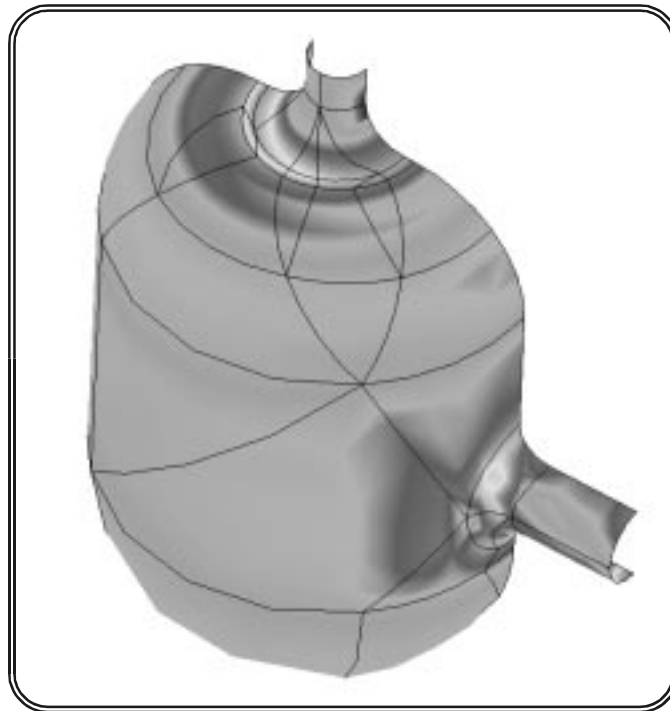


# ***Pro/MECHANICA Tutorial Structure***

**Release 2001 - Integrated Mode**

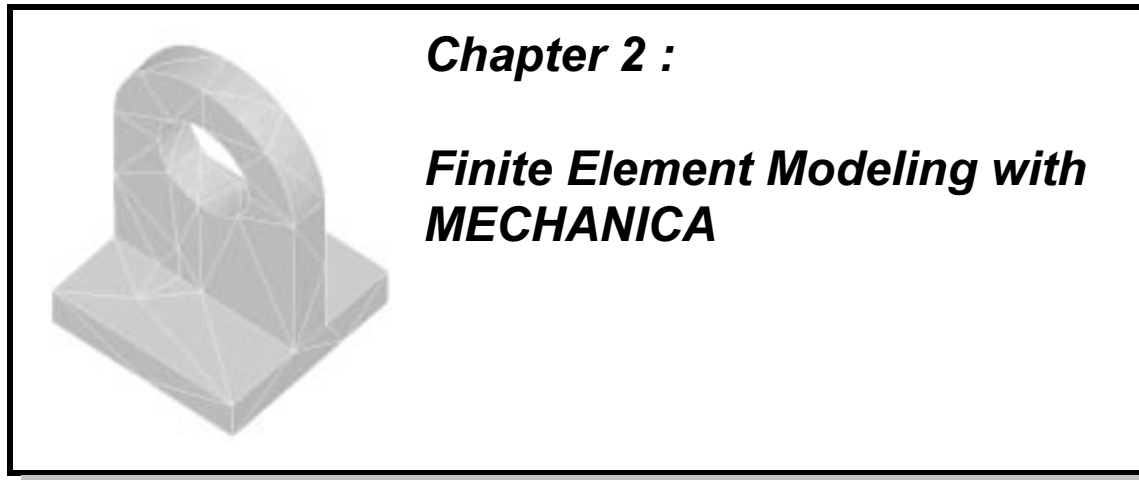


Roger Toogood, Ph.D., P. Eng.  
Mechanical Engineering  
University of Alberta

**SDC**  
PUBLICATIONS

Schroff Development Corporation

[www.schroff.com](http://www.schroff.com)



## Synopsis

Background information on FEA. The concept of modeling. Particular attention is directed at concerns of accuracy and convergence of solutions, and the differences between h-code and p-code FEA. Overview of MECHANICA.

