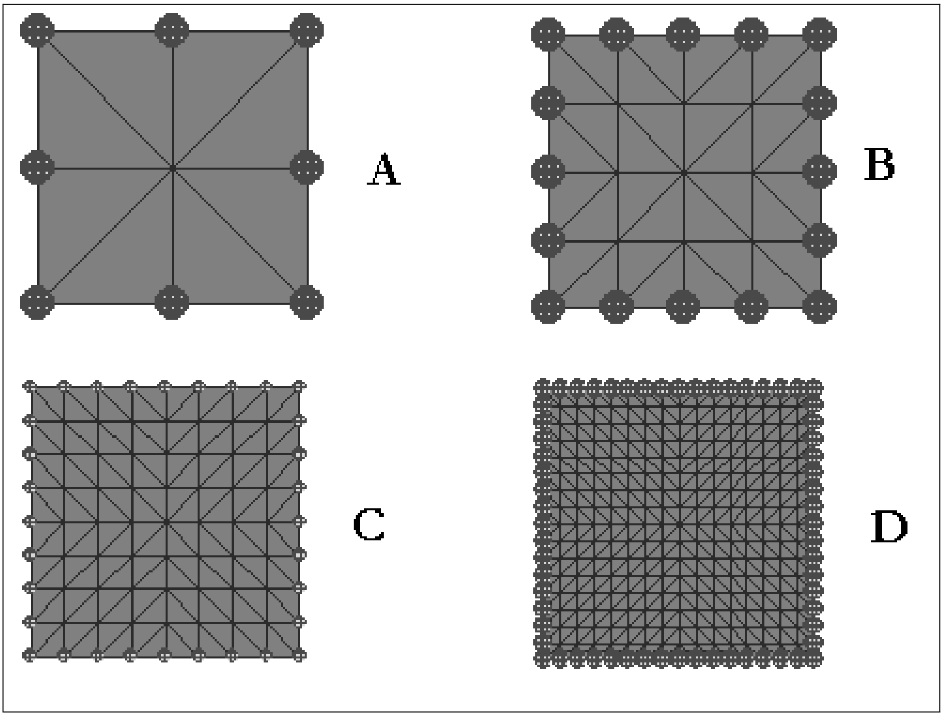


Figure 11-30: Meshes adopted for comparative study.

As we can see, the convergence of displacements is not monotone, because (surprisingly) the 2x2 mesh achieves a better result than the 4x4 or 8x8 meshes. Similar results have been reported in the literature. Table 11-10, for example, is taken from [Cook, Malthus and Plesha 1989], and was cited earlier in [Stricklin et al. 1969].

Mesh	**Displacement at centre w_A (Error %)**	**w_a/w_{exact}**	**Moment m_x at point B**	**$m_{xB}/m_{xB\ exact}$**
A 2x2	23.88 (1.01%)	1.01	136.1	1.080
B 4x4	24.66 (4.31%)	1.043	132.9	1.055
C 8x8	24.04 (1.69%)	1.017	129.0	1.024
D (16x16)	23.76 (0.5%)	1.005	127.5	1.012
Exact	23.64	1	126	1

Table 11-9: Results of convergence for DKT elements (SARGON).

Mesh	Simply supported Uniform load SS-U	Clamped Uniform load CL-U	Simply supported Concentrated load SS-C	**Clamped Concentrated load CL-C**
2x2	1.025	1.500	1.076	1.012
4x4	0.999	1.228	1.008	1.046
8x8	1.001	1.069	1.003	1.019
16x16	1.001	1.021	1.001	1.007

Table 11-10: DKT elements. Ratios of the calculated to the exact values, deflection of the central point of the square plate with ν=0.3. The exact value was calculated according to the theory of thin plates [Stricklin et al. 1969].

We also show the results obtained for the same problem shown in fig. 11-29 with the four-node DKQ elements [Batoz and Ben Tahar 1982], which is the element used by the family of SAP programs (SAP80®, SAP90® and SAP2000®), and with a 4DKT element obtained by placing four DKT elements on the quadrilateral (this element was proposed by ANSYS® and is also used by SARGON).

Mesh	Displacement at centre w_A (Error %)	w_a/w_{exact}	Moment m_x at point B (Error %)	$m_{xB}/m_{xB\ exact}$
A 2x2	26.325(11.4%)	1.114	150.00(19.0%)	1.190
B 4x4	27.001(14.2%)	1.142	117.10(7.1%)	0.929
C 8x8	24.831(5.0%)	1.050	123.97(1.6%)	0.984
D (16x16)	24.010(1.6%)	1.016	125.21(0.6%)	0.994
Exact	23.64	1	126	1

Table 11-11: Results of convergence for DKQ elements (SAP90®)

Mesh	Displacements at centre w_A (Error %)	w_a/w_{exact}	Moment m_x at point B (Error %)	$m_{xB}/m_{xB\ exact}$
A 2x2	24.98 (5.7%)	1.057	94.90(24.7%)	0.753
B 4x4	25.19 (6.5%)	1.065	117.84(6.5%)	0.935
C 8x8	24.20(2.4%)	1.024	124.61(1.1%)	0.989
D (16x16)	23.81(0.7%)	1.007	125.22(0.6%)	0.994
exact	23.64	1	126	1

Table 11-12: Results of convergence for 4DKT elements (SARGON).

11.7.3 DE VEUBEKE ELEMENTS (THIN PLATE)

Among the first elements proposed for thin plates were the De Veubeke elements [De Veubeke 1968]. What makes this element special is that it uses various interpolating functions in triangular subdomains of the original quadrilateral domain. The element is compatible: an analysis of table 11-13 makes its convergence toward the exact solution evident.

Mesh	Displacement at centre w_A (Error %)	w_a/w_{exact}	Moment m_x at point B (Error %)	$m_{xB}/m_{xB\ exact}$
A 2x2	21.935(7.2%)	0.928	29.169(76.1%)	0.239
B 4x4	21.228(10.2%)	0.898	96.575(23.4%)	0.766
C 8x8	22.782(3.6%)	0.964	112.794(10.5%)	0.895
D (16x16)	23.386(1.1%)	0.989	118.477(6.0%)	0.940
exact	23.64	1	126	1

Table 11-13: Results of convergence for De Veubeke elements (ALGOR).

The De Veubeke elements (used, for example, by ALGOR) converge rather slowly and – it seems to us – do not compare favourably with DKT.[6]

11.7.4 HUGHES ELEMENTS (THIN OR THICK PLATES)

The modelling of elements suitable for plate theory, often following approaches based on the direct interpolation of the three components of unknown displacement w, φ, and ψ, can give rise to significant locking effects, especially if the element tends to be thin. These can be overcome by means of techniques of reduced or selective integration which, however, can give rise to spurious modes, and thus to elements whose behaviour is unpredictable or unreliable.

Instead, the 4-node Hughes element performs as well as elements with reduced integration without however leading to the onset of any spurious modes of strain. There is also a triangular 3-node version of this element. The Hughes element performs well in modelling both thin plates (that is, relatively well, but not as well as the DKT element) and thick plates. The study in which it is presented [Hughes and Tezduyar 1981] provides a series of very detailed test cases, but unfortunately they are in the form of graphics rather than tables. The data that we give here is taken from those graphics and thus are liable to inevitable errors in measurement. They do, however, give an idea of how this element behaves.

Problem	**Mesh 8x8** **(Mesh C)**	**Mesh 32x32**
SS-U	0.977	0.995
CL-U	0.966	0.990
SS-C	0.995	≈1
CL-C	0.875	0.965

Table 11-14: Ratios of the calculated to the "exact" deflection at the centre of the plate according to thin plate theory. SS: simply supported; CL: clamped; U: uniform load; C: concentrated load. From [Hughes and Tezduyar 1981], data extracted from the graphics.

[6] We are obliged to point out that, for unknown reasons, the results reported in the literature for the constrained square plate under uniform load modelled with De Veubeke elements differ rather significantly from those obtained with ALGOR. The percentage errors for the displacement reported in [Zienkiewicz and Taylor 1989] are as follows: 4x4 -2.5% (instead of -10.2%); 8x8 +2% (instead of –3.6%). It is possible that Algor (which does not provide exact reference values for its own finite elements) used a version of the element that was different from that used by Zienkiewicz and Taylor.

Problem	Mesh 8x8	Mesh 32x32
SS-U	0.855	0.965
CL-U	0.822	0.955

Table 11-15: Relationships between calculated and "exact" moment at the centre of the plate according to thin plate theory. SS: simply supported; CL: clamped; U: uniform load; C: concentrated load. From [Hughes and Tezduyar 1981], data extracted from the graphics.

As we can see, the convergence for the displacement for our problem, that is, a square plate clamped on four sides with a concentrated load, is, given equal meshes, decidedly less satisfactory than that of the DKT element (0.875 compared to 1.019). However, in other cases, the convergence is more rapid, although it is never as rapid as DKT elements.

11.7.5 DVORKIN-BATHE ELEMENTS (THIN OR THICK PLATES, MITC4)

The so-called Dvorkin-Bathe elements [Dvorkin and Bathe 1984; Bathe 1996], which are valid for thin as well as thick plates, are formulated starting with the equations that appertain to solids. At present the corresponding 4-node element, or MITC4, Mixed Interpolation of Tensorial Components, is considered to be one of the best finite plate elements available, because it can contemporaneously deal with thin and thick plates, it allows the thickness to be variable, and it does not contain spurious modes.

Mesh	Displacement at centre w_A (Error %)	w_a/w_{exact}	Moment m_x at point B (Error %)	$m_{xB}/m_{xB\ exact}$
A 2x2	5.5714e-4(99.9%)	2.35e-5	125(0.8%)	0.992
B 4x4	20.408 (13.7%)	0.863	94.96(24.6%)	0.754
C 8x8	22.761 (3.7%)	0.963	115.57(8.3%)	0.917
D (16x16)	23.397 (1.0%)	0.990	123.109(2.3%)	0.977
Exact	23.64	1	126	1

Table 11-16: Results for the Dvorkin-Bathe element for the problem shown in fig. 11-29. (XFINEST: by courteous concession of Engineer Paolo Sattamino of Harpaceas srl).

Table 11-16 shows the results obtained for the problem shown in fig. 11-29 using a number of elements per side equal to 2, 4, 8, and 16, as in the case of DKT elements (although two DKT elements correspond to one MITC4 element).

Table 11-17 presents a comparison of MITC4 and DKT elements, and is taken from the study in which the MITC4 element was proposed [Dvorkin-Bathe 1984]. The problem concerned is shown in fig. 11-31.

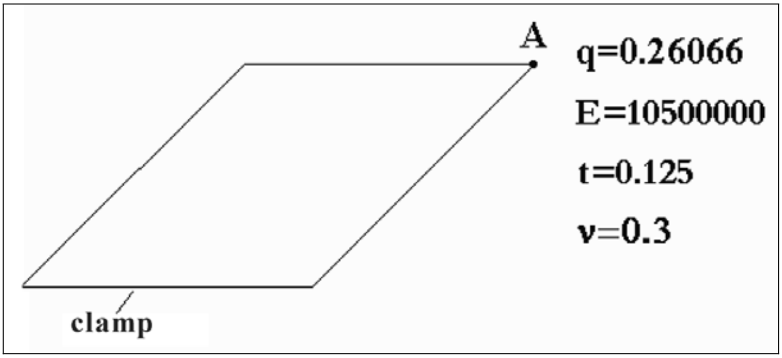


Figure 11-31: Plate problem: a rhombus-shaped plate under uniform pressure.

Element	Mesh	CPU time	Deflection at A
DKT	4x4	1	0.293
MITC4	4x4	2	0.272
Exact	-	-	0.297

Table 11-17: Analysis of a rhombus-shaped plate (distorted elements) subject to constant pressure. Comparative results taken from [Dvorkin-Bathe 1984].

In addition to the 4-node element, there are also elements with eight nodes (MITC8) or nine nodes (MITC9) and even higher [Bathe 1996].

11.7.6 THE "DRILLING DOF" PROBLEM

In examining the behaviour of membrane elements and bending elements, we have seen that while some formulations (for example, we have cited [Taylor and Simo 1985]) make use of the degree of freedom relative to the rotation about an axis perpendicular to the plane of the element, most do not. If that degree of freedom (called *drilling DOF*) is not used for either the membrane part or the bending part (and in effect none of the most widely-used elements use it), then this degree of freedom remains free.

For this reason, a simple model of a plate would show as many conditions of pseudo-mechanism as there are nodes, unless all of these conditions of pseudo-transience are not eliminated by explicitly constraining the rotations of these nodes.

If the model is “mixed”, in the sense that there might be other elements that are attached to this plate that are capable of limiting the pseudo-mechanism thanks to their contribution of their stiffness to the node, then deciding which nodes to constrain and which not, can be so slow and troublesome that effectively the pseudo-mechanism on the drilling mode becomes a problem.

In order to overcome this difficulty, the usual praxis is to perturb the stiffness matrix of the plate element, adding a fictitious term of stiffness on the null diagonal term of the matrix itself. Thanks to this addition, the drilling DOF of the plate is no longer free, but is rather attached to a modest stiffness that eliminates the pseudo-mechanism.

The amount of stiffness to add to the null diagonal term is usually set equal to the maximum diagonal term of stiffness in the matrix multiplied by a suitably small reductive coefficient (called a “drilling factor”), which is usually between 1.e-8 and 1.e-3 (although in special cases there is nothing to prevent values smaller or larger than this from being used, except for considerations we will mention below).[7] The amount of the correction is decided taking into account the following considerations.

The corrective term cannot be too small, because it will generate numerical instabilities due to the presence of terms on the main diagonal of the matrix that are too different from each other.

If there are never any other stiffnesses from other elements connected to the drilling DOF, then it is uncoupled from the other degrees of freedom and the value of the correction can be large without perturbing the solution in any way. This is, for example, the case of a flat plate subject to any kind of a load.

If instead there are other elements connected to the drilling DOF (as, for example, happens in a slab modelled with plate elements to which beam elements or plate elements normal to it are connected, to simulate columns or dividing walls),then the addition of the corrective terms to the stiffness is superfluous and might interfere with the “true” stiffness, modifying the solution. Note that the corrective term is added when the matrix of the element is created, and thus during assembly, when we don’t know – nor can we know – if in the end there will be other, non-fictitious stiffnesses on that degree of freedom. In

[7] In SARGON the default value is 1e-5.

these cases, the corrective term should be as small as possible (except for numeric problems), in order to reduce to a minimum any perturbing effects.

If it is undertaken to constrain all of the pure drilling DOFs (that is, those without any contribution of real stiffness), leaving all the others free, then the drilling factor can be set equal to zero and the matrix won't be perturbed in any way.

11.8 SOLID ELEMENTS (BRIEF REMARKS)

11.8.1 OVERVIEW

Solid elements are used when no other kind of modelling is suitable for schematising the problem at hand. In solid problems the unknown displacement functions are the three translations u(x, y, z), v(x, y, z) and w(x, y, z). The nodal displacements normally used are the translations of the nodes in directions x, y and z, while the rotations of the nodes do not enter into the formulation and have to be blocked.

The development of solid elements follows rather closely along the lines already discussed for elements of plane stress. Here too simple (tetrahedral) elements of constant strain can be defined, followed eventually by elements of a higher order. In this case as well it is possible to have isoparametric elements with a variable number of nodes, which make the treatment of the problem uniform and efficient.

11.8.2 4-NODE TETRAHEDRAL ELEMENT (TETRA4)

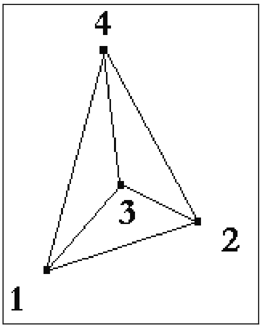


Figure 11-32: Solid 4-node tetrahedral element.

As its name implies, the tetra4 element has four nodes and thus interpolates the displacement functions by means of a complete linear polynomial (note that here there are three variables, rather than two in the case of plane stress and plane strain):

$$u(x, y, z)= ax+by+cz+d$$
$$v(x, y, z)= ex+fy+gz+h$$
$$w(x, y, z)= ix+ly+mz+n.$$

The shape functions make it possible to write the interpolations in the standard way:

$$u(x, y, z)=u_1\Phi_1+ u_2\Phi_2+ u_3\Phi_3+ u_4\Phi_4$$
$$v(x, y, z)=v_1\Phi_1+ v_2\Phi_2+ v_3\Phi_3+ v_4\Phi_4$$
$$w(x, y, z)=w_1\Phi_1+ w_2\Phi_2+ w_3\Phi_3+ w_4\Phi_4.$$

where the values for **Φ** can be obtained by inverting the following matrix relationships:

$$\begin{vmatrix} 1 \\ x \\ y \\ z \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ z_1 & z_2 & z_3 & z_4 \end{vmatrix} \begin{vmatrix} \Phi_1 \\ \Phi_2 \\ \Phi_3 \\ \Phi_4 \end{vmatrix}.$$

The strain predicted by the element is constant, and is thus easy to understand that the convergence that can be obtained with an element of this kind is quite slow, comparable to that for a CST element seen previously for plane problems.

11.8.3 10-NODE TETRAHEDRAL ELEMENT (TETRA10)

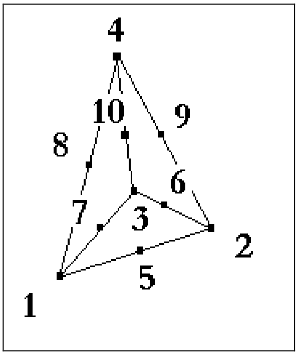


Figure 11-33: Solid 10-node tetrahedral element (tetra10).

In the case of the tetrahedral element with ten nodes, the shape functions provide the following approximations u, v and w:

$$u(x, y, z) = ax^2+by^2+cz^2+dxy+eyz+fxz+gx+hy+lz+m,$$

with analogous polynomials used for v and w.

This is a second-order complete polynomial. An examination of the strain allows us to see how strain depends on coordinates (taking ε_x, for example, since all the other dependencies of the other components of strain are the same):

$$\varepsilon_x \rightarrow 2ax+dy+fz+g$$

that is, a first-order complete polynomial: the dependence of the stress is therefore linear.

11.8.4 8-NODE ISOPARAMETRIC ELEMENT (HEXA8)

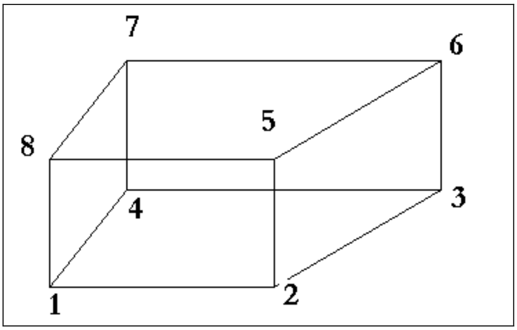


Figure 11-34: Solid 8-node isoparametric element.

For the isoparametric element with eight nodes, the interpolating function for u, v and w contain eight terms, and are third-order incomplete polynomials:

$$u(\xi, \eta, \zeta)=a\xi\eta\zeta+b\xi\eta+c\eta\zeta+d\zeta\xi+e\xi+f\eta+g\zeta+h$$

with analogous polynomials used for v and w.

The shape functions can be expressed in the following way

$$\Phi_i=\frac{1}{8}(1+\xi\xi_i)(1+\eta\eta_i)(1+\zeta\zeta_i) \qquad i=1,\ldots 8$$

where we always have

$$u(\xi,\eta,\zeta)=\sum_1^8 \Phi_i u_i$$

$$v(\xi,\eta,\zeta)=\sum_1^8 \Phi_i v_i$$

$$w(\xi,\eta,\zeta)=\sum_1^8 \Phi_i w_i \ .$$

11.8.5 20-NODE ISOPARAMETRIC ELEMENT (HEXA20)

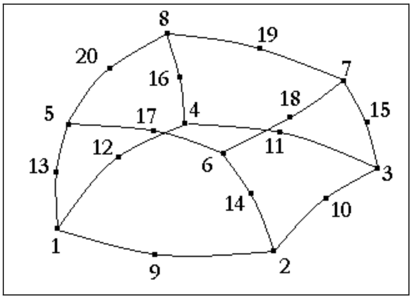


Figure 11-35: Solid 20-node isoparametric element.

In the solid isoparametric element with twenty nodes the interpolating functions for u, v and w contain twenty terms, and are fourth-order incomplete polynomials:

$$u(\xi, \eta, \zeta)=a\xi^2\eta\zeta+ b\xi\eta^2\zeta+ c\xi\eta\zeta^2+ d\xi^2\eta+ e\xi^2\zeta+f\xi\eta^2+ g\eta^2\zeta+ h\xi\zeta^2+ i\eta\zeta^2+$$
$$l\xi\eta\zeta+ m\xi^2+ n\eta^2+ o\zeta^2+p\xi\eta+q\eta\zeta+r\zeta\xi+s\xi+t\eta+u\zeta+v$$

with analogous polynomials used for v and w. These correspond to the following choices for the shape functions:

$$\Phi_{\mathrm{i}}=\frac{1}{8}(1+\xi\xi_i)(1+\eta\eta_i)(1+\zeta\zeta_i)(\xi\xi_1+\eta\eta_1+\zeta\zeta_i-2) \quad \mathrm{i}=1,\ldots 8$$

$$\Phi_{\mathrm{i}}=\frac{1}{4}\left(1-\xi^2\right)(1+\eta\eta_i)(1+\zeta\zeta_i) \qquad \mathrm{i}=9,\ 11,\ 17{,}19$$

$$\Phi_{\mathrm{i}}=\frac{1}{4}\left(1-\eta^2\right)(1+\xi\xi_i)(1+\zeta\zeta_i) \qquad \mathrm{i}=10,\ 12,\ 18,\ 20$$

$$\Phi_{\mathrm{i}}=\frac{1}{4}\left(1-\zeta^2\right)(1+\xi\xi_i)(1+\eta\eta_i) \qquad \mathrm{i}=13,\ 14,\ 15{,}16$$

where

$$u(\xi,\eta,\zeta)=\sum_{1}^{20}\Phi_i u_i$$

$$v(\xi,\eta,\zeta)=\sum_{1}^{20}\Phi_i v_i$$

$$w(\xi,\eta,\zeta)=\sum_{1}^{20}\Phi_i w_i\ .$$

12 PROBLEMS IN MODELLING

12.1 RENDERING AND MODELLING

In the course of this discussion, we have mentioned more than once that a finite element model is not a drawing. Unless this concept is thoroughly understood, no completely valid finite element model can be constructed.

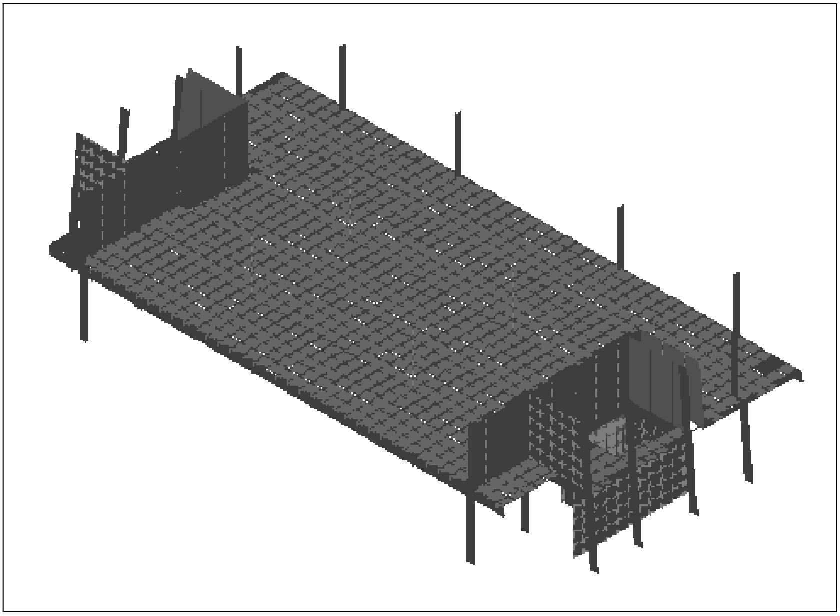


Figure 12-1: A mesh that appears well made at first glance ...

The object of a finite element model is *not* to depict the actual structure to be modelled. The tendency to make models that physically resemble the structure in question, so that the analyst and the client can look at the model and forget that it is just that – a model – is *absolutely unproductive and even reprehensible*. This *painter's approach* only leads us astray, because *finite element models are made of finite elements that model simplified theories, not direct physical reality*.

What counts is not that the model looks like the physical reality, but that it is capable of grasping some fundamental aspects of the structure's displacements, strains and stresses that can be measured quantitatively as they are progressively transformed by the analyst into a coherent system of patterns and components. Finite element models are like poetry translated from one language to another: we can have either *beautiful untruthfulness* or *ugly truthfulness*: there is no guarantee that a model that is almost photographic in its resemblance is a truthful depiction of behaviour. It is one thing to photograph, but another thing to create a truthful finite element model.

Structural analysis with finite elements
978-0-7277-4093-9

Often these considerations lead to parts that are modelled with scant regard for problems of discretization, so that, for example, slabs (which are nice to see because "they're there") are modelled with one or two plate-shell elements, with the resulting low degree of reliability that we might expect. In other cases it is the basic concepts that are lacking. Take, for example, the structure shown in fig. 12-1, which seems at first glance to be adequately modelled.