## Overview of this Lesson

This chapter presents an overall view of FEA in general, and discusses a number of ideas and issues involved. The major differences between Pro/M, which uses a p-code method, and other packages, which typically use h-code, are presented. The topics of accuracy and convergence are discussed. The major sections in this chapter are:

- ◆ overview and origins of FEA
- ◆ discussion of the concept of the “model”
- ◆ general procedure for FEA solutions
- ◆ FEA models versus CAD models
- ◆ p-elements and h-elements
- ◆ convergence and accuracy
- ◆ sources of error
- ◆ overview of MECHANICA

Although you are probably anxious to get started with the software, your understanding of the material presented here is very important. We will get to the program soon enough!

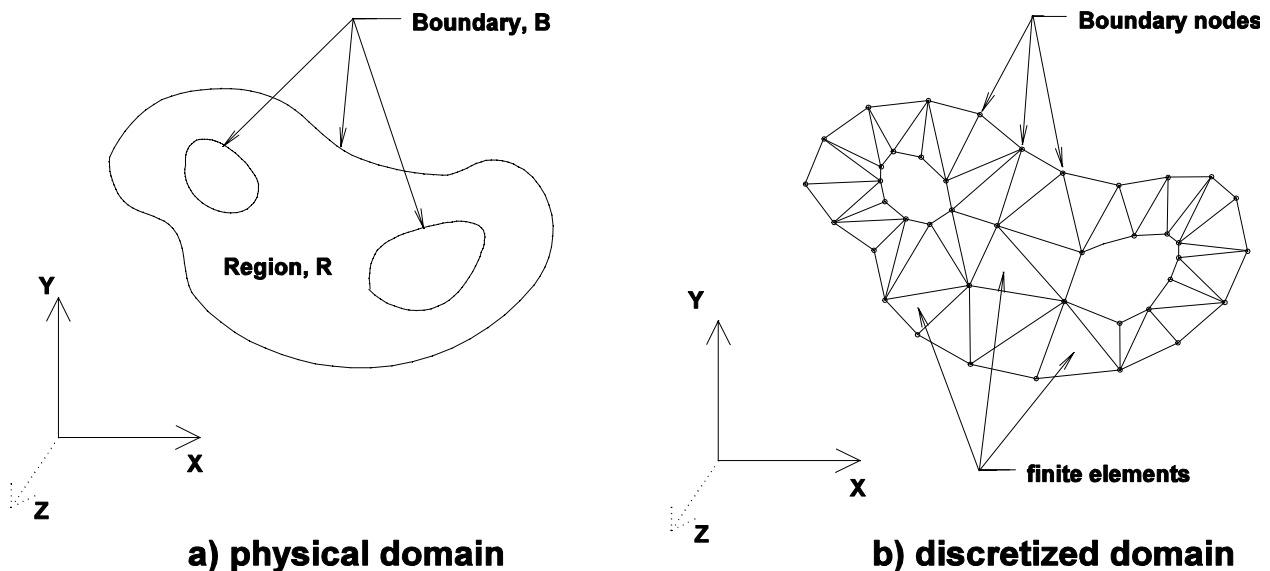
---

## Finite Element Analysis : An Introduction

In this section, we will try to present the essence of FEA without going into a lot of mathematical detail. This is primarily to set up the discussion of the important issues of accuracy and convergence later in the chapter. Some of the statements made here are generalizations and over-

simplifications, but we hope that this will not be too misleading. Interested users can consult a number of text and reference books (some are listed at the end of this chapter) which describe the theoretical underpinnings of FEA in considerably greater detail.

In the following, the ideas are illustrated using a planar (2D) solution region, but of course these ideas extend also to 3D. Let's suppose that we are faced with the following problem: We are given a connected region (or volume)  $R$  with a boundary  $B$  as shown in Figure 1(a). Some continuous physical variable, e.g. temperature  $T$ , is governed by a physical law within the region  $R$  and subjected to known conditions on the boundary  $B$ . In a finite element solution, the geometry of the region is typically generated by a CAD program, such as Pro/ENGINEER.