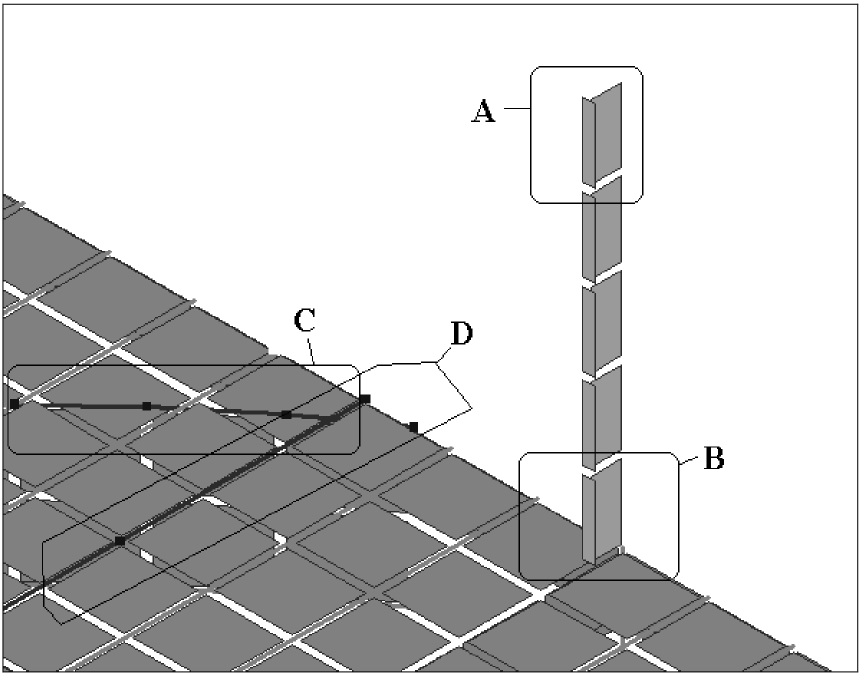


Figure 12-2. **…turns out to be ill disposed.**

If we zoom in on it, we can see that the model (which was actually made by a draftsman who decided to improvise as a structural analyst and then sent it to us for checking) is full of errors, and makes the incompetence of its creator evident. In fig. 12-2 we can observe the following errors:

- Area A shows that the web and the flange are not connected, but only facing: the entire flange is modelled with a single finite element without regard for the intersection with the web, to which it is not connected. The *drawing* is correct; the *model* is completely in error. Note also – among other things – the (revealing) attempt to model the bending of a T element *with a single finite element for the web*;
- Area B shows that this vertical T element – itself erroneously modelled, as we have just pointed out – comes down to a deck to which it appears to be connected, but is actually only facing, because the elements below don't have nodes that correspond to the nodes of the vertical element.

Thus the already erroneously modelled element, not connected to the plane as it should be, is thus doubly erroneous. Note that once again the *drawing* is correct, but the model is completely incorrect. Our designer simply did what he is trained to do: he *drew*.

- Area C shows a wall arriving at an angle, bounded by reinforcing beams: it shoots in like a meteor without any connection to the part above it. The structural elements are individually modelled and thrown into the three-dimensional world of the model without being connected in any way;
- Area D shows the connecting beam element going past three elements without connecting to them, thus violating all possible laws of congruency;

An examination of the remaining parts of the structure confirmed that the entire model was made like this. When faced with models such as this one, the only thing to do is discard them, because any attempt to patch or repair them is more time-consuming than simply starting over. This is because any adjustment made, whether manual or automatic (if this is even possible) has to be in any case verified in a capillary manner. It's just better to start from scratch.

A final example, this too taken from an actual case in practice, is shown in fig. 12-3.

Here the numbering of the elements shows that a single finite element was used to model two of the columns, that in the centre and that on the left (element 1 and element 3), rather than using the minimum four elements required. The result is the deformed configuration shown on the right, which is clearly incorrect. The inexperienced user has mistakenly modelled the frame, violating the congruence between beams and diagonals on one side and columns on the other. Note that in spite of this, the model is not hypostatic and can be solved, and thus if the user does not check and uses the stresses directly rather than asking himself "Is the model correct?", then the result will be that the numbers used in the design are trash.

Other examples of structures with disconnections are shown in figs. 2-8, 2-9, 4-9, 4-13, 4-14, 4-15, 4-16, 9-8 (this last a classic error regarding trusses).

A further example (although I have encountered innumerable similar examples in practice) of the "painter's approach" is this one: take shelf structures in steel made by joining vertical and horizontal components. The verticals are cold rolled sections with holes; the horizontals have teeth at the ends,which are intended to go into the holes of the verticals to create the joint. The idea of modelling the teeth with small beam elements is bizarre: it is obvious that modelling the horizontal at the scale of the beam element makes it impossible to model details such as the joint system (which if anything has to be accounted for by means of semi-rigid nodes). If we want to model the tooth, then the entire horizontal has to be modelled with plate-shell elements (as well as the vertical, obviously), and modelling a single horizontal would require tens of thousands of degrees of freedom. The inexpert but zealous analyst thinks that adding the teeth (which are actually physically present) with tiny beam elements helps improve the model, but in reality the teeth shouldn't be there. *When making the model it is always necessary to take into account the scale adopted to model the structure*: if it was decided to adopt a model that disregards local effects, it is not possible to then add smaller, more or less refined areas in the hope that they will restore the lost details.

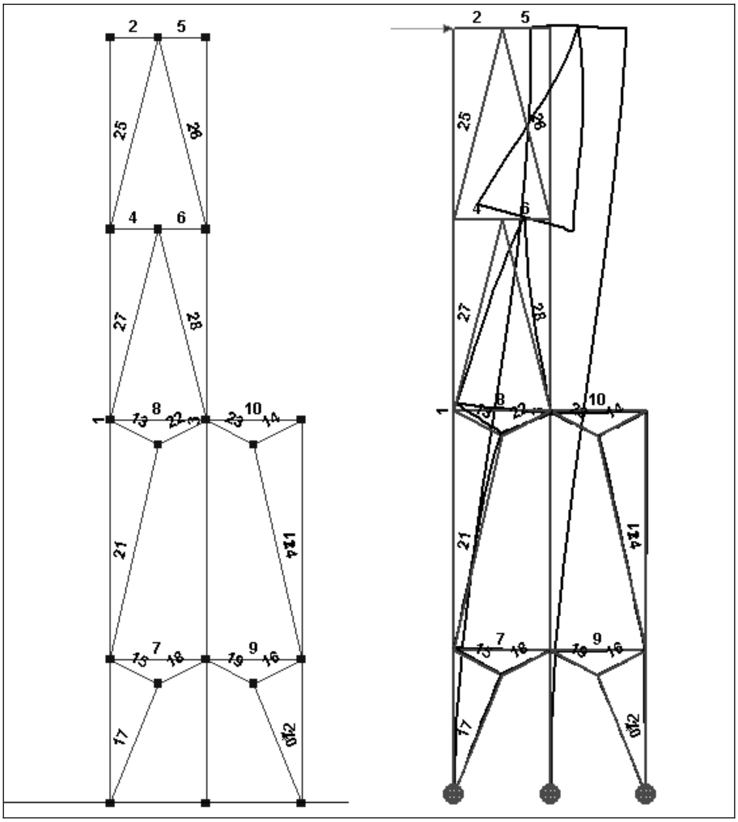


Figure 12-3: Example of a model with erroneous connections and its deformed configuration: note that even though the model is completely incorrect, it is not hypostatic, and thus it is solvable.

Finite element models do not have to include *everything*. They need only include what makes it possible to describe the structural behaviour completely and in keeping with other parts at a given scale of analysis.

12.2 Typical structures

12.2.1 Continuous beams

Beam elements are used to model continuous beams, using at least one element per bay. Given that the beam is continuous, there are no constraints to be applied. The three translations of nodes that correspond to the ground hinges that do not break member continuity have to be constrained but they must be left unconstrained for rotation. Table 12-1 shows a possible pattern of constraints for use in the various cases shown in fig. 12-4 (presuming that the loads are directed along axis Z: case A is axially isostatic; cases B and C are axially hyperstatic). We presume that the behaviour is planar, and thus that the translations perpendicular to the plane of the drawing are null).

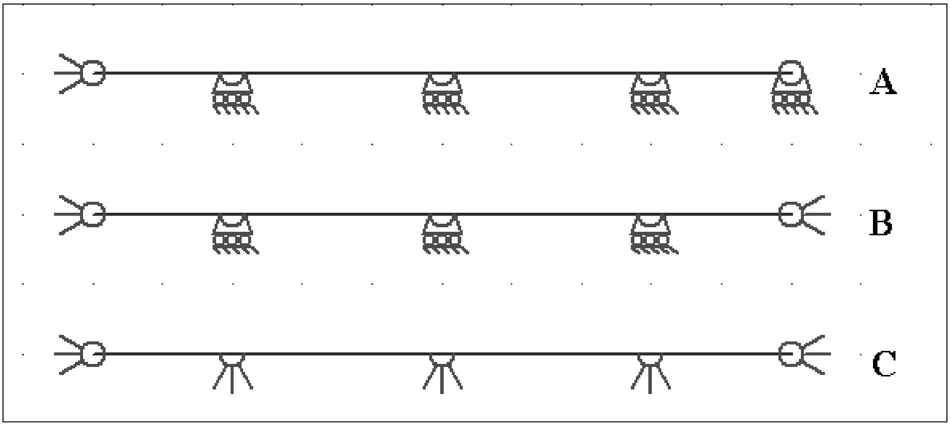


Figure 12-4: Various possible models for continuous beams.

In the case where the beam shows some discontinuities, careful consideration must be given to the end releases to be introduced on the beam elements.

Fig. 12-5 shows various possible cases of discontinuity: the axis line of beam D is discontinuous at its centre point. Beam E shows a discontinuity but is also supported. Beam F has no discontinuities, but does have unsupported ends.

Table 12-2 shows how these situations can be modelled, that is, what constraints and end releases to apply (the numbering of nodes and elements is given; the element numbers appear at the mid-point of the axis line, the node numbers appear above the nodes).

Direction of the axis of the element	Type	Typical nodes simply supported	Starting nodes	Final nodes	Possible intermediate bay nodes
X	A	Ty, Tz	Tx, Ty, Tz, Rx	Ty, Tz	Ty
X	B	Ty, Tz	Tx, Ty, Tz, Rx	Tx, Ty, Tz	Ty
X	C	Tx, Ty, Tz	Tx, Ty, Tz, Rx	Tx, Ty, Tz	Ty
Y	A	Tx, Tz	Tx, Ty, Tz, Ry	Tx, Tz	Tx
Y	B	Tx, Tz	Tx, Ty, Tz, Ry	Tx, Ty, Tz	Tx
Y	C	Tx, Ty, Tz	Tx, Ty, Tz, Ry	Tx, Ty, Tz	Tx

Table 12-1: Constraints for the various nodes on a continuous beams, with the Z-axis vertical, and axis X or Y horizontal.

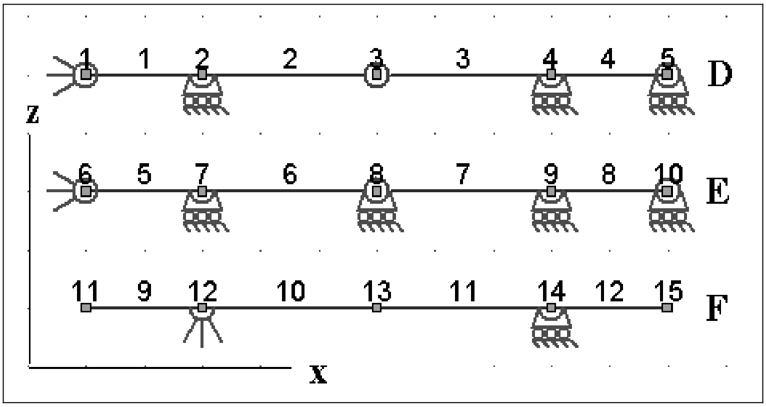


Figure 12-5: Continuous beams with interruptions and unsupported ends (graphic representations made with C.E.S.CO; the small squares indicate nodes).

For simplicity's sake we suppose that the direction of the axis line coincides with that of global axis X, that axis Z is vertical, and that the three local axes of the elements are all oriented such that

Axis 1 = Axis X

Axis 2 = Axis Y

Axis 3 = Axis Z.

This results in the constraints and end releases given in Table 12-2:

Node	Out of plane constraint Solution A	In plane constraint
1	**Ty**, Rx	Tx, Tz,
2	Ty	Tz
3	Ty	
4	Ty	Tz
5	Ty	Tz
6	Ty, Rx	Tx, Tz
7	Ty	Tz
8	Ty	Tz
9	Ty	Tz
10	Ty	Tz
11	Rx	
12	Ty	Tx, Tz
13	Ty	
14	Ty	Tz
15	Ty	

Table 12-2: Constraints on the nodes shown in fig. 12-5.

There are various aspects of the problem to be discussed:

Translation Ty is constrained. Thanks to this precaution, all of the out-of-plane translations and the global rotation about the z-axis are constrained. It would also have been possible to constrain translation Ty and rotation Rz at only one (the first or any other) node in order to eliminate rigid motions (see Table 12-4, solution C). In this last case, however, the out-of-plane behaviour of the structure would have been like a cantilever. If there are no forces at all acting to create this displacement, then these two methods are equivalent.

Rotation Rx is constrained. If this rotation is constrained at a single element, then all rotations Rx are constrained thanks to the torsional stiffness of the beam (solutions A and C). If it is not constrained, then all of the rotations are unconstrained because globally the beam is free to rotate about the x-axis, giving rise to a condition of hypostaticity, and therefore a mechanism. It is also possible to constrain rotations Rx at all nodes (solution B). Practically speaking, it is possible to apply the constraint template TyRxRz to all nodes in order to make the problem planar (solution B).

Element	Left release	Right release
1	None	None
2	None	R2
3	None	None
4	None	None
5	None	None
6	None	R2
7	None	None
8	None	None
9	None	None
10	None	None
11	None	None
12	None	None

Table 12-3: Releases on the elements shown in fig. 12-5.

Node	Out of plane constraint Solution B	Out of plane constraint Solution c
1	Ty, Rx, Rz	Ty, Rx, Rz
2	Ty, Rx, Rz	
3	Ty, Rx, Rz	
4	Ty, Rx, Rz	
5	Ty, Rx, Rz	
6	Ty, Rx, Rz	Ty, Rx, Rz
7	Ty, Rx, Rz	
8	Ty, Rx, Rz	
9	Ty, Rx, Rz	
10	Ty, Rx, Rz	
11	Ty, Rx, Rz	Ty, Rx, Rz
12	Ty, Rx, Rz	
13	Ty, Rx, Rz	
14	Ty, Rx, Rz	
15	Ty, Rx, Rz	

Table 12-4: Alternate formulations for out-of-plane constraints for beams shown in fig. 12-5.

Hinge at the centre (where present: D, E). Here it has been decided to apply an end release to the beam to the left of the node and not to apply one to the beam on the right (the solution shown on the top of fig. 12-6). We could also do the inverse: apply an end release to the beam to the right of the node and not to apply one to the beam on the left (the solution shown in the middle of fig. 12-6). If both of the rotations of the node had been released, and it was no longer associated to any stiffness, it would have been free, giving rise to a pseudo-mechanism. Thus in that node rotation Ry *should have been constrained.* Usually determining the correct measures of constraints and end releases to apply is problematic for the beginner who has not yet completely grasped exactly how one differs from the other.

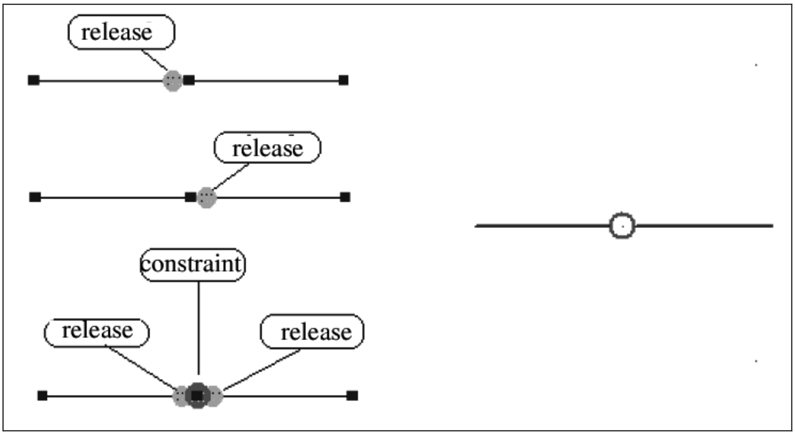


Figure 12-6: Three ways of simulating a hinge between elements (graphic conventions used by SARGON): a dot for the releases (*next to* the node), a dot for the constraints (*on* the node).

Terminal ground hinges or simple supports. In this case the continuity of the beam is guaranteed by the absence of end releases. On the other hand, the freedom of the node to rotate but not to translate is guaranteed by the constraints on the ground that leave the node free to rotate.

Mid-span simple supports. In this case the interruption of the axis line is guaranteed by a suitable release. That node (8) has ground constraints that are different from the homologous node not on the ground (3) only implies that the hinge cannot move down, that is, that Tz is fixed (and thus implying a simple support). The simple support is modelled by the constraint on the node. The mid-span hinge is modelled by the release on an element (two in the case where the rotation is constrained).

Naturally, models that generate situations of varying degrees of hypostaticity such as those shown in fig. 12-7 are not permitted. Note that though it is clear that the structures shown have some degree of hypostaticity, *it is not always clear that placing the constraints or releases in a certain way in fact implies that the resulting structure will be like those shown.*

In the case where it is axis 3 rather than axis 2 that coincides with axis Y, then the release should not be placed on R2, that is, on the rotation about local axis 2, but on R3. The case in which some beams are oriented so that axis 2 is parallel to axis Y and others are oriented so that axis 3 is parallel to axis Y is particularly subtle. Here it is necessary to make sure exactly which component of internal action needs to be released.

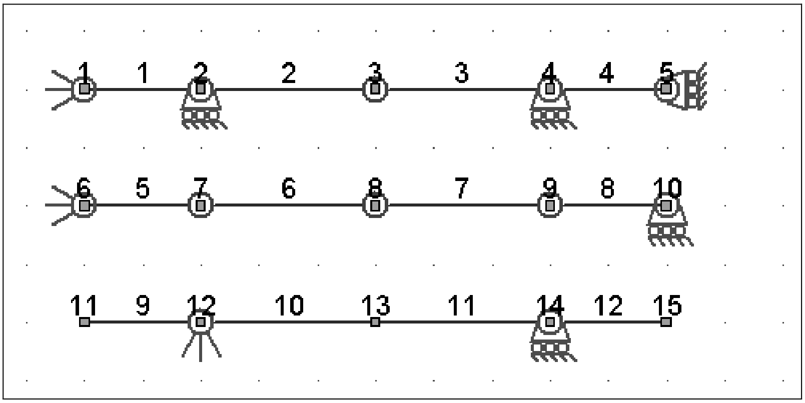


Figure 12-7: Structures that present conditions of hypostaticity.

We should mention again the fact that if the displacements are what interest us, then it is usually appropriate to "put a node" under the concentrated loads within the elements (which thus come to be considered as subdivided), while if only the internal actions are of interest to us – and if the software program adds the local effects, as it should – then a single finite element per bay is sufficient. It is obviously totally incorrect to model the entire continuous beam with a single element.

12.2.2 TRUSSES

In the case of trusses, the first thing to decide is if we want to model only the axial actions, or if we want to model the shears and moments as well, and if so, on which elements we want to model them. If there are only axial actions, then the necessary condition is that all of the forces with components normal to the axis line of the elements are applied at the nodes, and that all the elements are hinged at their ends (each vertical and diagonal element and each individual sub-span element).

If we want only axial actions, then the model can be made (for the sake of brevity of analysis) entirely with rod elements. In this case it is necessary to constrain all of the components of displacement to which no stiffness is associated. *No hinges should be placed because rod elements are hinged by definition.* Thus all of the rotations of the nodes should be constrained, as should be all of the translations normal to the plane of the truss (assumed parallel to one of the coordinate planes). The nodes that correspond to ground hinges should be further constrained for both of the remaining components of translation; the nodes that correspond to simple supports should be constrained only for translation in the direction normal to the support. Table 12-5 shows the pattern of constraints for trusses modelled with rod elements in plane XZ or in plane YZ.

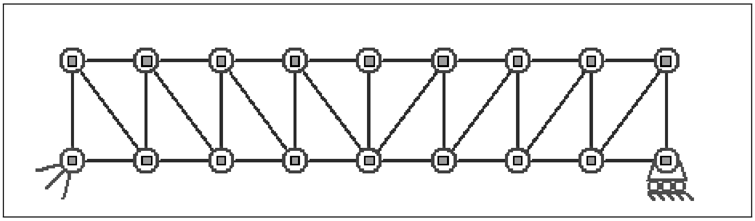


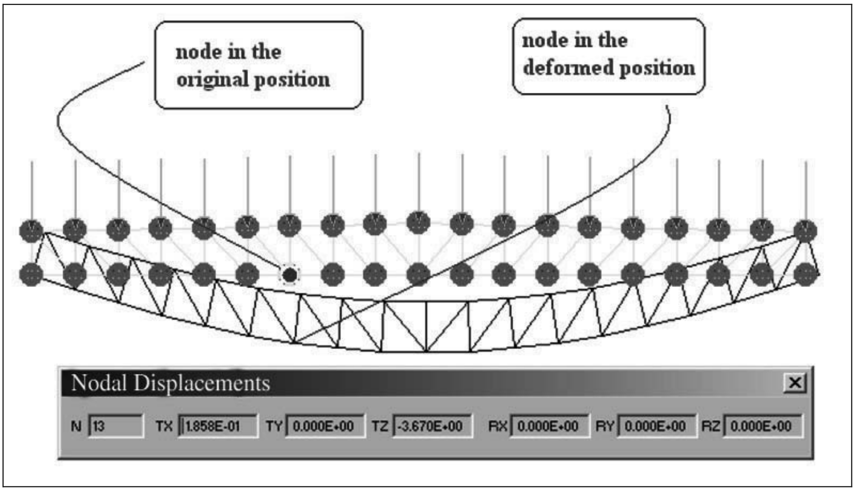
Figure 12-8: Diagram of a truss modelled with rod elements: there are only axial actions if the forces are applied only at the nodes (otherwise it is necessary to use beam elements).

Truss in plane	**On all nodes**	**On ground hinges**	**On simple supports (direction of free move)**
XZ	TyRxRyRz	TxTyTzRxRyRz	TyTzRxRyRz (X)
YZ	TxRxRyRz	TxTyTzRxRyRz	TxTzRxRyRz (Y)

Table 12-5: Constraints to be placed on the nodes in trusses modelled with rod elements.

The fact that all six of the possible constraints have to be put on the hinges (as if this was a rigid joint) is always confusing for the beginner, who might ask, “If this is a hinge, then why do I have to make it fixed?”. The fact is that rod elements are “hinged by definition” and thus cannot carry any moment. The nodes to which they are attached are then always free to rotate: *it is this “spurious” rotation, that component of the node’s movement to which no stiffness is attached, which has to be constrained*, not the possibility of the element’s translation, so that their ends rotate with respect to the original position.

This is demonstrated by fig. 12-9, which shows the deformed configuration of a model of a truss made entirely with rod elements, with all nodes constrained as described above (in particular, Ry is null everywhere, and we are in plane XZ). As we can see, the fact that the rotation is null does not mean that the axis line remains horizontal; in fact, the rod element is not connected rotationally in any ways at its end nodes, and it is thus free to rotate at will (in accordance with the translation of its end nodes).


node in the original position

node in the deformed position

Nodal Displacements

N 13 TX 1.858E-01 TY 0.000E+00 TZ -3.670E+00 RX 0.000E+00 RY 0.000E+00 RZ 0.000E+00

Figure 12-9: Truss modelled with rod elements: rotation Ry of the node is null, but the chords have deformed, and in particular, when deformed the axis line is not horizontal.

♣

A second scheme typically used for trusses calls for a mixed model, with beam elements used for the chords, and truss elements for the struts on the sides (verticals and diagonals). This scheme has two advantages with respect to the one examined above. The first advantage is that it makes it possible to model situations in which the loads are not directly transmitted to the nodes, but are rather transmitted in ways that can even include distribution along the chords, that is, loads that are concentrated but at points other than the nodes. The second advantage is that we can study the parasite bending moments (with respect to the pure grid scheme) arising from the fact that the chords are generally made of just a few continuous pieces.

Fig. 12-10 shows two possible schemes for modelling trusses, each with beam elements used for the chords and truss elements used for the verticals and diagonals. The two schemes are different because the second one uses beam elements for the two end verticals as well. The figure also shows the constraints to place on the nodes, and the releases (R2) to place on the beam elements: we presume that the plane is XZ, and that axis 2 of the beam element is axis Y, perpendicular to the plane of the structure. If axis 3 were perpendicular, then in place of R2 we would have to consider R3.

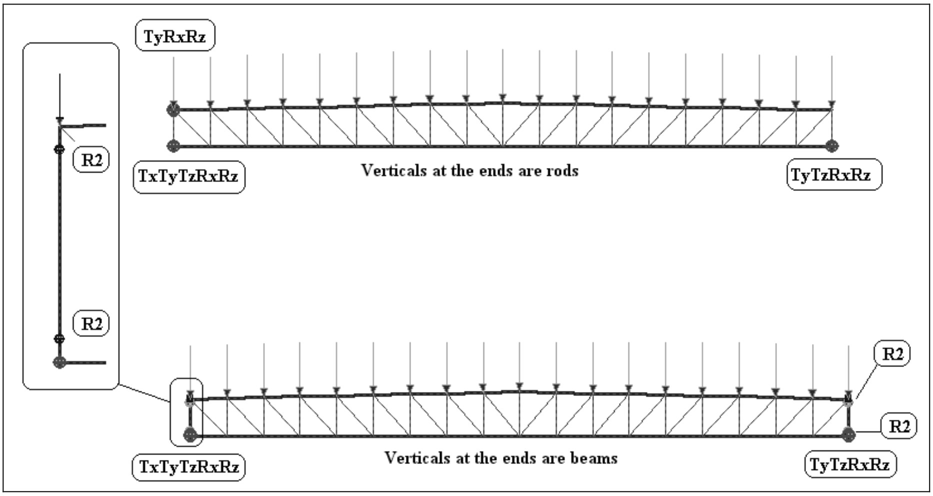

TyRxRz
R2
TxTyTzRxRz
Verticals at the ends are rods
TyTzRxRz
R2
R2
R2
TxTyTzRxRz
Verticals at the ends are beams
TyTzRxRz

Figure 12-10: Possible schemes for modelling trusses with beam elements.