**Figure 1** The problem to be solved is specified in a) the physical domain and b) the discretized domain used by FEA

For a two dimensional problem, the governing physical law or principle might be expressed by a partial differential equation (PDE), for example<sup>1</sup>:

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

that is valid in the interior of the region  $R$ . The solution to the problem must satisfy some boundary conditions or *constraints*, for example  $T = T(x,y)$ , prescribed on the boundary  $B$ . Both interior and exterior boundaries might be present and can be arbitrarily shaped. Note that this governing PDE may be (and usually is!) the result of simplifying assumptions made about the

---

<sup>1</sup> The PDE given represents the temperature within a solid body which is governed by the conduction of heat within the body. There are no heat sources, and temperature on the boundary of the body is known.

physical system, such as the material being homogeneous and isotropic, with constant linear properties, and so on.

In order to analyze this problem, the region  $R$  is *discretized* into individual *finite elements* that collectively approximate the shape of the region, as shown in Figure 1(b). This discretization is accomplished by locating *nodes* along the boundary and in the interior of the region. The nodes are then joined by lines to create the finite elements. In 2D problems, these can be triangles or quadrilaterals; in 3D problems, the elements can be tetrahedra or 8-node "bricks". In some FEA software, other higher order types of elements are also possible (e.g. hexagonal prisms). Some higher order elements also have additional nodes along their edges. Collectively, the set of all the elements is called a *finite element mesh*. In the early days of FEM, a great deal of effort was required to set up the mesh. More recently, automatic meshing routines have been developed in order to do most, if not all, of this tedious task.

In the FEA solution, values of the dependent variable ( $T$ , in our example) are computed only at the nodes. The variation of the variable within each element is computed from the nodal values so as to approximately satisfy the governing PDE. One way of doing this is by using interpolating polynomials. In order for the PDE to be satisfied, the nodal values of each element must satisfy a set of conditions represented by several linear algebraic equations usually involving other nodal values.

The boundary conditions are implemented by specifying the values of the variables on the boundary nodes. There is no guarantee that the true boundary conditions on the continuous boundary  $B$  are satisfied between the nodes on the discretized boundary.

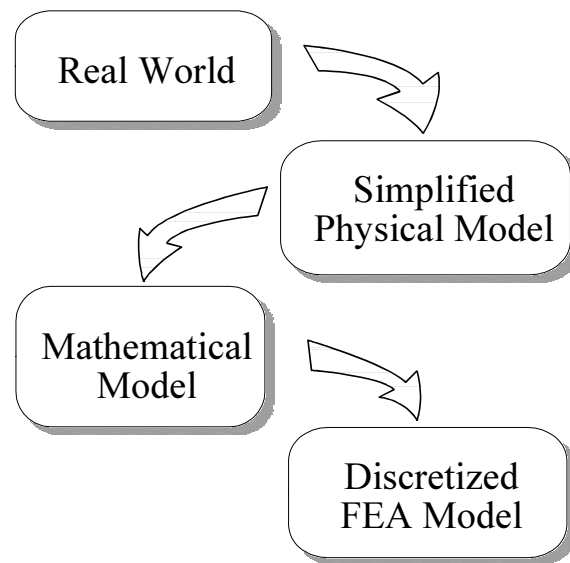
When all the individual elements in the mesh are combined, the discretization and interpolation procedures result in a conversion of the problem from the solution of a continuous differential equation into a very large set of simultaneous linear algebraic equations. This system can typically have many thousands of equations in it, requiring special and efficient numerical algorithms. The solution of this algebraic system contains the nodal values that collectively represent an *approximation* to the continuous solution of the initial PDE. An important issue, then, is the accuracy of this approximation. In classical FEM solutions, the approximation becomes more accurate as the mesh is refined with smaller elements. In the limit of zero mesh size, requiring an infinite number of equations, the FEM solution to the PDE would be exact. This is, of course, not achievable. So, a major issue revolves around the question "How fine a mesh is required to produce answers of acceptable accuracy?" and the practical question is "Is it feasible to compute this solution?" We will see a bit later how Pro/M solves these problems.