We can make the following observations:

- With respect to the previous case in which the structure was modelled entirely with rod elements, the number of Ty constraints is much lower, because, since the chords are modelled with unreleased beam elements, we need only fix translation y of *one* of its points and the rotation RZ of *that* point in order to prevent (all of) the points of the chord from translating freely in a direction perpendicular to the plane of the truss (see the model shown on the top in fig. 12-10).
- To this can be added the constraint on rotation Rx that impedes the rotation of the node about the X axis. Once again a single point is sufficient because the chord has a torsional stiffness, since it has been modelled with beam elements.
- The second model uses beam elements for the verticals on the ends, and thus those verticals – if not released – would be rigidly joined to the chords, generating undesirable parasite moments. Thus the two ends of both end verticals have to be released for R2 so that the rotation of the nodes, joined to the chords but not with the verticals, does not provoke any bending moments in them.
- The behaviour of the end verticals for out-of-plane bending is instead not released with respect to the chords, so that, out of plane, there is a kind of elastic frame that thus makes it possible to specify the out-of-plane constraints definitively, on only the bottom chord.

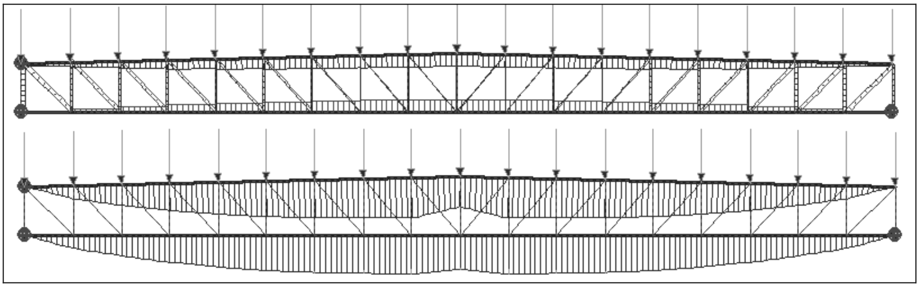


Figure 12-11: Diagram of axial actions and bending moments (there are bending moments only on the chords).

Fig. 12-11 shows the diagrams of axial actions and bending moments. The most part of the bending moment acting on the truss as a whole is generally carried by the axial action. For example, in the case shown in fig. 12-11 we have:

NH=33610 * 1200=4.033e6

Ms+Mi=82070+94040=1.76e5

NH/(Ms+Mi)=22.9

where N is the maximum axial action of the chords; H the maximum height of the truss; Ms and Mi are the moments at the centres of the top and bottom chords.

If we chose sections with a low moment of inertia (as is usually the case with trusses), then the values of the moment can be further reduced. For example, if instead of HEB100 we use double angles 75 x 7, we have:

NH=33720 * 1200=4.046e6

Ms+Mi=23640+27810=5.14e4

NH/(Ms+Mi)=78.7.

Thus, the effect of parasite moments increases as the moment of inertia of the chord increases, so that it makes sense to keep them rather slender in terms of bending.

If we want to take into account that the chords may be divided into two, for example, because in the central part there is a bolt, then we want to release the chord for bending, as shown in fig. 12-12, for example.

In this case we should release only one of the two beam elements on either side of the node (it doesn't matter which one), because if both are released then we must also – in order to avoid conditions of hypostaticity – add a constraint on rotation Ry of the node.

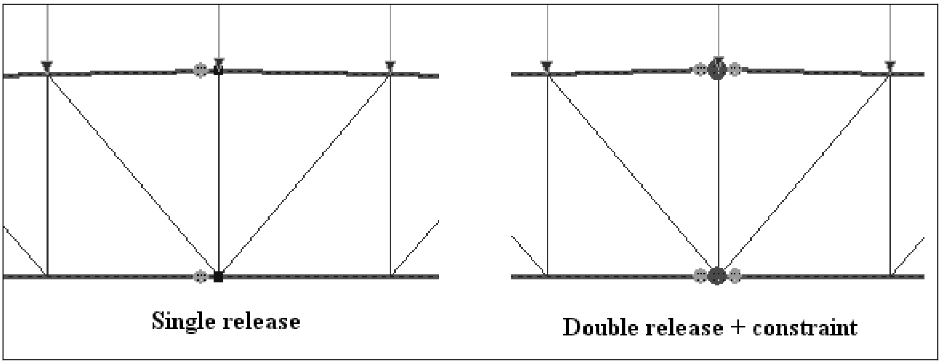


Figure 12-12: Releasing the centre point of the chords.

♣

A third way of modelling trusses, although actually only rarely used, is that of using beam elements that are not released, and thus taking into account all of the "parasite" forces on all of the elements. In this case there are no releases to be placed, and the application of constraints concerns only two nodes, those at the ends of the bottom chord (or, for specific kinds of trusses, the top chord). Finally, we should mention one typical (serious) error that is unfortunately made rather often. If the truss has been modelled with unreleased beam elements and instead we want to have only axial actions, then it is absolutely incorrect to constrain all of the rotations "so that the elements won't bend". If this is done, then the bending elements will remain and constraints will be introduced that do not actually exist, thus irreparably perturbing all of the stress. Fig. 12-13 shows a detail of the deformed configuration of a model incorrectly made in this way. The incorrect model derives from that shown in fig. 12-10, modified by imposing a null rotation on all of the nodes of the chords. The overall deformed configuration recalls morphologically that of fig. 12-9 (although the deflection is clearly less than the exact one; see fig. 12-14), but locally we can see that all of the elements of the chords are deflected with the point of deflection located almost exactly at the centre. The resulting moment diagram is shown in fig. 12-14, where the maximum incorrect moment arrived at 8e5, against an expected maximum of 2.7e4. Errors such as this one are often made when no checks are performed, and even by those who are professionals in the analysis of finite elements[1].

[1] In my own practice I once had to check a very important model in a nautical application that was made with plate-shell elements by a firm that was supposed to specialise in finite element analysis, in which, in order to eliminate the bending part of the stress in the shell, all of the nodes had been constrained for rotation. This was followed by a modal analysis and a frequency response analysis to estimate the vibrations. The model turned out to be totally incorrect, because the shells continued to deflect, discharging "to the ground" reactive couples that were completely non-existent at all nodes that were incorrectly constrained. The actual frequencies of vibration were all incorrect (much higher than in reality), even though qualitatively the modal deformations appeared

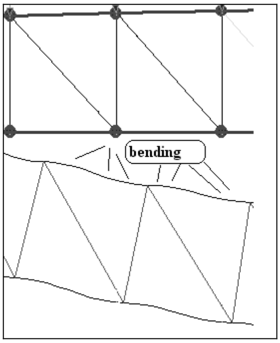


Figure 12-13: Incorrect constraints: the deformed configuration of a truss modelled using unreleased beam elements for the chords and rod elements for the verticals and diagonals with the nodes constrained for rotation.

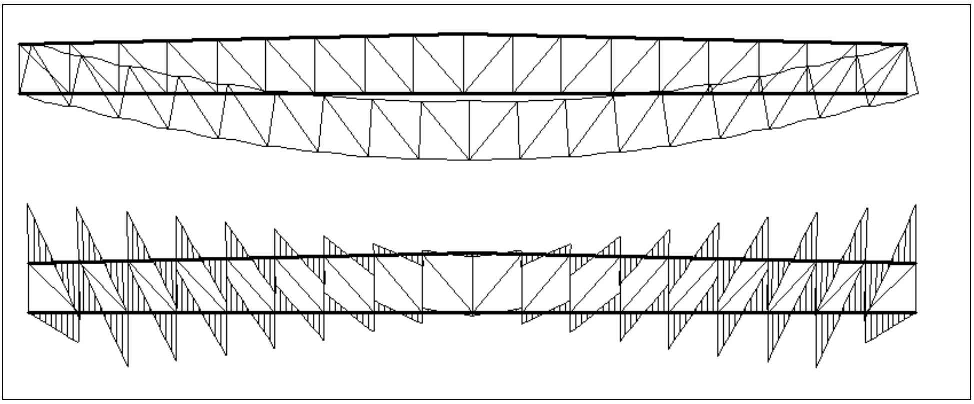


Figure 12-14: (Incorrect) global deformed configuration and (incorrect) moments due to the constraining of the nodes.

12.2.3 TOWERS AND MASTS

Towers and masts are space frame structures in which the regime of axial actions is dominant. However, it is not advisable to build these structures using only truss elements, since the staggering of the diagonal wind braces can lead to aligned hinged elements, in which the central nodes are free to translate, if all the components of unconstrained displacement are not subjected to suitable and

plausible. The person who made the model believed that in that way all of the bending strain energy in the shells was eliminated. The incorrect analysis was given to the clients, who accepted and paid for it, not being able to see the error committed. It was only after the fact that the analyses were found (and numerically proven) to be *completely incorrect*.

carefully studied schemes of constraint. It is thus preferable to use beam elements for the corner struts and the main frames, to which elements are then attached diagonal trusses, which can also be staggered.

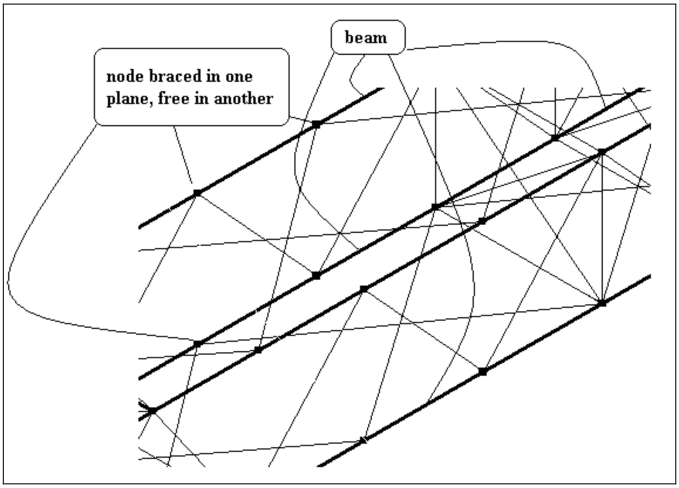


Figure 12-15: Staggered diagonals in a pylon structure.

The ribs made with beam elements receive the diagonal trusses and are not released. Their bending stiffness impedes the rigid motions of the nodes, which would instead be permitted with truss elements.

One of the most important design problems is related to the need to arrange enough elements in space to impede rigid motions or conditions of hypostaticity (i.e. mechanisms). In order to investigate whether or not conditions of hypostaticity exist (that is, if there are modes of deformation connected to insufficient strain energy, and thus "weak points" in the structure), it is useful to perform a modal analysis, which will bring to light areas where there might not be sufficient stiffness.

As we would expect, among the first modes of vibration are the local deformations of the corner struts, in which the nodes oscillate like pendulums restrained by the beam elements.

A typical case is that of torsion or the loss of the form of the pylon's section, which in order to be inactive requires a suitable arrangement of a set of diagonals.

Separate mention should be made of the discussion regarding elements that just brace the others. These can lead to meshes that are quite dense, without there being a real need for such density, because the elements that interrupt the span serve only to impede the displacement of the elements that they subdivide.

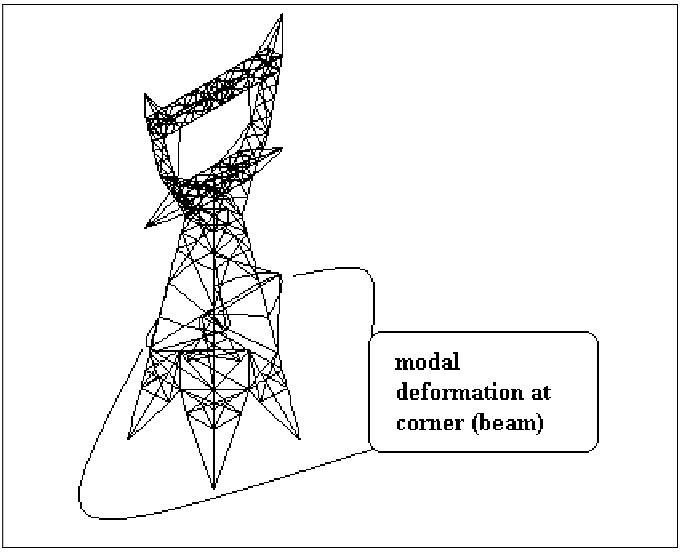


Figure 12-16: Local deforming modes revealed by a modal analysis.

12.2.4 PLANE FRAMES

Plane frames are widely used, and constitute perhaps the most frequently encountered kind of modelling problem. For clarity's sake, we should specify right away that if the frame is flat, and loaded in the plane, then there are prescribed degrees of freedom that can be constrained on all nodes; these constraint templates are shown in table 12-6.

In what follows, when we speak of "unconstrained rotation" in the node, we will be referring to the rotation about an axis perpendicular to the plane of the structure (in the three cases shown in table 12-6, respectively rotation Rz, Ry and Rx).

Because in general the releases needed are those for rotations (hinges), here by release we mean the release on a beam element of the rotation which, in the three local axes of the element, corresponds to the direction normal to the plane of the structure.

Plane of the structure	Constraint template
XY	Tz, Rx, Ry
XZ	Ty, Rx, Rz
YZ	Tx, Ry, Rz

Table 12-6: Constraint templates for plane frames according to the plane of the structure.

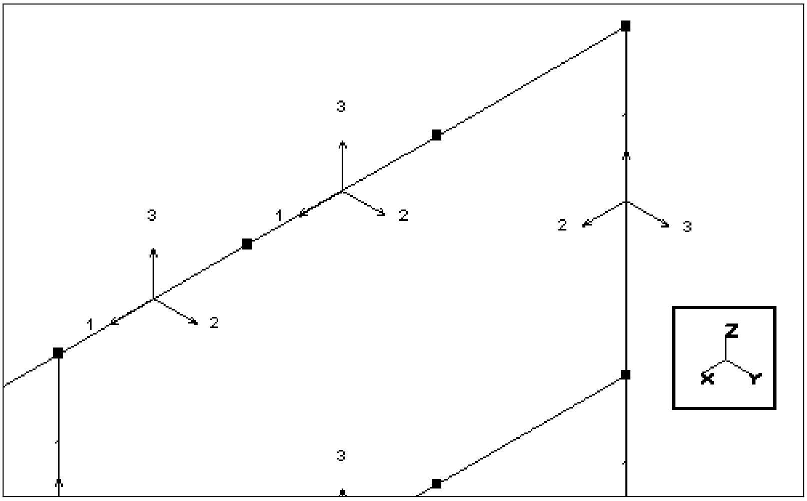


Figure 12-17: Partial axonometric view of plane frame (XY) showing the three local axes used to model the beams and columns.

For example, fig. 12-17 shows that the beam elements that model the beams have two axes normal to the plane of the frame, and thus – if necessary – we have to release R2 (the rotation about axis 2), while the beam elements used to model the column have axis 3 normal to the plane of the frame and thus – if necessary – we need to release R3 (the rotation about axis 3).

This said, fig. 12-18 shows six typical cases of connections between beams and columns, with the related finite element schemes shown to the right of them. These use the conventions of a dot next to the node to indicate a release, and a larger dot on the node to indicate a constraint on the node (not used in this particular figure, but which we will see in figures below). To make these easier to read graphically, the nodes have been shown as solid squares, and the thinner lines used to distinguish the axis line of truss elements from those of beam elements.

We can see that:

- Case A shows a rigid node between beam and column. All beam elements; no releases.
- Case B shows a continuous column and a discontinuous beam. Two end releases are needed on each horizontal element.
- Case C is the dual of case B; here the releases are placed on the vertical elements.

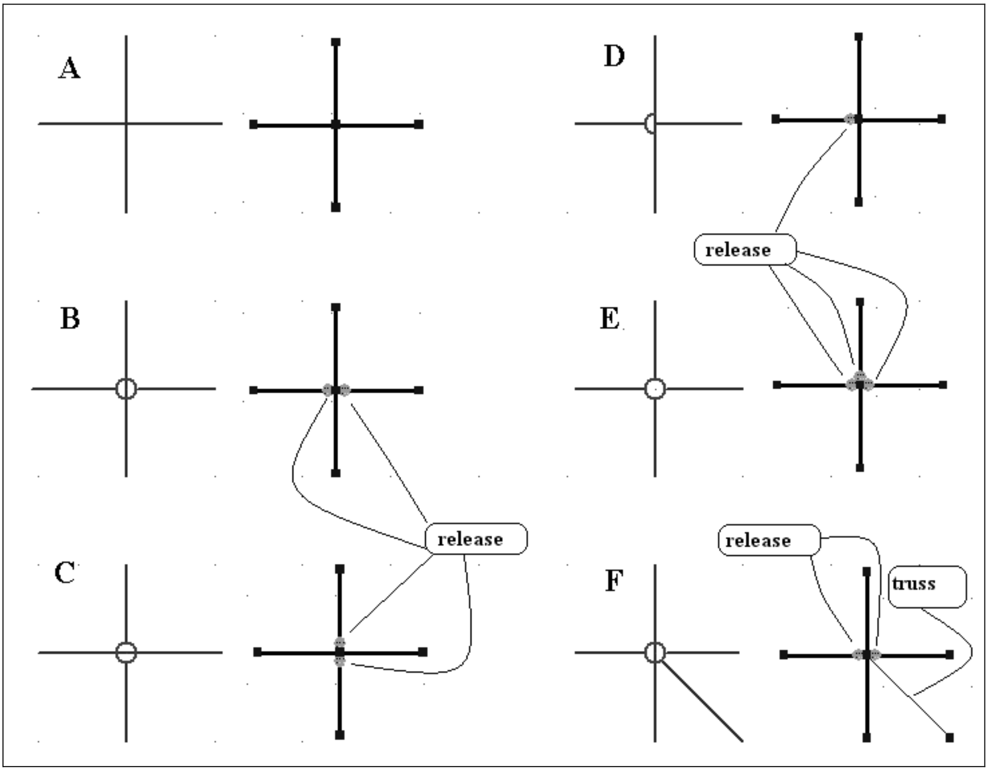


Figure 12-18: Typical beam-column connection schemes and their related finite element schemes. The dot represents a release; the thinner lines represent truss elements.

- Case D shows an end release applied to only one of the horizontal elements, the only one that is hinged.
- Case E shows a continuous hinge, but releases are placed on only three elements instead of four in order to avoid having to place a constraint on the rotation of the node. If a fourth release is placed, then the node has to be constrained for rotation.
- Case F is similar to case B, but another element is added which – since it is a diagonal wind brace – has been modelled with a truss element and thus does not need to be hinged, because a truss is hinged by definition.

Fig. 12-19 shows more examples of connection schemes that differ from those shown in fig. 12-18 only by the addition of diagonals, all modelled with truss elements. Of particular note are cases L (where the four diagonals are hinged because they are truss elements); M (where, appropriately, some elements do not have releases); and N (a variation of M).

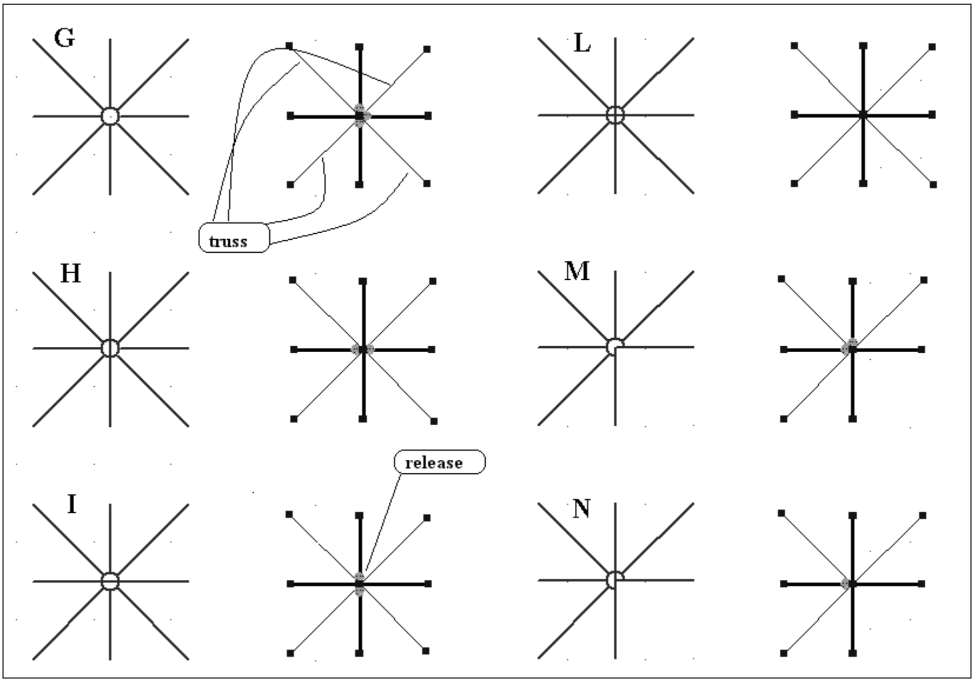


Figure 12-19: More typical cases of connection schemes and the related finite element schemes. Dots represent releases; squares, nodes; the thinner lines, truss elements.

Case G is a case of a completely continuous hinge. As we can see, only three of the four beam elements are released, while the fourth is not, in order to avoid having to place a constraint on the rotation of the node. If the fourth release is placed, then the rotation of the node has to be constrained.

Fig. 12-20 shows cases where a column with diagonal wind bracing arrives to the ground:

In case O the nodes is completely fixed, but there is a hinge because the diagonal is modelled with a truss element.

Case P shows an analogous situation, but there are two (truss) elements.

Fig. 12-21 shows cases in which, rather than being fixed, the column is hinged at the ground. In this case there are two possible ways to model it. In the first (shown in the middle column of schemes), the rotation of the node is left unconstrained and the beam element used to model the column does not have any releases. In the second (shown in the column on the right), the element is released for rotation, *but in this case the rotation of the node (which is no longer associated* *to any stiffness and is thus completely fictitious) has to be constrained.* Case Q shows the case of a single diagonal; case R a double diagonal.

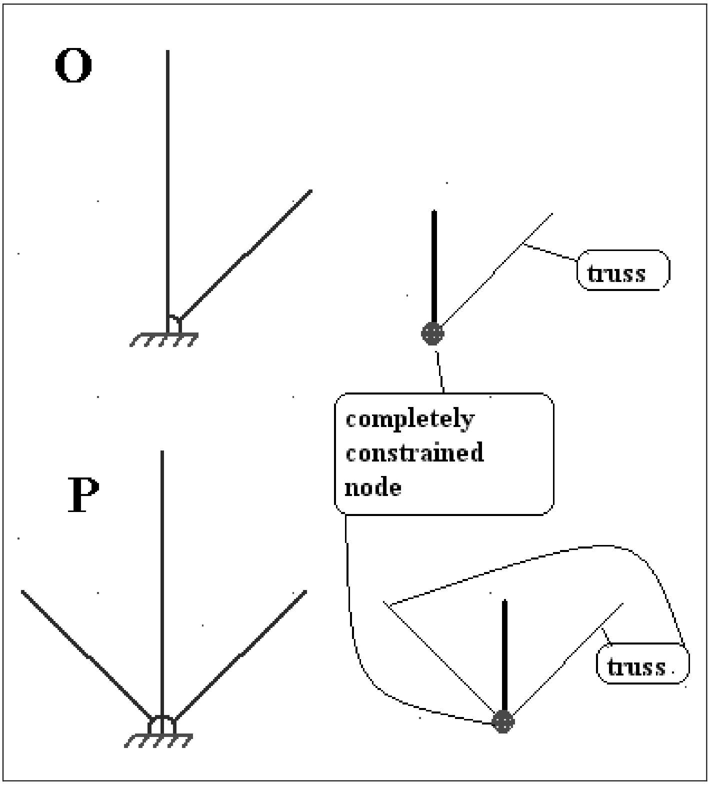


Figure 12-20: Possible models of a fixed column with diagonal wind braces.

When working with models of plane frames, where we normally find beam elements as well as truss elements, it is also possible to encounter nodes to which only truss elements are attached (typically in the wind braces). In this case these nodes have to be constrained for the three rotations and the translation in the direction normal to the plane of the frame. Note that this constraint corresponds to that of the constraint mask, with the addition of the rotation of the node with respect to the axis normal to the plane of the frame. The constraint mask can in any case be omitted when the stiffness of the beam elements constitutes a restraint on the act of motion of the nodes in the direction corresponding to the constraint template. However, in such cases the nodes where there are only truss elements attached must be constrained for all of the rotations and the out-of-plane translation.

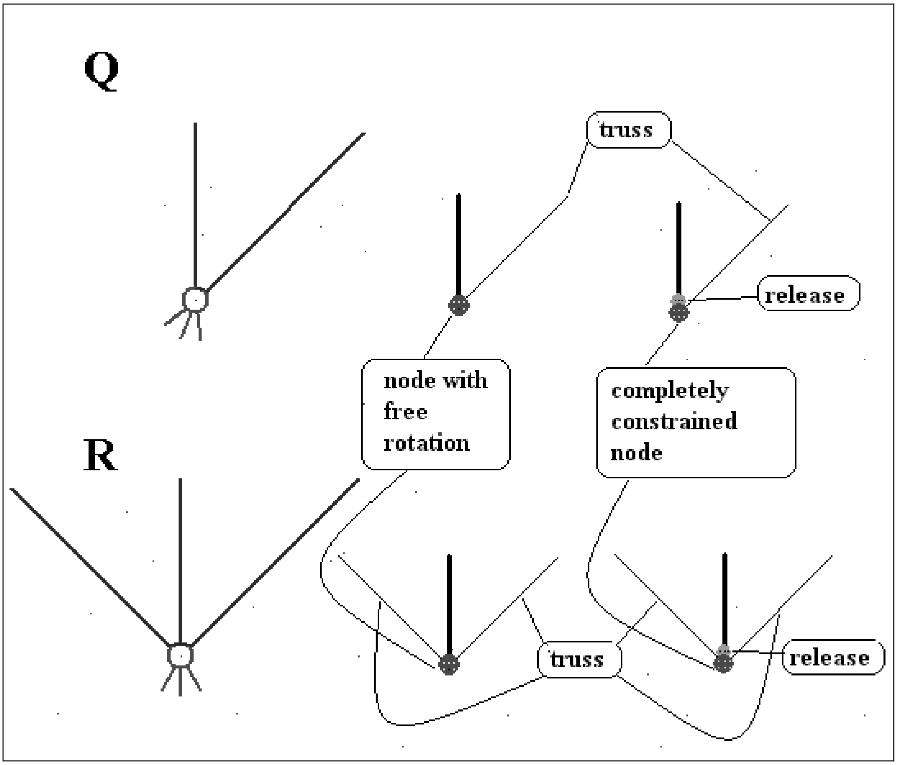


Figure 12-21: Possible models of a hinged column with diagonal wind braces.

For example, fig. 12-22 shows a simple plane frame with wind braces, defined in plane XZ. If node 7 and node 10 are constrained with the constraint template TyRxRz, then it is not necessary to apply the constraint template to nodes 8, 11 and 9, because the bending and torsional stiffnesses of the beam elements are such that translation Ty and rotations Rx and Rz of these nodes are not unconstrained (they are associated to a stiffness).

Now let's look at nodes 12 and 13. There are only truss elements attached to these, and thus translation Ty and the three rotations Rx, Ry and Rz are inevitably unconstrained and have to be constrained in any case, because if they are not, this will lead to a mechanism. Translations Ty are unconstrained because the trusses have no bending stiffness and thus, because of the out-of-plane translation of nodes 12 and 13, they behave like a mechanism.

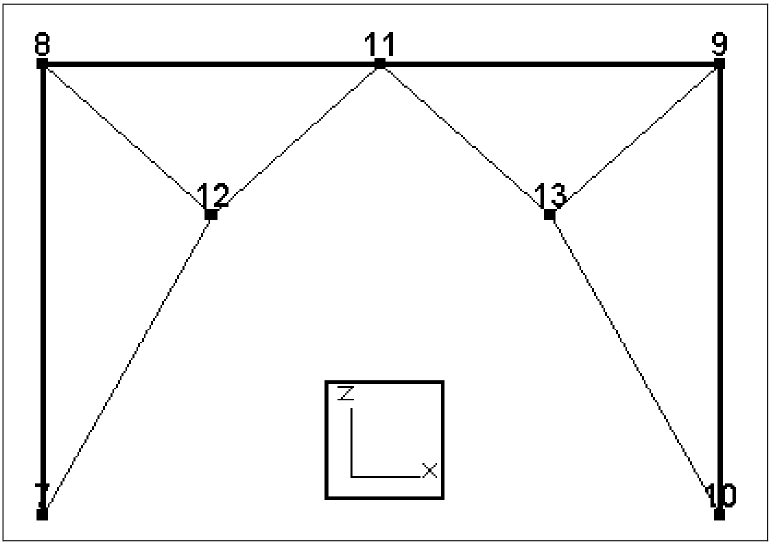


Figure 12-22: Elementary plane frame with wind bracing trusses.

♣

Now that we have discussed the problems relative to constraining the nodes and releasing the elements, we should mention the problem of how to model the loads. Doubts often arise about how many finite elements to place, and whether or not to add a node under a concentrated load.

The correct procedure to adopt generally depends on the software program used. If the local effects of the loads are re-added to the beam elements, then in terms of stress it is absolutely possible to avoid having to put a node under a concentrated load. The particular characteristics of the beam element make it possible to arrive at precise results even without increasing the discretization too much, as long as the local effects are re-added. If instead, in keeping with the formal orthodoxy of the FEM, the effects of local loads are not re-added, then it is absolutely necessary to subdivide the structural element sufficiently by means of an appropriate number of finite elements.

For example, fig. 12-23 shows two portals modelled using either one or two elements for the horizontal. Since the software program used (in this case, SARGON) re-adds the local effects of the stresses on the beam elements, the results are exact in both cases. Each user should verify what his or her own software program does: if it does not re-add the effects of the stresses, then the discretization has to be sufficiently dense.

As far as displacements are concerned, the situation is more subtle and more complex.

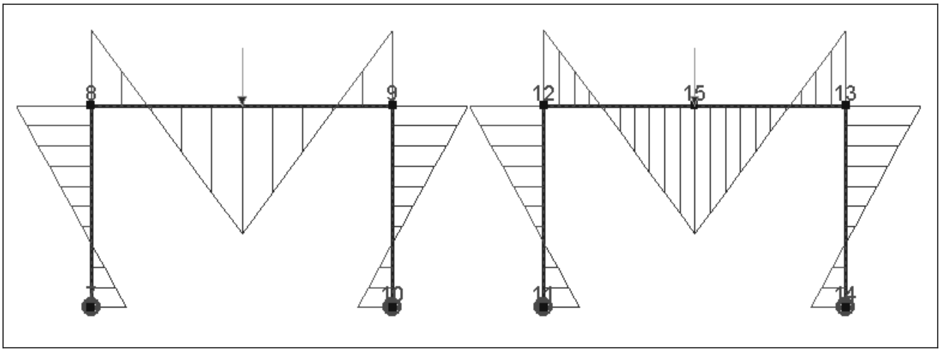


Figure 12-23: Moment diagrams of two frames modelled in different ways: the results are equal (modelling performed with SARGON).

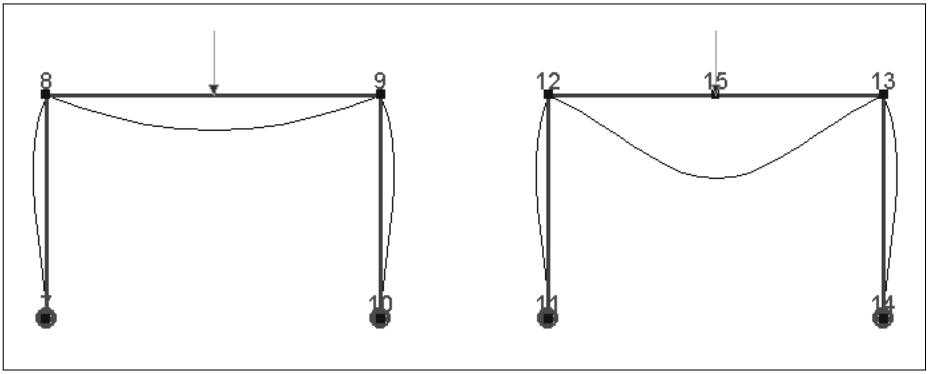


Figure 12-24: Deformed configurations of two frames modelled in different ways. Here the results are different: on the left there is a single cubic (the shape function of the single finite element); on the right there are two cubics (those of the two finite elements) and the solution is rigorously exact (modelling performed with SARGON).

Most programs limit themselves to giving the displacements of the nodes, and thus if we want to know the deflection at the centre point it is necessary to subdivide the elements. In addition to giving the displacements of the nodes, SARGON also gives the displacements within the beam elements, where there are no nodes. In drawing the deformed configuration, it is very much preferable to use a cubic for the deformed configuration rather than to linearise it between the nodes as some programs do.

Is the displacement within the elements exact or not?

If "normal" cubic shape functions are used, if there are no effects of shear, then the displacement is exact as long as there are no distributed loads. If there are distributed loads (that is, in the terminology used in the FEM, *internal loads*) then the displacement, even when there aren't any effects of shear, is not exact, unless the software program re-adds the local effects of displacement on the elements due to the internal loads, determined keeping the nodes fixed.

Fig. 12-24 shows that the deformed configurations are noticeably different for the two cases discussed, since in order to render the deformed configuration, SARGON pre-release 9.0 limited itself to using the shape function of the element without re-adding the deforming local effects on the element itself[2]. Thus the case on the right is exact (two cubics), while the case on the left is only approximate (a single cubic for the displacement). Note that, instead, the displacements of the columns (where there are no distributed loads) are identical in both cases.

Up to this point we have discussed what happens when there are no effects due to shear. When such effects are present, then the displacement is no longer "exact" except at the ends of the elements (cf. chap. 11), unless the software program used adopts modified shape functions (because the rotation of the node is not entirely due to w' but also to α; see chap. 11, §11.3.2.6). Practically speaking, if we create a scheme for the portal shown in fig. 12-24 taking into account the effects of shear, and compare the two models (fig. 12-25), one made with two finite elements for the horizontal and one made using forty finite elements for that same horizontal, we can see that in correspondence to the common nodes, the displacements are identical (at the centre and at the ends, in particular), *while they are different at the intermediate points* (although the differences are small, and in any case in a measure that is inversely proportional to the slenderness of the element).

It seems therefore reasonable to make the following remarks (when using SARGON or a similar software program):

- If we are only interested in the diagrams of internal actions, it is not necessary to increase the discretization of the structural elements like beams.
- If we are interested in the displacement of the elements where there are no distributed loads, it is possible to use the commands that determine internal displacements, because we are certain that there is one "exact" displacement when there are no effects of shear, or one very close to the "exact" one with slender beams even when there are effects of shear.
- If we are interested in the displacements and there are internal loads in the elements (whether concentrated or not), then it is useful to increase the discretization in proportion to the kind of actions present. In general,

[2] From version 9.0 onwards Sargon also re-adds the effects of local displacements.

one node is added under a concentrated load, or a few elements (for example, four) are added for a distributed load.

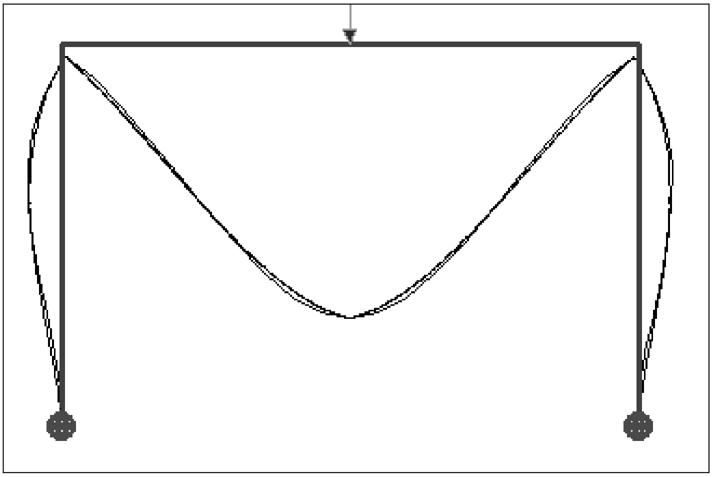


Figure 12-25: Effects related to shear: the difference in the deformed configurations when 2 finite elements or 40 finite elements are used to model the horizontal. When there are no effects of shear, the two deformed configurations coincide, since two cubics already give the exact displacement.

12.2.5 SPACE FRAMES

The problems of space frames are essentially related to the three-dimensionality of the mesh and the possible movements. The constraint of the nodes or the release of the elements can no longer be considered in the context of a plane, and there are no longer "constraint masks" that can make the job easier.

However, it is also true that once an understanding is reached of how the FEM works and what the basic characteristics are of each kind of finite element, then what has to be done in a three-dimensional context becomes clear rather spontaneously.

As far as the constraints of the nodes are concerned, we must remember that these are subject to three translations and three rotations with regards to the global coordinate axes, and thus it is necessary to determine which stiffnesses limit these movements in order to understand whether or not there are any conditions of pseudo-mechanism.

If the model shows deformed configurations that are different from what was expected, then it is absolutely incorrect to add constraints indiscriminately in order to bring the deformed configuration back into a qualitatively acceptable form (an error that is actually committed with some frequency): adding constraints in this way can lead to the distortion of the stresses and the assignment of significant constraint reactions at inexistent points "on the ground".

As far as the releases of the beam elements are concerned, it is of the utmost importance that the orientation of the three local axes be taken sufficiently into account, because the release is defined with regards to the local axes, not the global axes. We have already mentioned this question in connection to planar problems, where there is an analogous problem.

In the study of space frames it is often possible to visualise how the structure behaves as the meeting of two complementary behaviours: that of plane XZ and that of plane YZ. In other cases, the complexity of the scheme prevents this from being possible, and despite checks performed on the model, the response is more difficult to interpret.

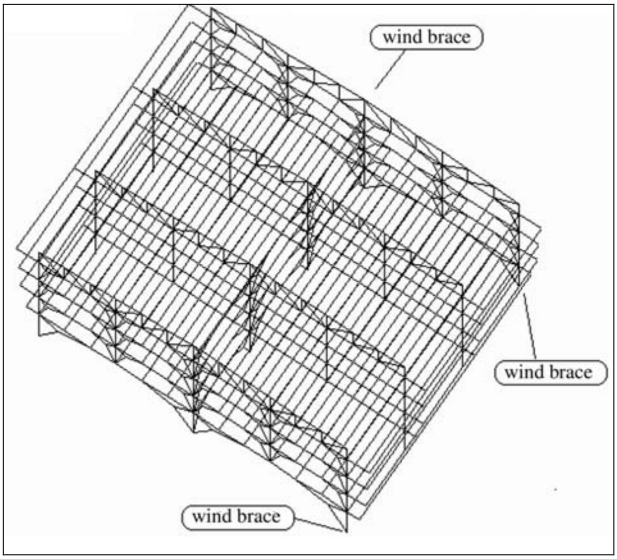


Figure 12-26: Example of the 3D model of a structure in steel with bracing arranged in an H-shape in plan.

In 3D modelling it can happen that we encounter local situations that require fictitious constraints that act to impede pseudo-mechanism. Fig. 12-26 shows a three-dimensional model of a steel structure with bracing arranged in an H-shape in plan, and fig. 12-27 shows details of this structure, with constraints and releases.

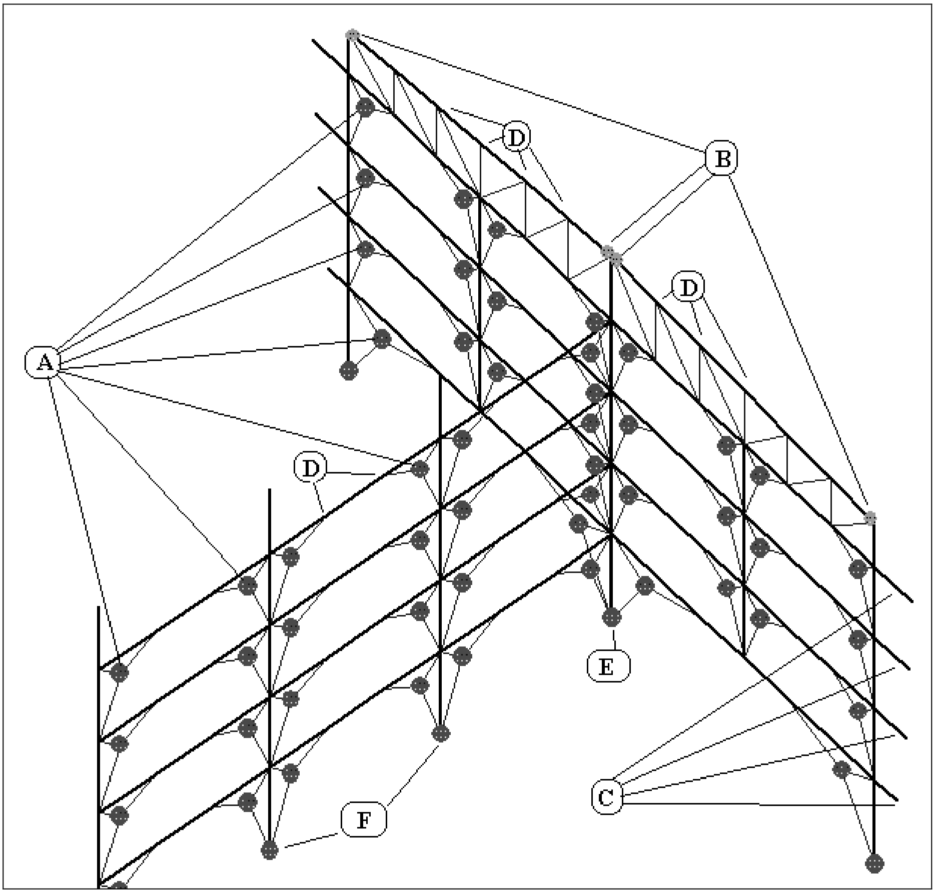


Figure 12-27: Details of the model shown in fig. 12-26 showing constraints and releases.

For example, detail A in fig. 12-27 shows nodes of the bracing to which only truss elements are connected. In order to avoid conditions of pseudo-mechanism, all three of the rotations and the translations normal to the plane of the braces (for some these are translations Y; for other, translations X) have been constrained. Here the non-expert might object that placing constraints on the translation constitutes an unacceptable restraint on the structure. In reality, however, the structure is completely free to move, because it is not held back by these constraints. In fact, the trusses that are attached to those nodes constitute a mechanism that leaves the node totally free to translate.

Restraining it with a constraint does not imply any constraint on the remaining parts of the structure, which is free to translate with respect to these nodes.

Detail D in fig. 12-27 shows nodes where this constraint is unnecessary, since the chords of the reticular girders have been modelled with beam elements rather than with truss elements.

Other interesting details that appear in fig. 12-27 are:

- The addition of hinges (releases) at the ends of the chords when these meet the columns (detail B);
- The cantilever elements that project beyond the columns (detail C: no end release is applied to these projecting elements);

The full constraint of the column at the base that receives all of the braces located either in space (detail E) or in the plane (detail F). The nodes have to be constrained without mixing the rules that we have introduced.

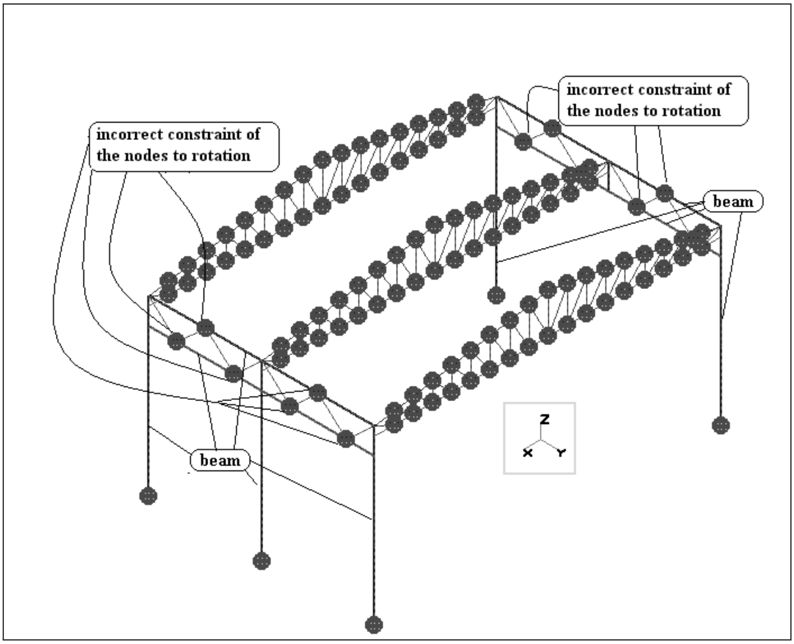

incorrect constraint of the nodes to rotation
incorrect constraint of the nodes to rotation
beam
beam
Z
X
Y

Figure 12-28: 3D model showing incorrect constraint of the nodes. The trusses are modelled with truss elements; the reticular girders with parallel chords are modelled with a combination of beam elements for the chords and truss elements for the sides.

For example, fig. 12-28 shows a three-dimensional model in which all of the nodes of the trusses that are modelled with truss elements are – correctly – constrained for translation Y and all three of the rotations. But the nodes of the reticular girders with parallel chords are incorrectly constrained. By constraining these nodes (attached to the beam elements) for rotation, a non-existent constraint is introduced on the bending of the chord (see instead detail D of fig. 12-27). Inexistent reactive couples are transferred to the "ground", and the model is incorrect.

Fig. 12-29 shows the out-of-plane moment diagrams for the chords.

In this case (which was actually encountered in practice), even if the analyst did not thoroughly understand how to constrain the nodes, he should have been aware of the significant reactive couples on the nodes constrained for rotation; their very existence should have made him wonder as to the cause.

Thus, as we can see, when we are working in a three-dimensional context different problems arise than those we have described and discussed in situations of two dimensions.

In a three-dimensional context the position of the three local axes of the beam elements is obviously of much greater importance, and the orientation of all of these should be systematically verified before the analysis is performed. If the beam elements are incorrectly oriented, this can lead to a non-planar response where a planar one is expected instead. For example, fig. 12-30 shows what happens to a simple portal if the horizontal is incorrectly oriented.

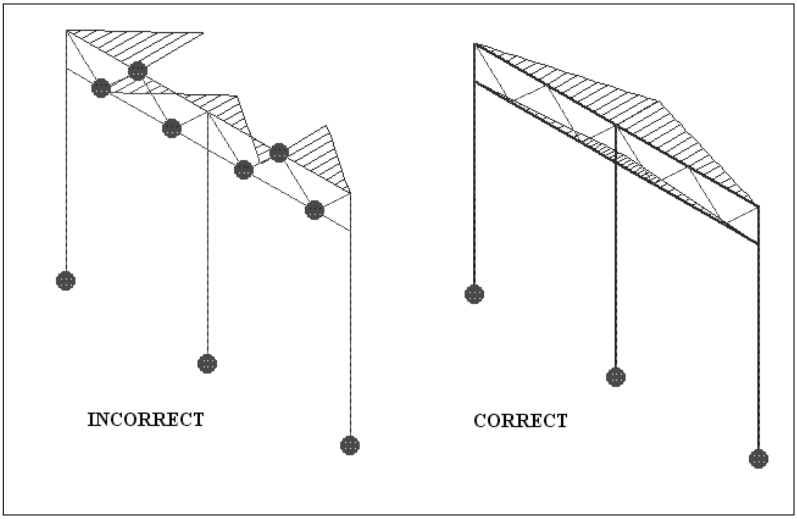
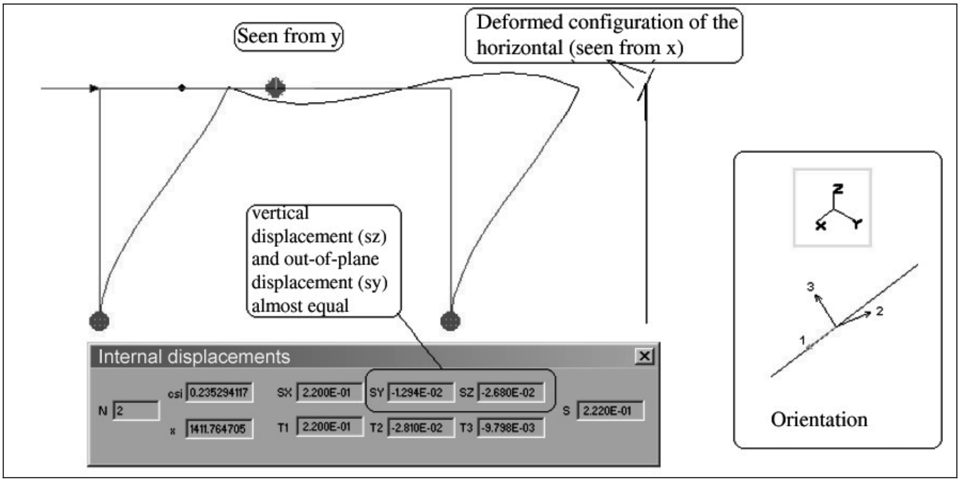


Figure 12-29: The effects of incorrect constraint: out-of-plane moment diagrams of the chords with and without the incorrect added constraints.


Seen from y
Deformed configuration of the horizontal (seen from x)
vertical displacement (sz) and out-of-plane displacement (sy) almost equal
Z
X
Y
3
2
1
Orientation

Internal displacements

N 2 csi 0.235294117 x 1411.764705

SX 2.200E-01 SY -1.294E-02 SZ -2.680E-02

T1 2.200E-01 T2 -2.810E-02 T3 -9.798E-03

S 2.220E-01

Figure 12-30: Non-planar deformed configuration of an incorrectly oriented portal.

12.2.6 MASONRY STRUCTURES

Masonry structures are often modelled with membrane elements, and thus have no out of plane stiffness. Typical problems concern how to make the correct adjustments for material values (normal and shear moduli of elasticity, Poisson's ratio) bearing in mind that masonry is anything but elastic, homogeneous and isotropic. Orthotropic modelling, which takes into account the different behaviour of masonry in two directions, should also be considered. Models made by representing each masonry component as a membrane in plane stress can give the expected order of magnitude of stresses and their qualitative tendencies, making evident the areas that require special reinforcement.

For example, fig. 12-31 shows a model of a stone wall 40 cm thick. The example is taken from an actual structure in Italy's Friuli region, and the analysis was undertaken to help solve problems that arose after *an opening was made under the ridge beams of the roof* inserting then a short beam in reinforced concrete *which was not designed by calculation but was "eyeballed"* and which was intended to transfer the reactions of the ridge beam of the roof to the load bearing wall. The state of affairs is shown in the photograph in fig. 12-32; the column that appears there is not a true column, but a false column in hollow brick[3]).

[3] This is just one unfortunate example of how inadequate actual construction is today: the entire load of the roof depended on that beam, and this was in a severe seismic area. It was not possible to know exactly what reinforcing was used in the beam built by "eyeballing", nor were any relative calculations found. The building's owners finally decided to leave the beam as it was without inserting any column and without filling the opening,

Plate-shell elements were adopted for the model only because these made it possible to avoid constraining the inactive components of the movements of the nodes (rotations and out-of-plane translations). In fact, given the kind of loads, only membrane behaviour was brought into play. The possible new column was modelled with a beam element that was released at its ends (and is thus hinged), to avoid a connection with a fictitious stiffness being added to the plate elements at the drilling DOF. Had this not been the case, the drilling factor would have to have been made null, and no releases placed, but the rotations of all of the nodes would have had to have been constrained about an axis perpendicular to the plane of the wall.

The problem here was to study the effects that the possible addition of a (real) column would have had on the walls, bearing in mind that the column had to take the loads coming from the ridge beam of the roof and distribute them to the load bearing wall. Given that openings in the wall had already been made in several places, the problem was to come up with an estimate of the values for the stress following the intervention.

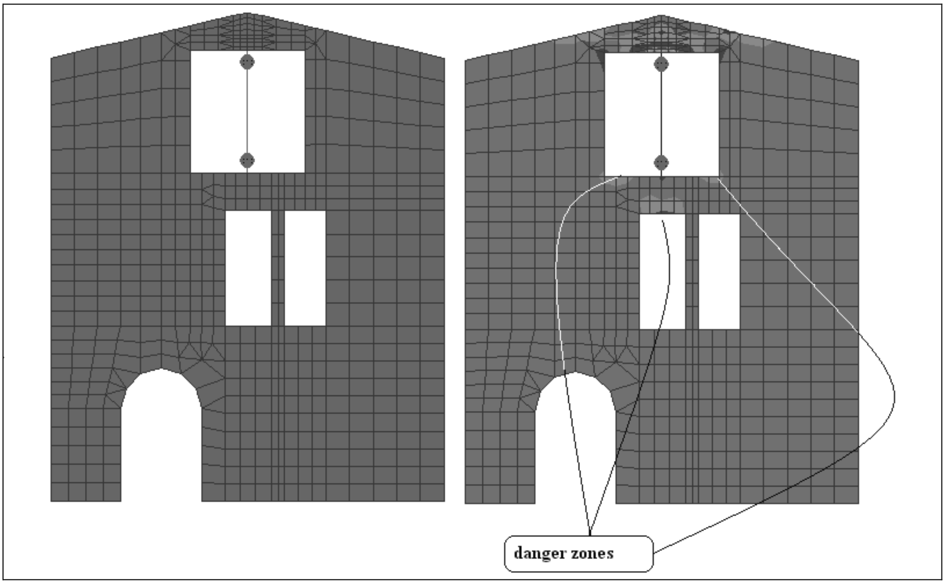


Figure 12-31: Model of a masonry wall: 4-node isoparametric elements and CST.

trusting the solution built by the construction firm and the report of a design-builder who, when queried, asserted that the beam was amply capable of sustaining the load and was fine as it was.

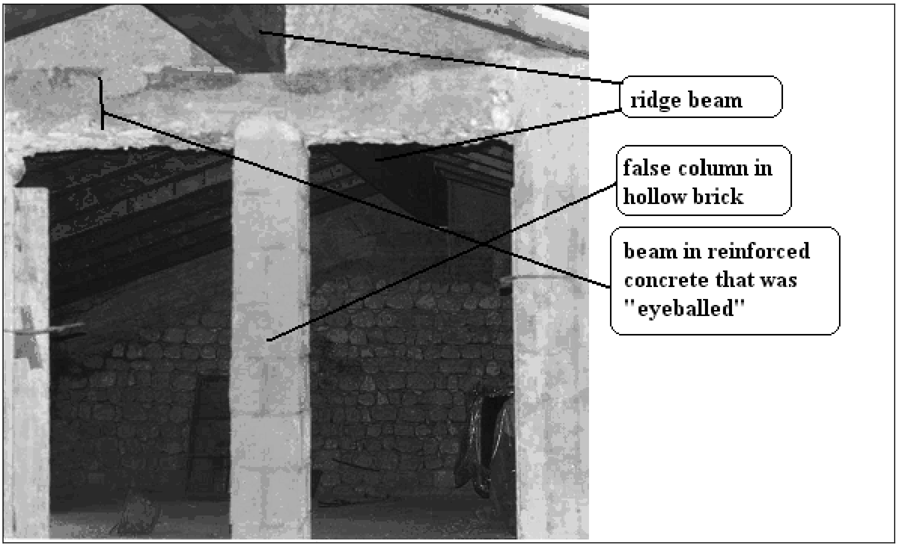

ridge beam
false column in hollow brick
beam in reinforced concrete that was "eyeballed"

Figure 12-32: The problem that led to the analysis shown in fig. 12-31.

The first model was made to provide a preliminary set of data to be used as a starting point. This was followed by a second, more refined model aimed at capturing in greater detail the aspects relative to the areas in which it was feared that significant primary tensile stresses would arise (for masonry "significant" can indicate a tensile stress equal to 0.2N/mm^2).

An examination of the model shows that a first significant simplification regards the column, which is modelled as transmitting its axial action at a point rather than across a given width. This gave rise to local effects that were quite evident in both models. Had the model been more refined, it would probably have modelled that column as having a certain bulk, in order to distribute the forces over a wider area.

The model shows concentrations of forces at the corners that cannot be realistic, given the certain redistribution as a consequence of microfractures in the area of the corners.

The masonry reacts to tension also when there are significant values of principal stress, and this is not exact, except when only negligible final values of tension, compatible with the limits of the masonry, are the result of the combination of the effects related to the various load cases.

A qualitative examination of this case showed that the column would have created problems in the architrave of the floor below: the *order of magnitude* (no greater precision could be hoped for) of the principal tensile tresses were found to be equal to 0.15N/mm^2, just about equal to the masonry's expected tensile resistance. The lack of sufficient margins of safety (given the amount of uncertainty in the calculations) would absolutely have made it advisable to insert a tension rod at the floor below (above the doors of the lower floor), in order to help carry the tensile stresses that the masonry alone could not carry.

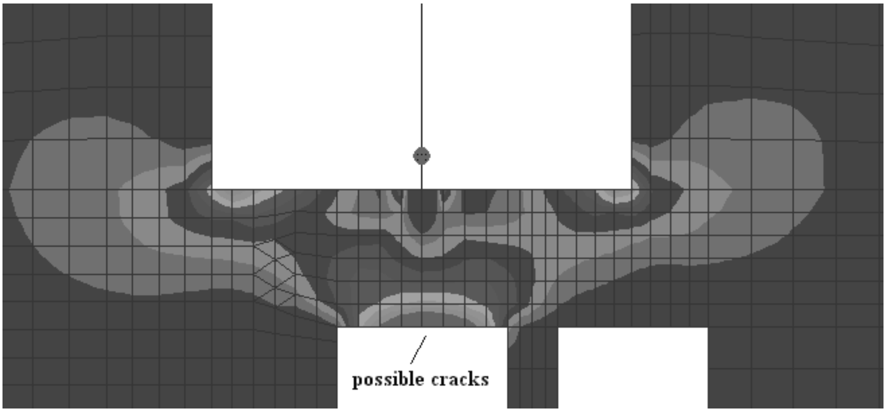


Figure 12-33: Examination of a detail of a somewhat refined model. Maximum principal stress.

12.2.7 PLATES

We have seen that in the modelling of plates (cf. chap. 11, § 7) the discretization plays a fundamental role, given the intrinsic difficulties of these elements in modelling the solving equations, which – we recall – are fourth-order partial differential equations. It therefore makes no sense to think that it is possible to model plates with just a few elements per side, especially when there are openings or other discontinuities. In modelling plates it appears to be indispensable to go through at least two cycles with successively finer meshes, because this makes it possible to determine what differences a finer mesh makes from one run to the next. Without this kind of painstaking study, the validity of the results of an analysis cannot be taken for granted unless a very, very dense discretization has been put into place from the very beginning.

If the plate is parallel to a coordinate plane it might be opportune to constrain the degrees of freedom corresponding to the rotations with respect to an axis normal to the plane of the plate, once it has been established that the finite element adopted does not use that degree of freedom. This makes it possible to save one out of the six degrees of freedom, thus creating, all other conditions being equal, a more refined model.

When there are concentrated loads, it must be borne in mind that there is usually a singularity, and thus the stresses should be read at a sufficient distance from the exact point where the forces are applied (as the discretization is increased the stress in the point that is directly loaded will tend to increase indefinitely).

Where the formulation does not make it possible to determine the out-of-plane shears, these should be determined with other systems (including engineering systems) in order to make sure that they do not constitute a hazard.

If the ratio $(Et^2)/(GL^2)$ is greater by a certain percentage point, then the possibility of studying the plate using Mindlin's theory should be considered: in this case solvers equipped with finite elements that are suited to that purpose must be used.

12.2.8 FLOOR SLABS

12.2.8.1 FLOOR SLABS MODELLED EXCLUSIVELY USING MULTI POINT CONSTRAINT (MPC)

When modelling slabs the kind of behaviour that we want to model must be kept firmly in mind, whether purely membranal or including bending. A first way to model floor slabs takes into account the effect they have of fixing the plane structure, such that the plane tends to maintain its shape during deformation. Practically speaking, slabs modelled in this way are comparable to a multipoint constraint (MPC): we are not so much interested in precisely modelling the details of the slab as we are in giving the plane a shape that remains the same during the course of deformation.

If this is our objective, then the best way of modelling the slab in question is to use membrane elements, without worrying too much about the density of the mesh (see, for example, the model shown in fig. 12-34) and thus possible locking. An overestimate of the membrane stiffness of the plane does not have any significant consequences, given that a considerable amount of stiffness is already present normally, nor is there usually any reason to study the stress field within the slab itself.

Note that the use of the membrane elements to model the floor slab means that the slab itself does not contribute in any way to resisting bending, which falls solely on the beams and, by continuity (if this exists), on the columns.

This kind of modelling is by far the most frequently used, because the bending effects of the floor slab (which might include small ribs, or an uncoupled corrugated sheet) are usually determined separately with local schemes (isostatic, or at best with continuous beams).

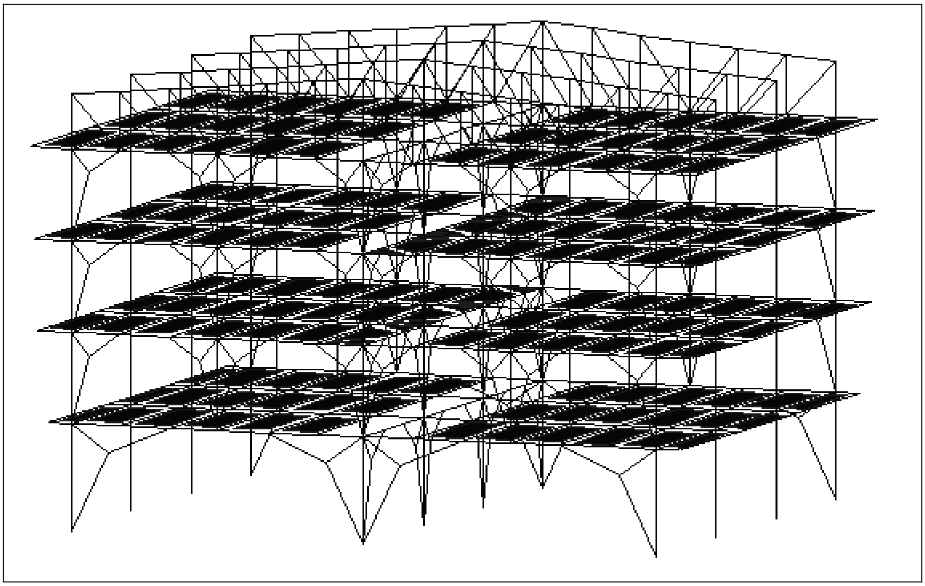


Figure 12-34: Building with slabs modelled using membrane elements. The mesh is not fine.

12.2.8.2 Solid slabs

When it is desirable to take into account the bending stiffness of the floor slab, because for example it is desirable to keep track of its contribution to the beams and to the building as a whole, then plate-shell elements should be used instead of membrane elements.

In the case of solid slabs, homogeneous and isotropic, this involves adding plate-shell elements with the desired thickness. Here the discretization used to model the slab has to be quite dense (similar to that of plates), because this prevents the onset of unwanted effects of locking, and also makes it possible to evaluate the stresses.

If the slab is solid but is built with a binary system (for example, using a corrugated sheet with poured fill), then the equivalent thickness is generally determined by considering the concrete and the steel to be uniform. If "equivalent" models are used, then the level of calculated stress is not significant, but should be corrected to take into account the equivalence that was introduced.

12.2.8.3 RIBBED SLABS

In the case of ribbed slabs, actually quite frequently used, the problem lies in determining the stiffness of the beams to be used in conjunction with the plate-shell elements. The precise problem is often how to evaluate the effective width of the plate that works with the web and flange of the beam in order to arrive at the equivalent beam to use in the calculations.

Alternately (although models of this kind are generally used only for a floor slab and not for an entire building), the ribs are also modelled with plate-shell elements, because this makes it possible to keep track of the actual effects of how the various parts of the structure work together.

12.2.9 BUILDINGS WITH A BRACING CORE OR SHEAR WALLS IN REINFORCED CONCRETE

In buildings that have a bracing core or shear walls in reinforced concrete to resist horizontal forces, the problem is what kind of elements and discretization to use in modelling them.

Generally speaking the stresses in these cores are not directly studied, because cores are designed without the help of a finite element model. This might lead to thinking that it is possible to do without an extremely fine discretization, but this is not generally the case. In fact, insufficient discretization can lead to an overestimation of the core's stiffness (due to the locking of the elements; see chap. 11 for a discussion of this), underestimating the horizontal displacements and thus leading to an analysis that is unreliable with regards to the safety of both the internal forces of the columns, and of the horizontal displacements.

The correct discretization would therefore seem to be the one that provides a relatively fine mesh for the cores and the shear walls, with at least four to five elements for each story. Thus a reasonable dimension for the elements could be about 0.5 x 0.5 m^2 (except in specific cases, or cases with particular elements). It is in any case of the utmost importance to perform an analysis after having doubled the size of the mesh at least for these elements (saving right connectivities) to see what differences this makes in the results obtained, in order to keep the locking effects under control.

For example, fig. 12-35 shows the map of the horizontal displacement for a building with its core modelled with only a few (insufficient) plate elements (type QUAD4 bilinear isoparametric for the membrane part, 4DKT for the bending part). The displacement is equal to 8.74 mm. Fig. 12-36 shows what happens if the elements are made smaller. The displacement grows to 11.7 mm, an increase of 33%. The behaviour of the QUAD4 membrane elements (isoparametric with bilinear displacement) are to blame for this excessive stiffness.

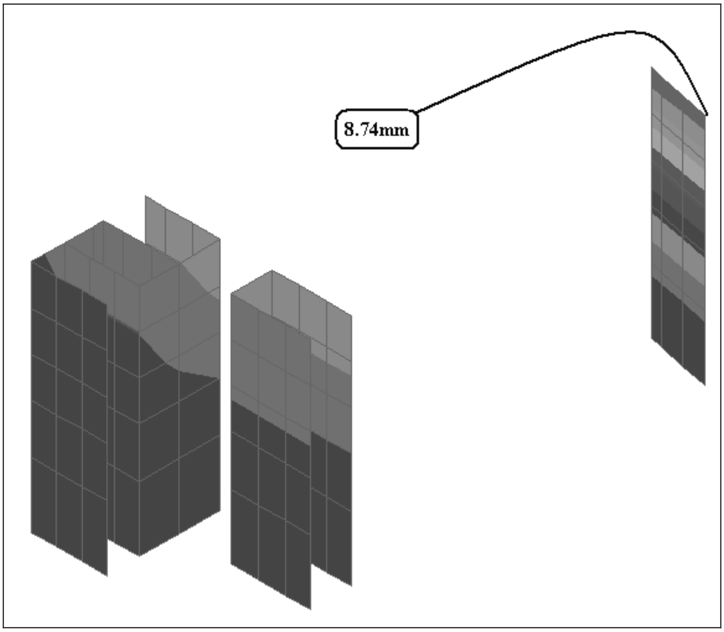


Figure 12-35. False colour map of the deformed configuration due to earthquake Y. Translation Y. Discretization with one element per storey, 4-node bilinear isoparametric elements (QUAD4), full 2x2 Gaussian integration for the membrane part, 4DKT elements for the bending part.

If different finite elements are used (especially for the membrane part, which here plays a role in overall bending because of how the bracing core is made) better results are obtained, but not so good as to permit modelling like that shown in fig. 12-35. For example, using incompatible elements for the membrane part (and 4DKT elements for the bending part), results in a displacement that is equal to 10.44 mm for the case shown in fig. 12-35 and 12.51 mm for that shown in fig. 12-36.

Note that a model made in the hypothesis that the membrane elements do not have any stiffness associated with drilling DOF generally cannot be converted automatically, identically transferring constraints and releases, to a solver that uses elements that offer the possibility of this stiffness (for example, Sap90® and Sap2000®). In fact, these added stiffnesses can unintentionally stiffen the model unless suitable release codes are provided for the beam elements that are connected to the two-dimensional elements. For example, when the "simple" model is converted automatically for Sap90® without any further modifications, the resulting displacement is only 7.043 mm, which can surely be attributed to the additional stiffness just mentioned and not to the way the elements are formulated: the model has to be modified to torsionally release, for example, all beam elements perpendicular to plate or membrane elements. Moreover, it appears that the addition of this kind of release when elements with drilling DOF are used is necessary in order to avoid having connections which often do not exist in the physical reality.

The core can be modelled entirely with membrane elements, disregarding the bending stiffness appertaining to the walls and thus with an added margin of safety. A model made in this way can, however, possibly require some adjustments regarding the connections of the beams and cores (presuming that the elements do not have any stiffness associated with drilling DOF).

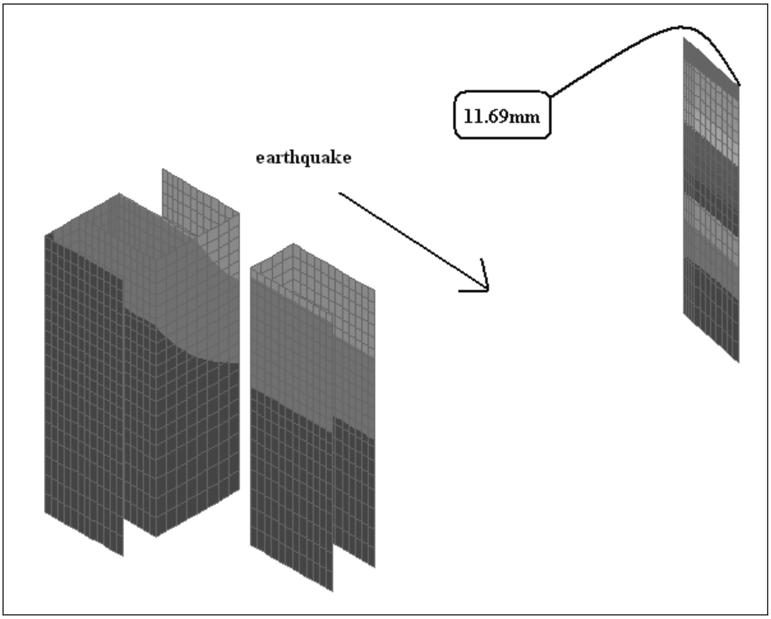


Figure 12-36: False colour map of the deformed configuration due to earthquake Y. Translation Y. Improved modelling for the bracing core (4 elements per storey, 4-node bilinear isoparametric elements (QUAD4), full 2x2 Gaussian integration for the membrane part, 4DKT elements for the bending part.

If the beams are hinged we will see the effects on the nodes of the membrane. If these have all been constrained for rotation (in order to prevent there being any pseudo-mechanism on the nodes of the membrane), then releases must be introduced. If instead the connecting nodes are all free, then both the releases and the constraints are superfluous: the releases because the beam is already hinged with respect to the membrane, which does not have any bending stiffness; the constraints because, since there is also a beam element attached to the node, there cannot be any pseudo-mechanism.

The case of a rigid connection is more subtle. In fact, it isn't possible to "fix" a beam on a membrane element normal to it, because the membrane has no bending stiffness. Thus, we either have to use "tricks", or we have to go to plate-shell elements for modelling the cores.

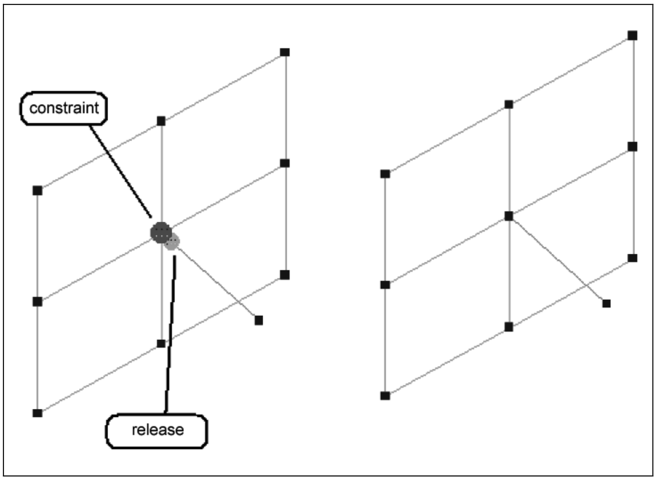


Figure 12-37: Hinged beams on walls modelled with membrane elements.

12.2.10 PLINTHS

In finite element modelling, plinths are not modelled per se, but according to the kind of restraint that they provide for the column above. If we disregard the interaction between the soil and the structure, these plinths can be considered as rigid joints, or even, when there are particular kinds of connections, as hinges. If instead we wish to investigate the interaction between the soil and the structure, then to the node that corresponds to the point where the column meets the plinth, translational and rotational springs are applied, whose characteristics of stiffness are determined as a function of the shape of the plinth and the kind of terrain.

12.2.11 REVERSE BEAMS

Even though beam elements with corrections and the additions of the local effects of distributed loads are quite precise, when the beam is on an elastic soil it must always be appropriately subdivided, in order to make correct modelling possible (this is true both when concentrated springs are used, as well as when beam elements are used on elastic soil).

In this case a preliminary study regarding the behaviour of the beam apart from its structural context is a good idea, because it is helpful for determining the correct discretization to use. In totally preliminary terms we can consider a dozen or so elements in each bay as the theoretical model of reference (cf. the test case performed in § 7.2.3.2). Where there is no foundation slab, it is relatively simple to increase the discretization and see – even after a first run – how much the solutions differ according to the discretization used.

Given that the Winkler foundation is a monolateral constraint, it is a good idea to make sure that there are no springs in tensile stress, because if there are, the analysis is no longer reliable.

12.2.12 FOUNDATION SLABS

Foundation slabs sum up all of the careful handling required to deal with plates using the typical treatments used for elastic continuum analyses of soil. The discretization of the foundation slab must therefore be very dense, in order to guarantee a reliable model.

By way of example, let's consider the model shown in fig. 12-38 of an underground tank on an elastic Winkler foundation, in which the base slab measures 15 x 9.5 m^2, and in which the elements have a typical dimension of about 30 cm per side.

Another example is shown in fig. 12-39, a foundation slab on a bed of Winkler springs (modelled using springs with adequate stiffness), loaded with vertical forces coming from the columns. The column grid is 5 x 5 m, the slab measures 10 x 30 m^2. The middle columns carry 105t, those on the sides 75t, those at the corners 37.5t. The object of study was how the response varied as the discretization was changed. We used 4-node elements obtained by assembling four triangular DKT elements (ANSYS finite elements implemented in SARGON).

All of the nodes are unconstrained, with the exception of those at the two opposite corners of the foundation's diagonal (see fig. 12-39); these two nodes were constrained as follows:

Tx Ty constrained on one node

Tx constrained on another node.

These constraints were sufficient to impede the translation in direction X or Y as well as the rotation of the entire slab about axis Z. The translation along axis Z and the rotation about axes X and Y are impeded by the bed of springs that model the Winkler foundation. Six models (A-F, figs. 12-39 and 12-40) were made, each of which had four times the number of elements of the preceding one: practically speaking, each element was subdivided into 2x2 elements at each step of increased density. Table 12-7 shows the values obtained for maximum displacement, the Von Mises stress under the middle column, slab moment m_x at the midpoint between the middle column and the next one placed at the centre of the slab (this information is missing in model A because there is no node there in that model).

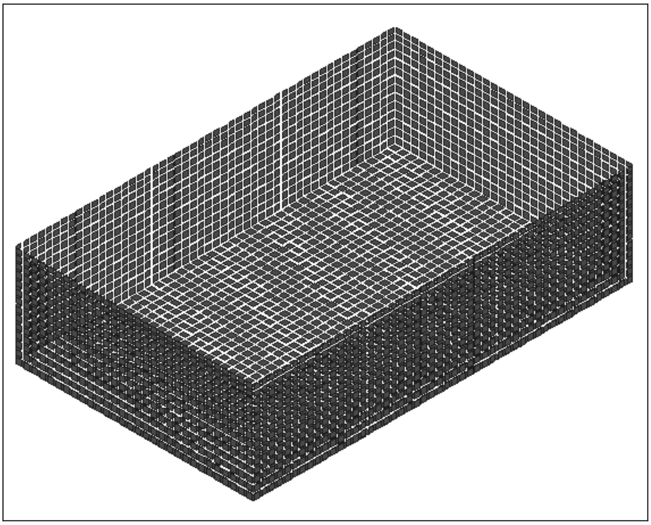


Figure 12-38: Example of a tank on elastic soil. The model has 22,900 degrees of freedom.

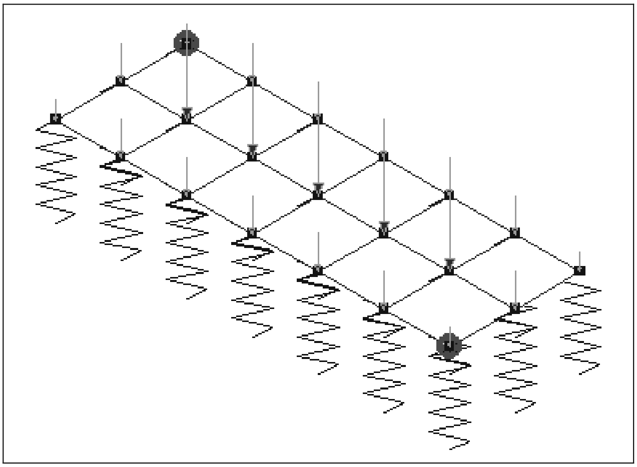


Figure 12-39: Model A, each grid module is 1x1 element. Column bays measure 5x5 m2. concrete R_{ck} 300, t=25cm, F_{max}=150t (middle columns), k_w=2Kg_f/cm^3.

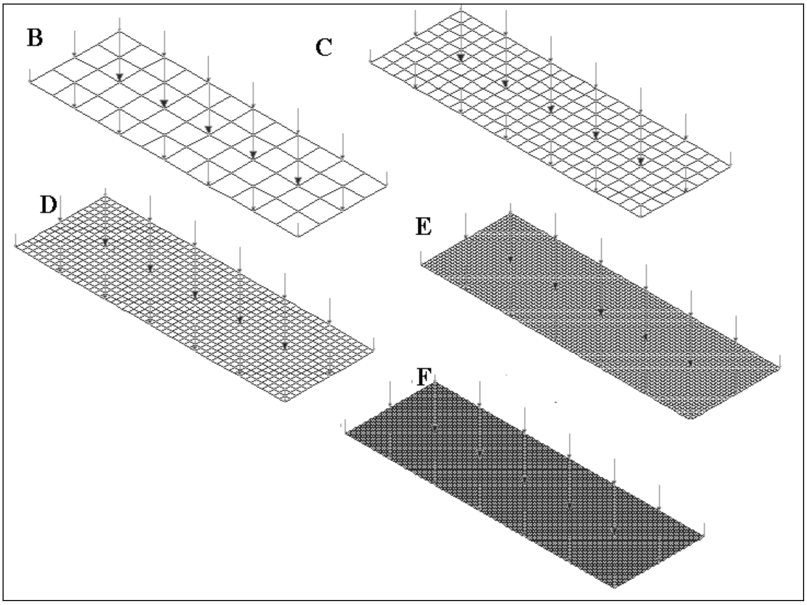


Figure 12-40: Models with increasing discretization of the original model of the foundation slab shown in fig. 12-39.

Our study allows us to make the following remarks:

- The convergence when using these kinds of elements and the springs below is very slow. Discretization with only 8x8 elements on a typical bay (model D) leads to errors that are still rather significant (11% for moment m_x, about 5% for maximum displacement, with respect to model F);
- The stresses under the columns grow indefinitely as the discretization is increased, because of the singularities in the problem. However, the peak area becomes increasingly smaller;
- The less refined models (A, B) severely distort the response and should not even be taken into consideration;
- A very high discretization required before we begin to find a convergence with an acceptable percentage of error.
- Another question is how do we correctly model a foundation slab with ribs? If the model allow us to do so, it may be useful to model the ribs as well with plate-shell elements in order to study the stress distribution correctly (in this case, the number of elements at the level of the ribs should be sufficiently high; it should be borne in mind that the bending

of the ribs is evidenced by the membrane behaviour of the finite elements, which is subject to the effects of locking mentioned earlier). If instead the foundation is part of a larger model (usually of the entire building) and it is necessary to keep the number of degrees of freedom to a minimum, then the ribs should be modelled using beam elements (typically, with a reverse-T section) of suitable stiffness, determined using the method of effective width.

Model	Grid	DOF	Displacement	σ Von Mises	m_x
A	2x6	123	-0.3	5.92e-3	--
B	4x12	387	-0.3	125.1	842.14
C	8x24	1347	-1.227	262.3	1475.5
D	16x48	4995	-1.437	369.4	2523.6
E	32x96	19203	-1.501	467.4	2780.4
F	64x192	75267	-1.517	563.3	2838.3

Table 12-7: Results of the comparative study of a Winkler foundation.

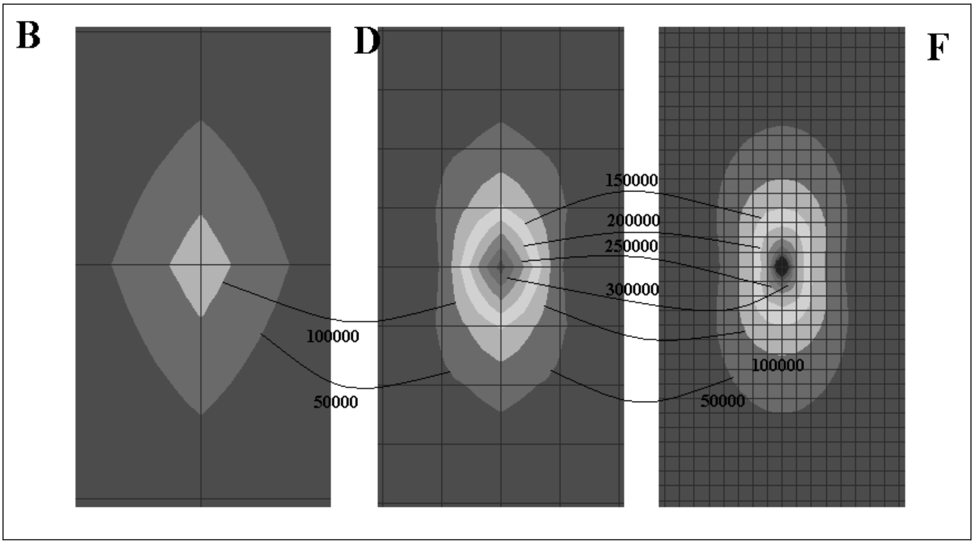


Figure 12-41: The increasing precision of contour lines of m_x under the central column in models B, D and F, Nmm/mm.

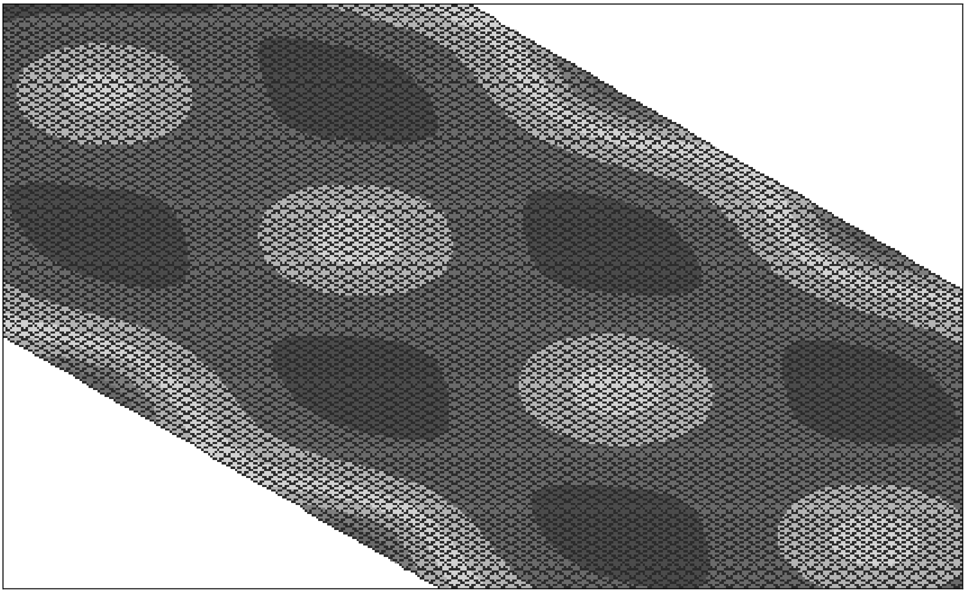


Figure 12-42: False colour map of vertical displacements, model F.

12.2.13 CYLINDRICAL TANKS, SHELLS

Mention is made here of how to model shell structures, which always require a very high degree of discretization for correct study. The discretization must be particularly dense in the case when low-order and planar plate-shell elements are used, where the geometry is actually similar to a given number of plane faces. In order to give an idea of what this problem entails, let's consider a cylindrical tank under an internal pressure, and see what happens as the discretization is varied.

Our example examines a steel cylinder 5m in height and 5m in radius, with a thickness of 10 mm, rigidly fixed at its base (fig. 12-43).

The cylinder is subject to a hydrostatic pressure that varies linearly between 0 (at the top) and 0.05 N/mm^2 at the bottom. The forces are modelled using a lumped approach. Triangular DKT elements are used.

The runs used are described in table 12-8, with five variations of the mesh (A-E). The last mesh (the densest) took advantage of symmetry and only ¼ of the cylinder was modelled.

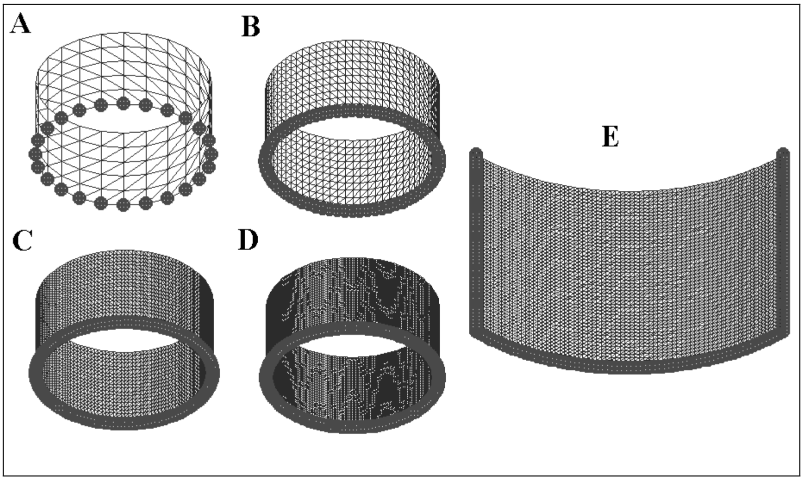


Figure 12-43: Meshes used for 5 case studies of the same cylinder with increasing discretization. Cylinder height 5m, radius 5m, thickness 10mm, E=206000N/mmq, ν=0.3. Head of hydrostatic pressure varying from 0 at the top to 0.05N/mm^2 at the base.

As can be seen, the results are less than impressive. The convergence is very slow, and for the value of the stress at the fixed joint the convergence is still not sufficient even with the most refined model (which is still 6.6% away from the values of the model immediately preceding it). In effect the problem has the (not rare) peculiar characteristic that the shell is subject to significant bending stress in the area of the fixed joint, as can be seen quite clearly in the deformed configuration shown in fig. 12-44.[4]

Therefore, in order to produce an analysis of the stress field that has any reliability at all, a high degree of discretization in the vertical direction is required. Meshes such as those of models A or B cannot possibly allow us to understand local effects of bending, which involve an area 600-700 mm high starting from the joint. It is very clear that finite elements with dimensions equal to 1000 mm or 500 mm cannot possibly show us these effects. What is needed is a mesh that is very dense in the vertical direction, while we might be able to accept a lower degree of discretization about the circumference (but without, however, losing sight of the problem of the shape of the elements, which should not be excessively distorted). Naturally, it is also possible to vary the pattern of the

[4] It is precisely in order to eliminate these effects that ribbed cylinders are made, so that the bending component is contained and the membrane component is increased.

mesh, increasing the dimensions of the elements gradually as they go up towards the area where deflection tends to disappear.

Model	Circumferential subdivisions (angle in degrees)	Subdivisions of height (height in mm)	DOF	Maximum displacement	σ Von Mises fixed joint	σ Von Mises at 1m from the fixed joint
A	24 (15°)	5 (1000)	720	0.3968	10.89	14.72
B	72 (5°)	10 (500)	4320	0.4843	13.58	19.59
C	144 (2.5°)	20 (250)	17280	0.5506	24.84	20.08
D	144 (2.5°)	40 (125)	34560	0.5540	36.19	20.06
E	288 (1.25°)	50 (100)	21600 (≈x4)	0.5645	38.61	20.03

Table 12-8: Results for models A-E. DKT elements with smoothing.

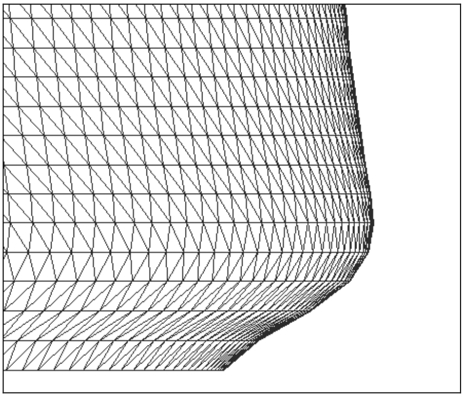


Figure 12-44: Deformed configuration of model E, near the fixed joint.

If we want to state the problem in a nutshell, we can say that in modelling shells, the utmost care must be taken when using plane plate-shell elements with three or four nodes, because they are very slow to converge. In all of these problems, what is called for is an analysis of how sensitive the solution is to variations in the discretization, in order to avoid using results that might actually be completely unreliable.

12.3 CONNECTIONS

Here we shall give some brief but useful notes in order to review the various possible kinds of connections between elements that are the same (homogeneous connections) or different (heterogeneous connections). Bear in mind that numerous examples have been introduced and discussed in other parts of this book, which shed more light on what will be only briefly reviewed here. To facilitate reference, the information will be treated point by point.

We presume that the elements have two nodes per side. Other presumptions have already been described elsewhere for elements that have more than two nodes per side.

12.3.1 HOMOGENEOUS CONNECTIONS

12.3.1.1 TRUSS ELEMENTS (RODS)

1. Each node to which only truss elements are connected has to be constrained for the three rotations unless the inactive degrees of freedom are automatically eliminated by the solver. If the trusses are coplanar, then the out-of-plane translation must also be constrained. If the trusses are aligned, the two translations normal to the alignment must be constrained.
2. A truss is "born" hinged at the node that it is attached to, and thus there is no reason to leave the rotations of the nodes of the trusses free.

12.3.1.2 BEAM ELEMENTS

1. Every beam element connected to another beam element is fixed with respect to it *unless an end release has been applied.* Every translation and rotation of the shared node is transmitted unaltered to the corresponding point on the axis line of the element.
2. Every node connected to a beam element that is not released in relation to it has a stiffness associated to all six of its possible motions. However, this does not mean that there cannot be singularities in the stiffness matrix.
3. In applying end releases to beam elements, internal and external conditions of hypostaticity should be guarded against.
4. If two beam elements meet at a node and are hinged with respect to each other, it is sufficient to apply an end release to only one of them rather than both; otherwise a condition of pseudo-mechanism is generated at the node.
5. Whether the analyst so desires or not, in the final analysis the bending behaviour of the beam elements depends on the orientation of the three local axes of reference, and therefore it is of the utmost importance to check their positions.

12.3.1.3 MEMBRANE ELEMENTS

In this section we presume that the formulation of the element does not include drilling DOF (cf. §11.7.5 and §11.5.3). If it does include drilling DOF, then it must be borne in mind that the rotation associated with drilling DOF is not unconstrained, and thus does not require constraint.

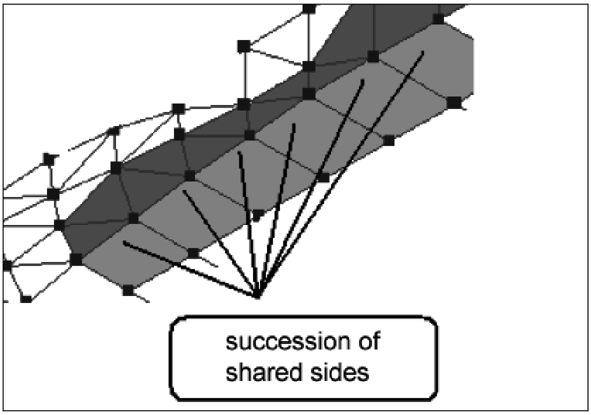


Figure 12-45: Succession of shared sides between two incident strips.

1. If only membrane elements are attached to a node, then the three rotations of the node are unconstrained and have to be constrained. If the membrane lies entirely in a plane parallel to the global coordinate planes, then the translation normal to that plane has to be constrained (because it is free).
2. A membrane has no bending stiffness, and thus it can never offer any resistance to the rotation of its nodes, regardless of the number and position of the membrane elements in question (the rotation about an axis normal to the plane of the element is thus associated with a stiffness if the element includes those degrees of freedom; cf. §11.5.3).
3. If two incident planes modelled with bidimensional elements meet, then along the shared side there is a series of nodes that form a succession of sides, and these sides must be shared by the elements of both planes.

12.3.1.4 PLATE-SHELL ELEMENTS

1. A plate element also models the membrane component, and should thus be considered as a bending element plus a membrane element.
2. All of the motions of each node in a plate element are associated with a stiffness, with the exception of the rotation about an axis normal to the plane of the plate. This rotation, in the most part of the formulations (see the type of element employed by the software program being used) is limited by a fictitious stiffness that is expressly added for this purpose. If several non-coplanar plate elements share a node, that node is then associated with stiffnesses for all of its possible motions.
3. See point 3 regarding membrane elements.

12.3.2 HETEROGENEOUS CONNECTIONS

12.3.2.1 TRUSS-BEAM

1. When a truss is connected to a beam, the truss is always hinged, and there is no need for end releases on the beam.
2. The mutual constraints of the beam elements have to be considered independently of the fact that truss elements are also connected to the node.

12.3.2.2 TRUSS-MEMBRANE

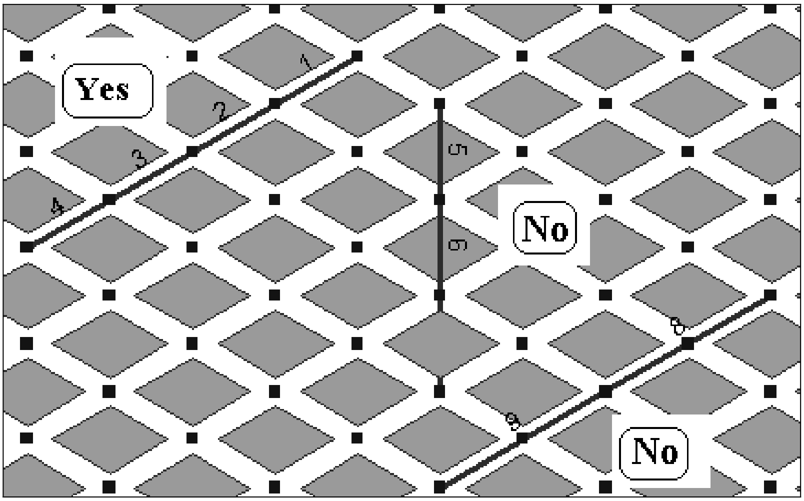


Figure 12-46: Incorrect connections and correct connections between beams elements and two-dimensional elements.

1. The connection of truss elements to membrane elements makes it possible to model membranes that are ribbed or orthotropic. In this case, as a rule each truss has to correspond to a side of the membrane element, without "skipping over" nodes or sides and without crossing the elements' diagonals (fig. 12-46).
2. If only truss elements or only membrane elements are connected to a node, then all of the rotations of the node have to be constrained.[5] If the membrane elements and the truss elements are coplanar, then the translation normal to the plane (presumed parallel to one of the global coordinate planes) has to be constrained.

[5] See the premise for §12.3.1.3: this is true if the formulation of the membrane elements does not include drilling DOF.

3. A truss element that meets in a normal direction a plane modelled with membrane elements will not encounter a restraint on the translations normal to the plane itself, but may offer a restraint if its second node (not lying in the plane) is suitably anchored.

12.3.2.3 TRUSS-PLATE

1. The connection of truss elements coplanar to plate elements makes it possible to model plates that are ribbed or orthotropic. The trusses modify the membrane behaviour without perturbing that of bending. In this kind of model as a rule each truss element has to correspond to the side of a plate element, without "skipping over" nodes or sides and without crossing the elements' diagonals (see fig. 12-46 above).
2. A truss element can meet normal to a plate, finding in it a support, thanks to the bending stiffness of the plate itself.
3. A truss element is naturally hinged with respect to the plate to which it is attached, without requiring any modifications.

12.3.2.4 BEAM-MEMBRANE

1. The connection of unconstrained, coplanar beam elements to membrane elements provides the membrane elements with a bending stiffness. However, the conditions mentioned in §12.3.1.3 apply to the nodes to which no beam elements are attached.
2. As a rule, each beam element (coplanar with the membrane elements) has to correspond to the side of a membrane element, without "skipping over" nodes or sides and without crossing the elements' diagonals (see fig. 12-46 above).
3. See point 3 of §12.3.2.2: the beam that is connected normal to a plane membrane does not encounter any restraint on movements normal to the membrane itself.
4. A beam element attached normal to or incident with a membrane element is hinged with respect to it without any end releases being applied (fig. 12-47). If end releases are placed on the rotation, then the node has to be constrained for rotation. If the end releases are omitted, the node does not have to be constrained. A beam element made in this way cannot be fixed with respect to the membrane element, unless special modelling tricks are used.

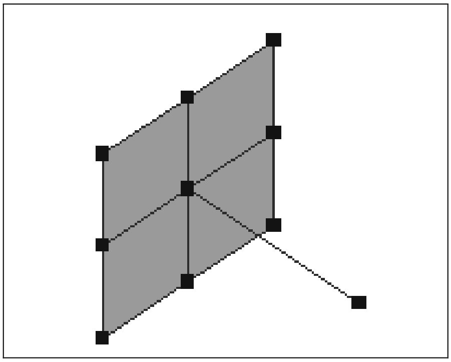


Figure 12-47: Attachment of a beam element to a membrane normal to its axis.

12.3.2.5 Beam-Plate

1. The connection of beam elements coplanar to a plate element makes it possible to model ribbed plates. To the plate element are added membrane stiffness (through the area of the beam element) and bending stiffness.
2. As a rule, each beam element (coplanar to plate elements) has to correspond to the side of a plate element, without "skipping over" nodes or sides and without crossing the elements' diagonals (see fig. 12-46 above).
3. The beams that connect normal to a plate are constrained for the movements normal to the plate, thanks to the bending stiffness of the plate itself.
4. The beam element incident with or normal to a plane of plate elements is fixed relative to it, unless end releases are applied to the beam.

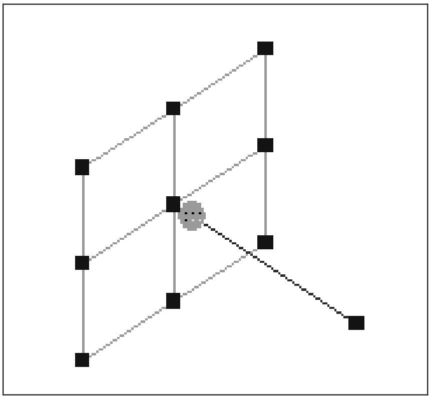


Figure 12-48: Attachment of a beam element to a plate normal to its axis.

12.3.2.6 Membrane-Plate

1. Membrane elements that are coplanar and contiguous with plate elements are continuous with them only as far as the membrane component of the stress is concerned, while the bending component terminates with the plate elements.
2. Plate elements are naturally hinged with respect to membrane elements without requiring any modifications.
3. A plane of plate elements that is connected normal to a plane of a membrane element is not supported for the movement in the direction normal to the plane of the membrane element. The plane of the plate element is hinged with respect to the plane of the membrane element.
4. If a plate element and a membrane element meet perpendicular to each other, then the rotation of the nodes about an axis perpendicular to the plane of the plate element is unconstrained. The other rotations are associated with the bending stiffnesses of the plate elements.

13 THE PROBLEM OF CHECKING

13.1 CHECKERS AND SOLVERS

In this final chapter we want to give a succinct outline of the set of general problems regarding the checking of structural elements, that is, to that set of analyses that are performed after the internal actions have been calculated, in conformity with a given set of standards (for steel, reinforced concrete, etc.).

In today's technical jargon, a clear distinction is made between *solvers*, which are the software programs for finite elements that we have discussed at length, and *checkers*, which are software programs which use the output of the solvers as the starting point to calculate a "coefficient of exploitation", that is, a dimensionless number related to the load bearing capacity of a structural element prescribed by a given set of standards.

Exactly how the "coefficient of exploitation" is defined varies from standard to standard and checker to checker. Generally speaking, the standards for allowable stresses involve the ratio between the calculated stress on the element and the allowable stress. Standards for limit states usually involve interaction formulas of this kind

$$k\left(\frac{N}{N_{pl}}\right)^{a} + l\left(\frac{M_2}{M_{2pl}}\right)^{b} + m\left(\frac{M_3}{M_{3pl}}\right)^{c} < 1,$$

which already lead to a dimensionless coefficient of exploitation.

The data required by a finite element software program and that required by a checker are only distantly related. Usually much less data is required by a solver than by a checker. For example, for a checker, information regarding the element's shape, its length-to-thickness ratios, the number, kind and arrangement of the reinforcing bars, and whether or not there are restraints on lateral sway, are all valuable. Instead, to describe a section a solver only requires a vector of the same six values: the area, the two shear areas, and the three second-order moments of inertia. With only this information, the checker couldn't even get started, and even adding the modulus of resistance W is not enough.

Checkers were created in order to keep the designer from having to do an enormous amount of calculating by hand, but also to perform a level of checking that would have been simply impossible to do manually: just think what it would mean to have to check dozens and dozens of different combinations, or to systematically perform thorough checks on all the elements, or to follow the structure through its nonlinear course.

In a trend that is by now consolidated, the calculations imposed on the designer by law are increasingly difficult to carry out manually. Thus in recent years there

Structural analysis with finite elements
978-0-7277-4093-9

has been a growing proliferation of tools for automatic checking aimed at lightening of at least a part of the burden imposed on the designer by legislation.

None of this, however, has translated into an improvement in absolute terms as far as the safety of structures is concerned, because designers have often abandoned the old design criteria which, although they were no longer in current use, still however provided significant protection against error.

The actual state of the art is that while calculations are no longer entirely performed manually, they are not yet completely performed by the computer. They are no longer manually performed because those calculations required by law are beyond the capacity of hand calculators (at least this is my opinion, but it is shared by many of my colleagues). But they are not entirely performed by the computer, because the laws are hardly ever written so that they can be programmed (more about this in the following section); because checkers can commit errors and omissions, and because the laws themselves can be incomplete and contradictory. These factors may become evident only at the moment the attempt is made to translate a law into a program.

The task of those using automatic checkers today is to aim for reasonable objectives and set inviolable limits. Here are some of the rules that should always be followed when using automatic checkers:

1. Be thoroughly aware of the norms implemented by the automatic checker;
2. Know the exact set of formulas and rules implemented by the checker, as well as all of the additional presumptions made by the developer of the checker;
3. Always pre-dimension a structure on the basis of simplified calculations that can be entirely controlled, independent of the checker;
4. Never pre-dimension a structure merely on the basis of calculations directly performed by the computer;
5. Always check the basic structural elements using the simplified rules, including those that are independent of the particular standards of reference;
6. Avoid accepting automatic designs of reinforcing or connections, unless their reliability has been systematically verified;
7. Perform crosschecks where possible, either by performing the checks on a single norm with more than one automatic checker, or using different automatic checkers to check against different norms;
8. Perform either by hand or with other systems all checks not carried out by the automatic checker;

9. Systematically perform sample checks on the results produced by the automatic checker.

Earlier, in chapter 9, we saw just how wrong it is to think that everything that comes out of a computer is correct. In the case of automatic checkers, we come up against the final step before the design goes into the phase of execution, and thus these must be given especially careful consideration.

13.2 DESIGNERS OF PROGRAMS AND DESIGNERS OF STANDARDS

13.2.1 HOW PROGRAMABLE IS A STANDARD?

One of the main obstacles in the preparation of computer software for checking consists in the fact that the prescribed standards cannot always be programmed. The real problems emerge when the standard cannot be readily complied with by means of manual techniques, but at the same time, is written in such a way that it cannot be programmed. Many standards have parts that cannot be programmed, and this is not necessarily a problem: programs are not required for generale principles, norms concerning execution, or standards for good design. But when a standard imposes certain calculations without defining unequivocally which data is to be used as a starting point, then suspicion and perplexity are naturally aroused.

The fact is that often those who write the standards have certain results in mind which may perhaps take into account a broad class of examples, but are, even so, still not general enough. The legitimate objective of a standards writer to prescribe general norms conflicts with the fact that in reality, he has to work from specific case histories.

One example of this is Italian standard CNR-10022, dealing with cold-rolled steel sections, which cannot in fact be programmed, except for the most basic cases. Here the norms regarding stiffened or unstiffened edges are extremely vague and impossible to apply generally because the definition of certain concepts, like that of "web", is taken for granted, while generally such concepts cannot be taken for granted at all.

Another example is found in Eurocode 3 (ENV 1993-1-1), where checks on stability make use of coefficients given in tables for particular cases, but which are unknown in general cases. This is specifically true with regards to Appendix F relative to the coefficients to be used to check lateral sway, where tables of coefficients are given for the classic cases of simple beams but no general formulas are given.

There are numerous other examples: things that appear "obvious" cannot always be programmed. For example, if a standard were to include the instruction "dead leaves are to be eliminated", but does not specify what a "dead leaf" is in a way that can be quantitatively programmed, then that instruction *cannot be programmed.* In reality, nothing is ever obvious in programming, and everything – or almost everything – must be defined.

There are cases in which the need to "program" a standard sheds light on problems of how the standard should be interpreted, or on possible errors contained in it. A recent example regarded the 2003 technical standards for structures in areas at risk for earthquakes [Italian ordinance 3274 of 20 March 2003]. Section 4.5.3 of the part regarding buildings described the way to combine the stresses of the structure's various dynamic modes, which was substantially by means of Wilson-Der Kiureghian's complete quadratic combination, or CQC [Wilson-Der Kiureghian 1981]. However, nowhere is it stated that the method used is Wilson's, nor is Wilson's paper cited, since the writer of the standard seemed to consider this unnecessary. In describing the defining formula of the method,

$$\mathrm{E}= \sqrt{\sum_i \sum_j \rho_{ij} E_i E_j}\ ,$$

the correlation coefficient ρ_{ij} is defined as follows:

$$\rho_{ij} = \frac{\left(8\xi^2\left(1+\beta_{ij}\right)\beta_{ij}{}^{3/2}\right)}{\left(\left(1-\beta_{ij}{}^2\right)^2 + 4\xi^2\beta_{ij}\left(1+\beta_{ij}\right)^2\right)}$$

where, and we quote, "ξ is the coefficient of equivalent viscous damping, β_{ij} is the ratio of the frequencies of each couple i-j of modes ($\beta_{ij}=\omega_i/\omega_j$)". This definition leaves us somewhat perplexed. In fact, coefficient β_{ij} is greater or lesser than 1 depending on the cycle of i and j, and it is legitimate to ask ourselves if we should take the value <1, > 1, or whether it matters which is used. Let's look at an example: imagine five modes with frequencies shown in the following table:

Mode	1	2	3	4	5
ω	1,5	13,5	14,5	15,5	15,0

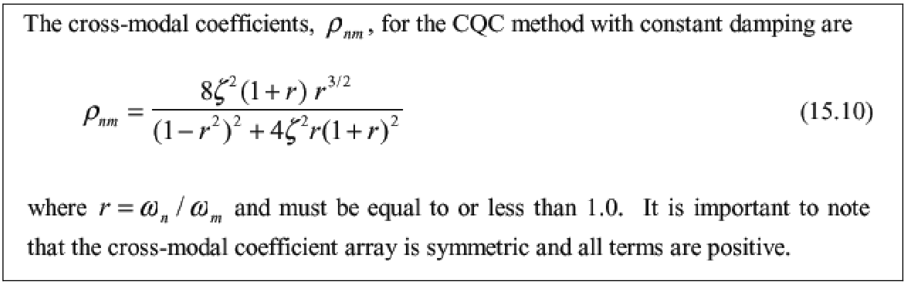
When i=1 and j=5 in the double summation β_{ij} will be equal to

$$\beta_{15} = (1{,}5/15{,}0) = 0{,}1.$$

Going forward with the double summation we arrive at i=5, j=1. In this case the formula gives the result:

$$\beta_{51} = (15{,}0/1{,}5) = 10$$

Is it then correct to use this number, or should we use the previous result? The norm says nothing about this, just as it said nothing about Wilson's original paper. But a doubt remains. Reading Wilson's paper[1] only adds to our discomfiture, because with regards to this he says:



The cross-modal coefficients, ρ_{nm}, for the CQC method with constant damping are

$$\rho_{nm} = \frac{8\zeta^2(1+r)\,r^{3/2}}{(1-r^2)^2 + 4\zeta^2 r(1+r)^2} \qquad (15.10)$$

where $r = \omega_n / \omega_m$ and must be equal to or less than 1.0. It is important to note that the cross-modal coefficient array is symmetric and all terms are positive.

It thus appears that our β must be <1. In actual fact, more in-depth analyses and checks show that the formula that gives the cross-correlation coefficient is invariant depending on whether we use (β) or (1/β). Here is the proof:

$$\frac{8\xi^2\left(1+\frac{1}{\beta}\right)\frac{1}{\beta^{3/2}}}{\left(1-\frac{1}{\beta^2}\right)^2+\frac{4\xi^2}{\beta}\left(1+\frac{1}{\beta}\right)^2}=\frac{\frac{8\xi^2(1+\beta)}{\beta^{5/2}}}{\frac{(\beta^2-1)^2}{\beta^4}+\frac{4\xi^2}{\beta}\left(\frac{1+\beta}{\beta}\right)^2}=\frac{\frac{8\xi^2(1+\beta)}{\beta^{5/2}}}{\frac{(1-\beta^2)^2}{\beta^4}+\frac{4\beta\xi^2}{\beta^4}(1+\beta)^2}=\frac{8\xi^2(1+\beta)\beta^{3/2}}{(1-\beta^2)^2+4\xi^2\beta(1+\beta)^2}$$

$$.$$

Formally, the standard is correct in not saying anything, but in our opinion the clarity that a norm has to have cannot be compared to that of a scientific paper. It would have thus been appropriate to specify in some way that it doesn't matter for the formula whether we set $\beta_{ij} < 1$ or > 1 (but was this evident to whoever wrote the standard?). Note that even Wilson's paper is misleading, because he should have said "and **they** must be equal to or less than 1.0" rather than simply "and must be equal to or less than 1.0" because what we understand is "and **it** must be equal to or less than 1.0".

Again, in the same norm, this time in the part regarding bridges, we find another example. The same CQC problem is addressed with a formula in which the viscous damping appears to disappear formally: in reality it is incorporated in the formula with the value of 0.05 (5%). In §7.1.2 of Wilson's paper the formula relative to the correlation coefficient is given like this:

$$r_{ij} = \frac{\left(0.02(1+\rho)\rho^{2/3}\right)}{\left((1-\rho^2)^2+0.01\rho(1+\rho^2)\right)},$$

[1] Available on the Internet at: http://www.csiberkeley.com/Tech_Info/15.pdf.

where the letter r now assumes the role played by ρ in the section regarding buildings, and ρ now assumes the role played by β (if we wanted to we could also criticise the way these symbols change meanings from section to section).

The formula gives every appearance of being incorrectly written: the (3/2) has become (2/3) and $(1+\rho)^2$ in the denominator has become $(1+\rho^2)$. In this way, when the frequencies are equal ($\rho=1$) the coefficient is equal to 2, while it should be equal to 1.

The point of view of the person who has to program a standard is often different from the person who has to write one. Everything must be carefully and unequivocally defined because it has to be transmuted into an instruction for the computer. If, as often happens, reference is made to a result found in the literature, the standard should state this explicitly (perhaps even providing a reference) in order to make further investigation possible (although at times we would hope to find further clarification within the standard itself). This may not be the case if the standards are not intended for specialists and if their applications are to remain obscure, but in this case why write standards?

In my opinion, standards should be comprehensible even to those who have no time to read the mass of literature behind them; neither should they presume that everyone will become a professional researcher. It is up to those who write them to translate the results of research into the standards in a way that is *comprehensible and unequivocal.* This includes citing the sources, not concealing them as though the standard had been written on tablets of stone, received directly from the word of God. And, if possible, the research results and formulas should even be written *correctly.*

Only rarely does the writer of standards offer the reader the luxury of everything being defined precisely and unequivocally, so that the programmer is forced to incorporate a set of assumptions that in fact imply "a standard" and not "the standard" (even though, objectively speaking, there is no such thing as "the standard"). For this reason every automatic checker is heavily influenced by a set of assumptions on the part of the programmer that are embedded in the checker and effectively constitute a possible interpretation of the standard but not "the" interpretation. Obviously, the closer that the programmer is to the discipline involved, then the more reliable the checker will be (all else remaining the same).

13.2.2 HOW FAR CAN A STANDARD BE EXTENDED?

Because of the considerations discussed above, checkers usually extend, more or less deliberately and more or less obligatorily, the prescribed standard in order to make it possible to address situations not strictly expressed in the standard.

It is very much to be hoped that the standard writer himself went to the effort to write the formulas so that safety is assured, or of making certain choices of what to simplify in the most complex cases rather than leaving this up to the programmer. But while the standard writer is reluctant to extend the prescribed standards beyond the experimental results, he is at the same time, however, reluctant to declare their limits explicitly.

Thus it can happen that prescribed standards that are valid in restricted contexts and under certain conditions can be, unwittingly and dangerously, extended to other situations without the risks and hazards being clearly apparent.

13.2.3 THE CONSISTENCY OF STANDARDS

A further problem in the interpretation of a standard, from the point of view of the programmer, is the search for consistency. It is quite possible to encounter standards that contain passages that are contradictory and inconsistent.

Let's take the case of the classifications of sections according to Eurocode 3. The standards state that classification should be made on the basis of the stresses present in each section of each element. Thus, strictly speaking, it is possible to have sections that are in one class for a certain load combination, and in a different class for other combinations. Along an element the class can vary from section to section, because in general the stress regime varies. In certain cases the sample being analysed is broken down in to parts in order to be able to take only effective parts into account (class 4), which leads to there being different sections at different points of an element and the various combinations being checked. The resulting partial sections are, however, calculated with their stress determined using gross data, since otherwise a double iteration would be necessary (one for solving, and one for the individual section). When the checks for stability are then performed, the gross characteristics are used. Frankly, all of this reeks of compromise, because it is quite evident that the standards are inconsistent and prescribe a lame method of calculating.

This is not a lone case. In calculating the sections in reinforced concrete, ample use is made of non-linear laws and rectangular parabolas starting from calculations that are almost always linearly elastic. Does it really make sense to try to jump through hoops after having already gone to all this trouble? I don't think so.

13.2.4 ERRORS AND TESTS

To the problems related to implementing standards – a process already fraught with difficulties – are added the problems related to errors in the software. We have already mentioned the significant problem of bugs in the software, but there is another problem as well that plagues checkers. While it is in fact possible in solvers to perform a comparative analysis of the results of solving, and theoretically there is an "exact" answer within the limits of the assumptions made in the model, in the context of checkers, because of all the problems of interpretation just discussed, *it isn't always possible to compare the results of two checkers* *directly, because they don't necessarily have to be the same.* In fact, if differences are found, what are they to be attributed to? To an error that only shows up in a latent way, or to the different ways in which the standards were implemented and interpreted?

This doesn't mean that comparing the results of different checkers is not advisable; to the contrary. The point is that mere comparison reveals little. It is necessary to retrace all the various steps in order to understand the reasons behind the differences, since, strange as it may seem, in general the "exact" solution is not a unique one.

We can thus see that it is up to the user, with the help of checkers, to validate the assumptions and results every time, using the checkers as auxiliaries rather than relying on them completely. The user has to subject the models to careful checks and compare his simplifying hypotheses, his own interpretation of the standards, with those of the software program.

13.3 THE RETURN OF ARBITRARINESS

From what has been sketched out here we can see that *mala tempora currunt*, these are difficult times. At the moment we are in a situation of transition that unfortunately leaves ample room for significant risks: the standards are not yet written in such a way to be clear and programmable; the users tend not to take responsibility for their results, convinced that the automatic checker is a kind of Superman; the software houses try to meet market demands by turning out products and checkers, not always advertising them in a way that is consonant with the actual state of the art.

This has all led to a state of unadulterated arbitrariness, a situation in which assumptions with no basis in truth can be directly applied to a structural project, almost uncontrolled by those who instead should be aware of every single detail of the design. We can even find the use of project designs created completely automatically by software as though the level achieved by today's computer made it possible to substitute the structural designer. It is essential to remember that although today we might be tempted to eliminate the designer and let a computer program produce reams of paper, *there is no substitute for the expert designer.*

14 APPENDIX I: NOTATION AND REVIEW OF CALCULUS

For the sake of simplicity, in keeping with the nature of this present work, tensor notation is excluded, while use is made of matrix notation.

In the text frequent use is made of the following conventions:

k,W non-bold, lower- or upper-case letters represent numbers

u, p lower-case letters in bold represent vectors

K, M upper-case letters in bold represent matrices

I is the unitary square matrix (the diagonal contains all 1s, with 0s elsewhere)

Column vectors are usually indicated by a lower-case letter in bold: **g**. Another way for describing a column vector is: $\{g_1, g_2, \ldots, g_n\}^T$.

Row vectors are indicated as transposed column vectors: $\mathbf{g}^T$. Another way of describing row vectors is: $\{g_1, g_2, \ldots, g_n\}$.

The product **c** of a rectangular matrix **A** and a column vector **b** is written:

$$\mathbf{c}=\mathbf{Ab}$$

In order for the operation to make sense, it is necessary that

$$p=q$$

where p is the number of columns in **A** and q is the number of rows in **b**. The resultant column vector **c** has the same number of rows *n* as **A**. This fact is written in symbols as:

$$(n, p)(p, 1)=(n,1)$$

The scalar product of a row vector $\mathbf{r}^T$ by a column vector **c** is a *number* k:

$$k= \mathbf{r}^T \mathbf{c}$$

If **Q** is a square matrix of *n* order, and **b** is a column vector of *n* order, the product

$$k= \mathbf{b}^T\mathbf{Qb}$$

is a number. This product is also said to be the *quadratic form* in **b** associated with **Q**.

A system of linear equations of order *n*, where **x** is the vector of unknowns, **b** is the vector of known terms, and **Q** the matrix of coefficients, is written

$$\mathbf{Qx}=\mathbf{b}$$

and has as its solution the vector

$$\mathbf{x}=\mathbf{Q}^{-1}\mathbf{b}.$$

Matrix $\mathbf{Q}^{-1}$ is called the *inverse* of matrix **Q**. The following rule holds:

$$\mathbf{Q}\,\mathbf{Q}^{-1}=\mathbf{Q}^{-1}\,\mathbf{Q}=\mathbf{I}.$$

If **R** is a rectangular matrix, then a matrix obtained by exchanging the rows and columns (the first row becomes the first column, the second row becomes the second column, and so forth) is defined as a transpose of **R**, and is indicated by the symbol $\mathbf{R}^{T}$.

Given **AB**, the product of two rectangular matrices **A** and **B**:

$$(\mathbf{AB})^{T}=\mathbf{B}^{T}\mathbf{A}^{T}.$$

A square matrix **O** is said to be *orthogonal* if its inverse coincides with its transpose

$$\mathbf{O}^{T}=\mathbf{O}^{-1}.$$

For an orthogonal matrix the following rule holds:

$$\mathbf{Q}\,\mathbf{Q}^{T}=\mathbf{Q}^{T}\,\mathbf{Q}=\mathbf{I}$$

♣

For simplicity's sake, it is often useful to consider a matrix or a vector as if partitioned into elementary blocks made up of submatrices or subvectors of a lower order. When a matrix or vector is imagined to be partitioned into blocks, the individual terms can be written in the concise form "$\mathbf{u}_{\mathbf{i}}$" and in this case the contemporary presence of boldface and of subscripts make it understood that this indicates a subblock (in contrast to u_i); the extended version is

$$\left|\begin{array}{c}\mathbf{u}_{\mathbf{1g}}\\ \hdashline \mathbf{u}_{\mathbf{2g}}\\ \hdashline \mathbf{u}_{\mathbf{3g}}\\ \hdashline \mathbf{u}_{\mathbf{4g}}\\ \hdashline \mathbf{u}_{\mathbf{5g}}\\ \hdashline \mathbf{u}_{\mathbf{6g}}\\ \hdashline \mathbf{u}_{\mathbf{7g}}\\ \hdashline \mathbf{u}_{\mathbf{8g}}\end{array}\right|$$

In this case as well it can be seen that the individual terms are in bold and are thus matrices (in this case, vectors). For example, a square 12x12 matrix can be seen as a matrix made up of four 6x6 blocks; a (12, 1) vector as two (6, 1) vectors; a (12, 1) vector as four (3, 1) vectors, and a (144, 144) matrix as a matrix of (12, 12) blocks of (12, 12).

♣

When a function depends on several variables, the derivative of the function with respect to one of the variables is called a partial derivative. The symbol for a partial derivative is "∂", as for example "∂f/∂x". For the sake of brevity, we will indicate the partial derivative with respect to variable *x* thus: $f_{,x}$.

When a function depends on only one variable, we will write its derivatives with superscripts: f', f'', f^{III}, f^{IV}, etc.

♣

The notation

$$(1.0e - 13)$$

is equivalent to

$$1.0 \times 10^{-13}$$

15 APPENDIX II: REVIEW OF THE THEORY OF ELASTICITY

In this appendix we will briefly summarise some of the important results in the theory of elasticity. Numerous textbooks are available for study in greater detail. Significant among these for its clarity, in spite of its being inevitably dated in terms of approach, notation and contents, is the classic text of Timoshenko and Goodier [1970]. A concise treatment from the point of view of physics more than of engineering is found in Landau and Lifshitz [1979]; this also includes the references to all the significant works comprising the history of the theory of elasticity. Corradi dell'Acqua [1992] is a complete, modern treatment suitable for in-depth study, and also includes broader topics such as the theory of plasticity, the Finite Elements Method, and instability.

Given a body or a system of bodies, constrained or free and subject to a set of actions and reactions, the following unknown quantities, functions of point P of coordinates (x, y, z) are thus defined:

u(P) translation of the point in direction x;

v(P) translation of the point in direction y;

w(P) translation of the point in direction z.

Given displacements **U**= $\{u, v, w\}^T$ we can define the *deformations* in the following manner:

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

$$\varepsilon_y = \frac{\partial v}{\partial y}$$

$$\varepsilon_z = \frac{\partial w}{\partial z}$$

[15-1]

$$\gamma_{xy} = \gamma_{yx} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}$$

$$\gamma_{yz} = \gamma_{zy} = \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}$$

Structural analysis with finite elements
978-0-7277-4093-9

$$\gamma_{zx} = \gamma_{xz} = \frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}$$

Briefly, the vector of deformation $\boldsymbol{\varepsilon}=\{\varepsilon_x, \varepsilon_y, \varepsilon_z, \gamma_{xy}, \gamma_{yz}, \gamma_{zx}\}^T$ is written as the function of the displacements

$$\boldsymbol{\varepsilon}=\mathbf{CU}$$

having set

$$\mathbf{C}\equiv\begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \end{bmatrix}.$$

There are nine deformations, but only six are independent. The first three are called elongations; the last three shears. There are various ways to define deformations when they are large, and these different ways lead to values identical to those for the deformation when the displacements are sufficiently small.[1] In general the measurement of the deformation begins with the difference between the square of the distance of two points that are infinitely close before and after displacements caused by applied loads. If *before* the points were at distance dL apart, and *after* they are at distance dl apart, then measurement of the deformation begins with the difference between dl^2 and dL^2, that is, $dl^2 - dL^2$.

In ordinary structures the displacements are usually so small that it is possible to measure the deformations using the formulas given. When the displacements become significant, the definitions indicated are no longer valid, and must be replaced with formulas which also include higher-order derivatives (Green-Lagrange strain or Euler-Almansi strain).

The deformations measure the unit quantities of displacement at a certain point. To this unit displacement is associated, in the material, a certain level of

[1] Deformation is defined in various ways. For a treatment of all the various possible tensors of deformation, see for example [Dell'Acqua 1992, vol. 1], [Mase 1970], and [Bathe 1996].

strain. In a solid there are six independent components of stress at each point, just as there are six independent components of deformation at each point.

The stresses are divided into normal stresses (indicated by the letter σ) and shear stresses (indicated by the letter τ).

Let us consider a plane passing through a generic point P and having as normal a vector parallel to the *x*-axis; this plane will "slice" the solid into two parts, which exchange internal forces. These forces tend to become increasingly small as the suface of the sampling surface around point P decreases, but the relationship between the force $\mathbf{f}_{xS}= \{f_x, f_y, f_z\}^T$ and the surface about point S tends, as surface S decreases, to a constant value (inasmuch as the force diminishes, but so does the surface being analysed)

$$\mathbf{f}_{xS}/S\text{------}>\boldsymbol{\phi}_x.$$

The components of this vector $\boldsymbol{\phi}_x$ are *by definition* the components of stress in P on the face having as normal a vector parallel to the *x*-axis:

$$\boldsymbol{\phi}_x \equiv \{\sigma_x, \tau_{xy}, \tau_{xz}\}^T.$$

If now the solid is sliced in P by a plane having as normal the *y*-axis, and the same limit is made by S tending to zero, the forces on a face having as normal a vector parallel to the *y*-axis are obtained:

$$\boldsymbol{\phi}_y \equiv \{\tau_{yx}, \sigma_y, \tau_{yz}\}^T.$$

If the solid is sliced in P by a plane having as normal the *z*-axis, then the stresses on a face having as normal a vector parallel to the *z*-axis are obtained:

$$\boldsymbol{\phi}_z \equiv \{\tau_{zx}, \tau_{zy}, \sigma_z\}^T.$$

The state of stress in point P is described by the set of three vectors $\boldsymbol{\phi}_x$, $\boldsymbol{\phi}_y$, $\boldsymbol{\phi}_z$, and thus by the nine components of the stresses. However, a reading of what occurs in a face passing through point P is not sufficient, and three faces orthogonal to each other are necessary.

If we write the equilibrium conditions for an infinitely small volume of the solid (the "little cube" with faces parallel to the axes of reference), we find that in actual fact only six of the nine components are independent, inasmuch as they are the result of the following six *indefinite equilibrium equations*:

$$\sigma_{x,x} + \tau_{xy,y} + \tau_{xz,z} + b_x = 0$$

$$\tau_{xy,x} + \sigma_{y,y} + \tau_{yz,z} + b_y = 0$$

$$\tau_{xz,x} + \tau_{yz,y} + \sigma_{z,z} + b_z = 0$$

[15-2]

$$\tau_{xy} = \tau_{yx}$$

$$\tau_{xz} = \tau_{zx}$$

$$\tau_{yz} = \tau_{zy}$$

in which $\mathbf{b}=\{b_x, b_y, b_z, \}^T$ are volume forces (such as, for example, gravity). The first three equilibrium equations are equilibrium equations of the translations of the small cube; the last three are the equilibrium equations of rotation.

The last three indefinite equilibrium equations tell us that there are three, not six, unknown stresses. Thus we can consider as an unknown the stress vector $\boldsymbol{\sigma}=\{\sigma_x, \sigma_y, \sigma_z, \tau_{xy}, \tau_{yz}, \tau_{zx}\}^T$.

In linear elastic materials there is a linear relationships with constant coefficients between deformations and stresses. With the introduction of the square matrix **D** (6x6) known as the *material stiffness matrix*, it is possible to write

$$\boldsymbol{\sigma}=\mathbf{D}\boldsymbol{\varepsilon},$$

which expresses Hooke's law in generalised form. Of the 36 unknown coefficients only 21 must actually be determined, because it can be proven that matrix **D** is necessarily symmetrical (and thus the unknowns are (36-6)/2+6= 21).

Matrix **D** must be symmetrical because the specific energy of elastic deformation stored in the small cube of material of unit volume, z, can be written in the following two ways (each component of deformation is multiplied by *its* component of stress, all of these products are added up, the sum multiplied by 0.5):

$$z = 0.5\, \boldsymbol{\sigma}^T \boldsymbol{\varepsilon}$$

$$z = 0.5 \boldsymbol{\varepsilon}^T \boldsymbol{\sigma},$$

and thus

$$\mathbf{D}=\mathbf{D}^T.$$

The deformation energy Z stored in volume V is

$$Z=0.5\int\boldsymbol{\sigma}^T\boldsymbol{\varepsilon}dV \qquad [\,15\text{-}3]$$

In the important case of homogeneous and isotropic materials, the unknown constants are reduced to 2 and matrix **D** can be written:

$$\mathbf{D}=\begin{array}{|c|c|c|c|c|c|}\hline \lambda+2G & \lambda & \lambda & 0 & 0 & 0 \\ \hline \lambda & \lambda+2G & \lambda & 0 & 0 & 0 \\ \hline \lambda & \lambda & \lambda+2G & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & G & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & G & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & G \\ \hline \end{array}$$

Where we set

$$\lambda=\frac{vE}{(1+v)(1-2v)} \qquad \text{(first Lamé parameter)}$$

$$G=\frac{E}{2(1+v)} \qquad \text{(second Lamé parameter, shear modulus of elasticity)}$$

and define

E Young's modulus of elasticity

ν Poisson's ratio

Poisson's ratio measures the transversal contraction of a sample of a material stretched in a longitudinal direction. Inverting the relationships of stress to deformations makes it possible to find the relationships of deformations to stresses, which take the form

$$\boldsymbol{\varepsilon}=\mathbf{D}^{-1}\boldsymbol{\sigma}$$

where the matrix $\mathbf{D}^{-1}$ is known as the *material compliance matrix*

$$\mathbf{D}^{-1}=\begin{bmatrix} \frac{1}{E} & \frac{-\nu}{E} & \frac{-\nu}{E} & 0 & 0 & 0 \\ \frac{-\nu}{E} & \frac{1}{E} & \frac{-\nu}{E} & 0 & 0 & 0 \\ \frac{-\nu}{E} & \frac{-\nu}{E} & \frac{1}{E} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G} \end{bmatrix}.$$

Poisson's ratio has to have a value between 0 and 0.5; in general, for the materials most frequently used, this assumes a value between 0.18 and 0.35. Poisson's ratio is a pure number.

Young's modulus has the physical dimensions of a stress (force per unit of surface) and provides the measure of resistance of the material to elongation.

The following table summarises the values of E and ν of various materials (at normal temperature).

Material	**Modulus E (N/mm^2)**	**Ratio ν**
CNR Steel Standard	206000	0.287
EC3 Steel Standard	210000	0.3
Stainless steel	215000	0.283
Aluminum	70300	0.345
Concrete Rck 20 [D.M. 14-2-1992]	25490	0.2
Concrete Rck 25 [D.M. 14-2-1992]	28500	0.2
Concrete Rck 30 [D.M. 14-2-1992]	31220	0.2
Concrete Rck 35 [D.M. 14-2-1992]	33720	0.2
Iron	212000	0.293
Brass	101000	0.350
Copper	130000	0.343
Titanium	120000	0.361
Siliceous glass [Dowling 1993]	70000	0.18
Glass [Saint-Gobain 1997]	73000	0.22

In addition to homogeneous and isotropic materials, there are also orthotropic materials. These are characterised by different mechanical behaviour in three orthogonal directions. Naturally, neither isotropic nor orthotropic materials actually exist: these are simplifications that can be more or less approximated by experimental behaviour.

The deformations-displacements laws for orthotropic materials make use of the following material compliance matrix:

$$\mathbf{D}^{-1}= \left|\begin{array}{c|c|c|c|c|c} \frac{1}{E_x} & \frac{-\nu_{xy}}{E_y} & \frac{-\nu_{xz}}{E_z} & 0 & 0 & 0 \\ \hline \frac{-\nu_{yx}}{E_x} & \frac{1}{E_y} & \frac{-\nu_{yz}}{E_z} & 0 & 0 & 0 \\ \hline \frac{-\nu_{zx}}{E_x} & \frac{-\nu_{zy}}{E_y} & \frac{1}{E_z} & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & \frac{1}{G_{xy}} & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & \frac{1}{G_{yz}} & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{zx}} \end{array}\right| .$$

Of the 12 constants introduced only 9 are independent, inasmuch as the matrix must be symmetrical, thus:

$$\nu_{yx}/E_x = \nu_{xy}/E_y$$

$$\nu_{zx}/E_x = \nu_{xz}/E_z$$

$$\nu_{zy}/E_y = \nu_{yz}/E_z.$$

The model of orthotropic material can be useful in the case of particular materials for construction such as wood, which effectively does exhibit different behaviours depending on the direction of stress, or because of an orthotropy due to construction, as for example occurs in slabs with directional ribs or in masonry.

It is possible to arrive at the solution set of equations by combining the congruency equations that relate displacements and deformations to each other, the constitutive relations that related deformations to stresses, and the indefinite equilibrium conditions that the stresses have to satisfy.

If the displacements are held as the main unknowns, we arrive at the Navier-Stokes equations, which are complicated partial differential equations in which the three unknown translations u(P), v(P), w(P) appear as unknowns. Recalling that

$$\nabla^2 = \partial^2(\cdot)/\partial x^2+\partial^2(\cdot)/\partial y^2+\partial^2(\cdot)/\partial z^2,$$

the Navier-Stokes equations are:

$$(\lambda+G)\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 v}{\partial x\partial y}+\frac{\partial^2 w}{\partial x\partial z}\right)+G\nabla^2 u+b_x=0$$

$$(\lambda+G)\left(\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 u}{\partial x\partial y}+\frac{\partial^2 w}{\partial y\partial z}\right)+G\nabla^2 v+b_y=0$$

$$(\lambda+G)\left(\frac{\partial^2 w}{\partial z^2}+\frac{\partial^2 v}{\partial y\partial z}+\frac{\partial^2 u}{\partial x\partial z}\right)+G\nabla^2 w+b_z=0$$

These equations tell us how to calculate the displacements and, in formal terms, resolve the problem, because they make it possible to calculate the displacements. When the displacements are known, the deformations are known, and from them, the stresses.

In substance, however, except for some particular problems where it is possible to introduce suitable simplifications, the Navier-Stokes equations are not useful because they are too complicated. These equations can be simplified by assuming a certain number of simplifying hypotheses about either the form of the structure (planes, cylinders, half-planes, spheres, etc.) or the complexity of the solution (plane stress or plane strain, axial symmetry, etc.). Thus we arrive at a certain number of simplified formulations that are widely used in practice and summarised in chap. 3.

16 APPENDIX III: PRINCIPAL SYMBOLS

Latin letters

A	area
a	side of a rectangle, semi-major axis of an ellipse
A	rectangular matrix that makes it possible to go from interpolation parameters to nodal displacements
b	side of a rectangle, semi-minor axis of an ellipse
b	vector of volume forces
B	matrix of functions that, when multiplied by nodal displacements, give the deformations in a point
C	kinetic energy, curvature
C	matrix of operators that, applied to **U**, give the vector of generalised deformations
D	bending stiffness of plates
D	stiffness matrix of the material in the formulation in question
f	vector of equivalent of nodal forces
F	force
k	stiffness
K	stiffness matrix of the overall structure
$\mathbf{K}_e$	stiffness matrix of element "e" in the local reference system
$\mathbf{K}_{eg}$	stiffness matrix of element "e" in the global reference system
E	Young's modulus (of elasticity)
G	shear elastic modulus; centre of gravity
I	moment of inertia
l	length of an elongated rectangle
L	length
m	mass; order of matrix $\mathbf{K}_e$, moment for the unit of length; plate moment
M	bending moment
N	number of nodes in an element
p	load per unit of length
P	generic point in a domain
p	vector of surface forces; vector of equivalent nodal forces

Structural analysis with finite elements
978-0-7277-4093-9

q	load per unit of length; load per unit of surface
T	shear; translation
$\mathbf{T}$	transfer matrix
$\mathbf{f}$	vector of equivalent nodal reactions
r	radial direction
R	rotation, reaction
s	thickness of hollow profiles
t	plate thickness; size per unit of length (with subscript); plate size (with subscript)
u	translation in direction x
$\mathbf{u}$	vector of the nodal displacements of the overall structure
$\mathbf{U}$	column vector that comprises the unknown displacement functions for a given formulation
$\mathbf{u_e}$	vector of nodal displacements of element “e” in the local reference system
$\mathbf{u_{eg}}$	vector of nodal displacements of element “e” in the global reference system
v	translation in direction y; velocity
V	volume, shear force (with subscripts)
z	generic unknown displacement function; specific strain energy
Z	elastic deformation strain energy (Z=zdV)
w	translation in direction z
W	work performed by applied loads

Greek letters

α	angle
β	angle
χ	shear factor
Γ	perimeter, border or boundary
ε	deformation
ϕ	Airy or Love stress function
$\mathbf{\Phi}$	shape function
λ	dimensionless abscissa; slenderness; Lamè parameter
ν	Poisson’s ratio

Π	total potential energy; elementary interpolation functions
θ	circumferential direction
σ	normal stress
$\boldsymbol{\sigma}$	vector of generalised stresses
τ	shear stress
Ω	area enclosed by the mean line of a hollow profile

Subscripts

e	element 'e'
g	global
i	generic element of a set; row of a matrix
j	generic element of a set; column of a matrix

17 APPENDIX IV: ESSENTIAL GLOSSARY

ALL IN CORE	Situation in which it is possible to resolve a mathematical system keeping all the data in computer RAM, without writing to disk
ALPHANUMERIC FILE	text file that can be edited
ANISOTROPIC	not homogeneous in all directions; model of behaviour of material that allows for substantial differences according to the direction in which it is stressed
ANTISYMMETRY	a particular load situation in symmetric structures that makes it possible to predict a response in terms of displacement, which is also antisymmetric
ASSEMBLY	procedure used to construct the global stiffness matrix of the structure adding the contributions of all the finite elements of which the structure is composed
AXIAL SYMMETRY	a particular condition of structural symmetry where the structure with its loads possesses infinite planes of symmetry all passing through a single axis
BACK SOLVING	inverse solution. A numeric procedure which, beginning with the last equation of a triangular system and one unknown, calculates all of the other unknowns in increasing numeric order. It is usually the final phase of solving (see *solving*)
BAND	1 + the difference between the position of the last non-null term and the position of the diagonal end on the row of the matrix being considered
BANDWIDTH OPTIMISATION	method that attempts to reduce bandwidth to a minimum by means of appropriate renumbering
BEAM (ELEMENT)	finite beam element
BLOCK	a part into which the stiffness matrix has been divided to make the solution of a system of equations possible. Each block usually has some pre-determined dimension
BRICK (ELEMENT)	solid finite element with 8 or 20 nodes
BUCKLING	instability
BUG	program error
BUG FIXING	elimination of a program error
BUG LIST	list of program errors
C, C++	the dominant programming language in modern software, equipped with a graphic interface
CHECKER	program used to check that the stresses and deformations resulting from calculation comply with the laws of a given norm or standard
CHOLESKY METHOD	a system of solving systems suitable for positive definite matrices
COMPATIBILITY	the capacity of an element or a formulation to show displacements along the edge equal to those of the adjacent element

Structural analysis with finite elements
978-0-7277-4093-9

COMPLETENESS absence of asymmetries and deficiencies in the number and type of the elementary functions used by an interpolating function; the ability to group rigid displacements and constant states of deformation

CONDENSATION a computational trick with which an *m*-order problem of finite elements is transformed into a problem of lower order *n*

CONDITIONING NUMBER index number intended to predict the numeric difficulty of solving a problem of linear algebra

CONGRUENCE when a set of displacements respects internal and external constraints; identity between predicted displacements of two separate elements with a shared edge

CONSISTENT APPROACH the opposite of lumped approach; the technique that evaluates nodal force and mass in keeping with the hypotheses formed about the displacement field within the element. It leads to more complex contributions and to denser nodal force vectors than the lumped approach

CONSTANT STRAIN TRIANGLE (CST) the simplest element for static planes of stress or deformation; it is triangular and has 3 nodes

CONSTITUTION typical relationships between stresses and strains in a given material

CONVERGENCE the tendency shown by successive steps of a procedure to reach a particular unique solution

CROSS CHECK a method of checking that compares one or more results in different models or programs, in order to demonstrate the correctness of the results themselves

DEFAULT value or choice assumed by a program in the absence of a specific indication to the contrary; standard procedure

DISCONNECTION the lack of relationship between one node and another, between a node and an element, or between two elements

DISCRETISATION the method by which a continuum is transformed into the sum of a (large) number of distinct elementary parts. Transformation of differential equations into systems of linear equations

DISPLACEMENT METHOD method of solving elastic problems that considers displacements to be primary unknowns

DISPLACEMENTS, SMALL basic assumptions of continuum mechanics that make it possible to write the deformations in a simplified way, ignoring higher-order terms (geometrically small displacements). Basic assumptions that make it possible to write the equilibrium conditions with respect to the undeformed configuration (physically small displacements)

DISTRIBUTED LOAD a load that concentrates on internal points of a finite element and that thus requires estimation of the equivalent nodal forces

DKT discrete Kirchhoff triangle; widespread and frequently used triangular Kirchhoff plate element

DOF degree of freedom

DOUBLE NODES nodes numbered separately but occupying the same position in space

DUMMY	a term introduced with no other aim than to exist and have some value (for example, unnecessary parameters)
END RELEASE	disconnection of one or more motions of a node from the displacements within an element. The motion in question is not hindered in any way by the released element
EQUIVALENCE	adjustment of a finite element aimed at reproducing the behaviour of complex parts
EULERIAN DESCRIPTION	description of the deformation that maintains a fixed position in space, taking into consideration the various particles that occupy that position over time
FEM	finite element method
FORCE METHOD	method of solving elastic problems that considers forces or stresses to be primary unknowns
FORMATTED FILE	text file in which numbers are arranged in strict accordance to a predefined format
FORTRAN	formula translation. Programming language once frequently used in the scientific world, by now quite dated
FORWARD SOLVING	numeric procedure that starts with the first equation in an unknown in a triangular system and calculates all of the other unknowns in increasing numeric order. Usually the intermediate phase between reduction and back solving
GAUSSIAN ELIMINATION	technique for solving systems of linear equations consisting in the progressive elimination of unknowns using elementary row operations followed by back substitution
GAUSS POINT	precise point in a domain where a function to be integrated is evaluated in order to numerically calculate the integral itself. Chosen point where stresses are evaluated.
GENERALISED (STRESS OR DEFORMATION)	quantity obtained by the appropriate integral summation of the point values of stress or deformation, these being organised in keeping with predictions made according to a given structural theory
GEOMETRIC NONLINEARITY	nonlinearity related to the impossibility of writing the equilibrium conditions regarding the undeformed configuration
GROUP	set of finite elements
REDUCED INTEGRATION	numeric technique of integration using fewer Gauss points than required in order to correct the overestimation of stiffness
INDEFINITE EQUILIBRIUM	condition of equilibrium imposed on a generic infinite portion of a continuum in the framework of a given structural theory
INTERELEMENT JUMPS	the difference between the stress values of a node predicted from the various elements that connect in it
INVARIANTS	quantities appertaining to the stress tensors that do not vary according to the system of reference used to describe the state of stress
ISOPARAMETRIC	a type of finite element formulation in which the shape functions are identical to the functions used to simulate the boundary
ISOTROPE	model of material behaviour that assumes identical behaviour of the material independent of the direction in which the force is

applied

JACOBIAN determinant of an appropriate matrix obtained with the derivative of the functions that relate the real to the transformed coordinates. Index of regularity and admissibility of an element

KEYWORD word or marking required in order to recognise and decipher correlated information

KIRCHHOFF PLATE theory in which the effects of shear deformation are disregarded (thin plate)

LAGRANGIAN DESCRIPTION description of the deformation that keeps a particle fixed and takes into account the positions it occupies over time

LAMÉ PARAMETER particular way of expressing the elastic constants of a homogeneous, isotropic body

LOAD CASE condition that groups actions having a common physical cause

LOAD COMBINATION a load situation obtained by combining linearly the effects of individual cases of loads appropriately weighted

LOCAL COORDINATE SYSTEM coordinate system of an element suitably oriented for describing in a simple way what takes place within the element

LOCKING numeric phenomenon by which some finite elements lead to abnormal overestimation of stiffness in certain conditions

LUMPED APPROACH technique that consists in attributing to the nodes an element of force or mass related to distributed loads or mass densities, evaluated by means of simplified, elementary schemes that are not in keeping with the assumptions made about the displacement field

MACRO a set of instructions followed automatically beginning with a few initial parameters (or even with no parameters at all)

MASTER-SLAVE relationship between two nodes (master and slave) such that when the displacements of the master are known, the displacements of the slave are wholly determined

MATERIAL NONLINEARITY nonlinearity related to the constitutive law of non-elastic material

MECHANISM condition where a degree of freedom can assume non-null values without there being any deformation energy

MEMBRANE continuous two-dimensional plane in which the stresses all lie in the plane itself and are constant within the thickness

MEMBRANE SYSTEM system of typical stresses in a membrane

MESH set of nodes in a FEM model with all the connectivity tables of various elements that make up the structure

MINDLIN PLATE plate modelling the effects of shear deformations (thick plate)

MIRRORING (REFLECTION) procedure with which a certain set of objects is copied symmetrically with respect to a plane, called the plane of reflection

MULTIPOINT CONSTRAINT (MPC) constraining link between degrees of freedom and various nodes of the structure

NAMING procedure in which the nodes and elements of a model are given an alphanumeric symbol in order to distinguish them from others independent of the progressive number assigned to them

NAVIER-STOKES EQUATIONS	Partial differential equations that formally solve elastic problems formulated with unknowns in the displacements
NODAL LOAD	load applied directly to a node
NODE	a point in a model identified by three coordinates and having a given number of degrees of freedom (usually three translations and three rotations)
OBJECT ORIENTED PROGRAMMING	programming technique that considers the execution of a program as the interaction between a given large number of abstract objects
ORTHOTROPE	model of the behaviour of a material admitting three distinct principal directions to which various values of elastic constants are related
OUT OF CORE	circumstance where part of the information required for a calculation has to be written onto the hard disk because the complete set will not fit within avaible RAM
PATCH TEST	test aimed at guaranteeing that a certain kind of finite element is capable of modelling constant states of deformation and rigid body motions
PLANE STRAIN	simplified theory of the problem of elasticity that assumes to be null longitudinal deformation normal to a plane that is repeatedly equal indefinitely for geometry and loads
PLANE STRESS	simplified theory of the problem of elasticity that assumes to be null the normal stresses perpendicular to a thin planar lamina
PLATE ELEMENT	finite plate element (thin or thick). Also called plate-shell because it can be used to model shell structures
POISSON'S RATIO	numeric coefficient between 0 and 0.5 that quantifies the transverse contraction of a body in tensile stress
PRINCIPAL STRESSES	maximum and minimum values of stresses normal to a point obtained by varying the angle of the plane being analysed
PROCEDURAL PROGRAMMING	technique of programming that considers the execution of a program as the ordered performance of a series of specific instructions
PSEUDOMECHANISM	circumstance such that a non-necessary degree of freedom is included in the model without being constrained
REDUCED INTEGRATION	numeric technique of integration using fewer Gauss points than required in order to correct the overestimation of stiffness
REDUCTION	see *triangulation*
RENUMBERING	procedure in which the numbering assigned to the nodes of a structure is completely redone in order to minimise bandwidth
RESIDUALS	small, unbalanced forces acting on nodes as a consequence of inevitable errors in the numeric solution to the problem
RIGID BODY MOTION	motion of a structure or one of its parts that does not necessitate any internal work (and consequently free motion)
RIGID LINK	constraint that imposes the invariability of mutual distances between two or more points
SELECTIVE INTEGRATION	numeric technique that integrates to a different degree of precision various components of a stiffness matrix, in order to reduce locking without the introduction of too many spurious

	modes
SELF-BALANCED	systems in which the set of applied forces is such that it is already balanced without requiring the addition of any constraining reaction
SHAPE FUNCTION	function intended to predict the unknown displacement field within a finite element
SHEAR FACTOR	number used to correct the energy of shear deformation such that when a constant distribution of shear stress is assumed an "exact" value of strain energy is obtained
SHELL	continuous two-dimensional surface of moderate thickness
SHELL ELEMENT	finite shell element, usually available in advanced programs
SINGULARITY	circumstance where a stiffness matrix has a null determinant. Area of the structure where the stresses or the deformation assume values that are theoretically unlimited
SINGULAR VALUE DECOMPOSITION (SVD)	technique used to extract solutions corresponding to the singularity of a non-invertible matrix
SKYLINE	profile of band length variability of various rows (or columns) of a stiffness matrix
SMOOTHING	numeric technique used to regularise or homogenise the area of stresses calculated through an analysis of the finite elements
SOLVER	program used for solving systems of equations associated with a finite element problem and the calculus of stresses
SOLVING	numeric procedure used for calculating the solution of a system of solution equations
SPLINE	kind of interpolation using polynomials at intervals having requisites of continuity between one interval and the next
SPRING ELEMENT	finite spring element
SPURIOUS MODES	abnormal states of deformation with null deformation energy, made possible by certain finite elements with reduced or selective integration (for example, hourglass mode)
SRI	Selective reduced integration
STATIONARITY	circumstance where a numeric quantity dependent on a function does not vary when a small variation is made in the function
STIFFNESS, DIRECT	stiffness demonstrated by a node when it is moved
STIFFNESS, GEOMETRIC	added (positive or negative) stiffness related to the motion of a body subject to a state of initial stress (stressed rods, cables)
STIFFNESS, INDIRECT	stiffness demonstrated by a node held still when it is moved by another node
STIFFNESS, MATRIX OF	a square matrix that summarises the elastic behaviour of the finite element model
STRAIN ENERGY	the quantity of energy stored in a solid due to a variation in its configuration
STRESS RECOVERY	procedure used to calculate the stresses beginning with the set of nodal displacements
SUBSTRUCTURE	part of a structure that is modelled separately and then assimilated through the use of appropriate techniques, to a kind of great

"finite element" described by means of a number of nodes that is distinctly inferior to that actually used

TIMOSHENKO BEAM — beam theory that takes account of the energy of shear deformation (thick beams)

TOTAL POTENTIAL ENERGY — sum of the deformation energy and the load potentials

TRANSFORMATION — technique used to place a domain referring to one coordinate system in correspondence with another domain referring to a different coordinate system

TRIANGULARIZATION — procedure with which a square matrix is decomposed into the product of two triangular matrices

TRUSS (ELEMENT) — finite truss element (models only the axial behaviour of a beam), hinged by definition at the extreme nodes

UNIFORM TORSION — kind of torsion that leads to identical warping in all sections, that is, free displacements in the direction of the axis of the prism

VECTOR ID — a vector that places the number of a degree of freedom in correspondence with the node number to which it appertains

WINKLER FOUNDATION — schematisation in which the foundation is likened to a bed of independent, elastic springs whose stiffness depends on the characteristics of the terrain and on the surfaces in contact with it

18 REFERENCES

BATHE, K. J. 1996. *Finite Element Procedures.* Prentice Hall.

BATOZ, J. L. K. J. BATHE, L. W. HO. 1980. A study of Three-Node Triangular Plate Bending Elements. *International Journal of Numerical Methods in Engineering* 15: 1771-1812.

BATOZ, J. L., M. BEN TAHAR. 1982. Evaluation of a New Quadrilateral Thin Plate Bending Element. *International Journal of Numerical Methods in Engineering* 18: 1655-1677.

BREBBIA, C.A., J. J. CONNOR. 1978. *Fondamenti del Metodo degli Elementi Finiti.* CLUP.

Economist. 2003. Building a Better Bug Trap. *The Economist* 367, 8329 (June 21-27 2003).

CARR, A. J. 1967. A Refined Element Analysis of Thin Shell Structures Including Dynamic Loadings. SEL Report number 67-9. University of California, Berkeley.

CESARI, F. 1997. *Codici di calcolo per l'analisi di strutture spaziali.* 2nd ed. Pitagora Editrice.

COOK, R. D., D.S. MALKUS, M.E. PLESHA, R. J. WITT. 2001. *Concepts and Applications of Finite Element Analysis.* 4th ed. John Wiley & Sons.

CORRADI DELL'ACQUA, L. 1992. *Meccanica delle Strutture.* Vol. 1: "Il comportamento dei mezzi continui" and Vol. 2: "Le teorie strutturali ed il metodo degli elementi finiti". McGraw-Hill.

DE VEUBEKE, F. 1968. A Conforming Finite Element Approach to Plate Bending. *Int. J. Solids Struct.* 4: 95-108.

D.M. 14-2-1992. *Norme tecniche per l'esecuzione delle opere in cemento armato normale e precompresso.*

DOWLING, N. E. 1993. *Mechanical Behavior of Materials.* Prentice-Hall International Editions.

DVORKIN, E. N., K. J. BATHE. 1984. A Continuum Mechanics Based Four-Node Shell Element for General Nonlinear Analysis. *Eng. Comput* 1 (March 1984): 77-88.

EMKIN, L. Z. 1998. Misuse of computers by structural engineers: a clear and present danger. *Proceedings of the 1st Structural Engineers World Congress* (SEWC 98), San Francisco, California, USA, 19-23 July 1998.

Structural analysis with finite elements
978-0-7277-4093-9

GORI, R., ed. 2001. *Crolli e Affidabilità delle Strutture Civili.* Proceedings of the National Convention organised by Angelo Di Tommaso and Enzo Siviero, Venice, 6-7 December 2001. Edizioni Libreria Cortina, Padua.

HUGHES, T. J. R., T. E. TEZDUYAR. 1981. Finite Element Based upon Mindlin Plate Theory. *Journal of Applied Mechanics* 48: 587-596.

LANDAU, L. D., E. M. LIFSHITZ. 1979. *Teoria della Elasticità.* Editori Riuniti, Edizioni Mir.

MACNEAL, R. H. 1994. *Finite Elements: Their Design and Performance.* Marcel Dekker Inc.

MASE, G. E. 1999. *Continuum Mechanics for Engineers.* 2nd ed. CRC.

"Primi elementi in materia di criteri generali per la classificazione sismica del territorio nazionale e di normative tecniche per le costruzioni in zona sismica", Ordinanza 3274 del 20/3/2003, Supplemento Ordinario alla Gazzetta Ufficiale n. 105, 8 maggio 2003.

RUGARLI, P. *C.E.S.CO. Computer Education to Structural Constructions.* Software for plane frames, part of E.Str.A.D.A., Education in Structural Assisted Design and Analysis. Produced by Castalia srl in collaboration with the Department of Structural Engineering of the Politecnico di Milano.

RUGARLI, P. 2003. Software di calcolo. *Manuale dell'Ingegnere*, S. 4, 84th ed. Hoepli.

Saint-Gobain Vetro Italia. 1997. *Manuale tecnico del vetro.*

SAINT-VENANT, Adhémar Jean Claude Barré de. 1855. Memoire sur la Torsion des Prismes. *Mémoires des Savants Etrangers*", vol. 14, pp. 233-560.

STRICKLIN, J. A., W. HAISLER, P. TISDALE, R. GUNDERSON. 1969. A rapidly converging triangular plate element. *AIAA J.* **7**(1): 180-181.

TAYLOR, R. L., J. C. SIMO. 1985. *Bending and Membrane Elements for Analysis of Thick and Thin Shells.* Proceedings of the NUMETA Conference, Swansea, Jan. 7-11, 1985.

TIMOSHENKO, S. P., J. N. GOODIER. 1970. *Theory of Elasticity.* McGraw Hill International Ed.

TIMOSHENKO, S. P., S. WOINOWSKY-KRIEGER. 1959. *Theory of plates and shells.* 2nd ed. McGraw-Hill.

TONIOLO, G. 1982. *Analisi strutturale con elaboratore elettronico.* Masson Italia.

WILSON, E. L. "Dynamic Analysis Using Response Spectrum Loading". Computers and Structures, Inc. Web Site.

WILSON, E. L., A. DER KIUREGHIAN, E. R. BAYO. 1981. A replacement for the SRSS Method in Seismic Analysis. *Earthquake Engineering and Structural Dynamics*, vol. 9: 187-192.

WILSON, E.L., A. IBRAHIMBEGOVIC. 1990. Use of incompatible displacement modes for the calculation of element stiffnesses and stresses. *Finite Elements in Analysis and Design* 7: 229-241.

ZIENKIEWICZ, O.C., R. L. TAYLOR, J.Z. ZHU. 2005. *The Finite Element Method.* 3 vols. 6th ed. Elsevier-Butterworth-Heinemann. (First ed. 1967).