**IMPORTANT POINT:** In FEA stress analysis problems, the dependent variable in the governing PDE's is the displacement from the reference (usually unloaded) position. The material strain (displacement per unit length) is then computed from the displacement by taking the derivative with respect to position. Finally, the stress components at any point in the material are computed from the strain at that point. Thus, if the interpolating polynomial for the spatial variation of the displacement field is linear within an element, then the strain and stress will be constant within that element, since the derivative of a linear function is a constant. The significance of this will be illustrated a bit later in this lesson.

## The FEA Model and General Processing Steps

Throughout this manual, we will be using the term “model” extensively. We need to have a clear idea of what we mean by the FEA model.

To get from the “real world” physical problem to the approximate FEA solution, we must go through a number of simplifying steps. At each step, it is necessary to make decisions about what assumptions or simplifications will be required in order to reach a final workable model. By “workable”, we mean that the FEA model must allow us to compute the results of interest (for example, the maximum stress in the material) with sufficient accuracy and with available time and resources. It is no good building a model that is over-simplified to the point where it cannot produce the results with sufficient accuracy. It is also no good producing a model that is “perfect” but will not yield useful computational results for several weeks! Quite often, the FEA user must compromise between the two extremes - accepting a slightly less accurate answer in a reasonable solution time.



**Figure 2** Developing a Model for Finite Element Analysis

To arrive at a model suitable for FEA, we must go through the simplifying steps shown in Figure 2, as follows:

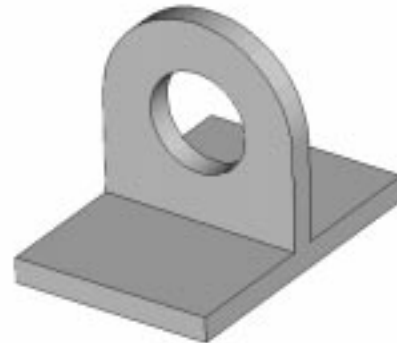
### **Real World → Simplified Physical Model**

This simplification step involves making assumptions about physical properties or the physical layout and geometry of the problem. For example, we usually assume that materials are homogeneous and isotropic and free of internal defects or flaws. It is also common to ignore aspects of the geometry that will have no (anticipated) effect on the results, such as the chamfered and filleted edges on the bracket shown in Figure 3, and perhaps even the mounting

holes themselves. Ignoring these “cosmetic” features, as shown in Figure 4, is often necessary in order to reduce the geometric complexity so that the resulting FEA model is practical.



**Figure 3** The “Real World” Object



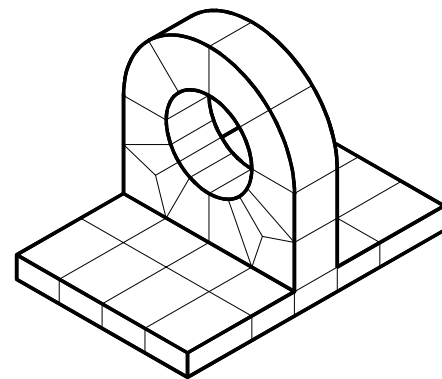
**Figure 4** The idealized physical model

### Simple Physical Model → Mathematical Model

To arrive at the mathematical model, we make assumptions like linearity of material properties, idealization of loading conditions, and so on, in order to apply our mathematical formulas to complex problems. We often assume that loading is steady, that fixed points are perfectly fixed, beams are long and slender, and so on. As discussed above, the mathematical model usually consists of one or more differential equations that describe the variation of the variable of interest within the boundaries of the model.

### Mathematical Model → FEA Model

The simplified geometry of the model is discretized (see Figure 5), so that the governing differential equations can be rewritten as a (large) number of simultaneous linear equations representing the assembly of elements in the model.



**Figure 5** A mesh of solid brick elements

In the operation of FEA software, the three modeling steps described above often appear to be merged. In fact, most of it occurs below the surface (you will never see the governing PDE, for example) or is inherent in the software itself. For example, Pro/M automatically assumes that materials are homogeneous, isotropic, and linear. However, it is useful to remind yourself about these separate aspects of modeling from time to time, because each is a potential source of error or inaccuracy in the results.

## Steps in Preparing an FEA Model for Solution

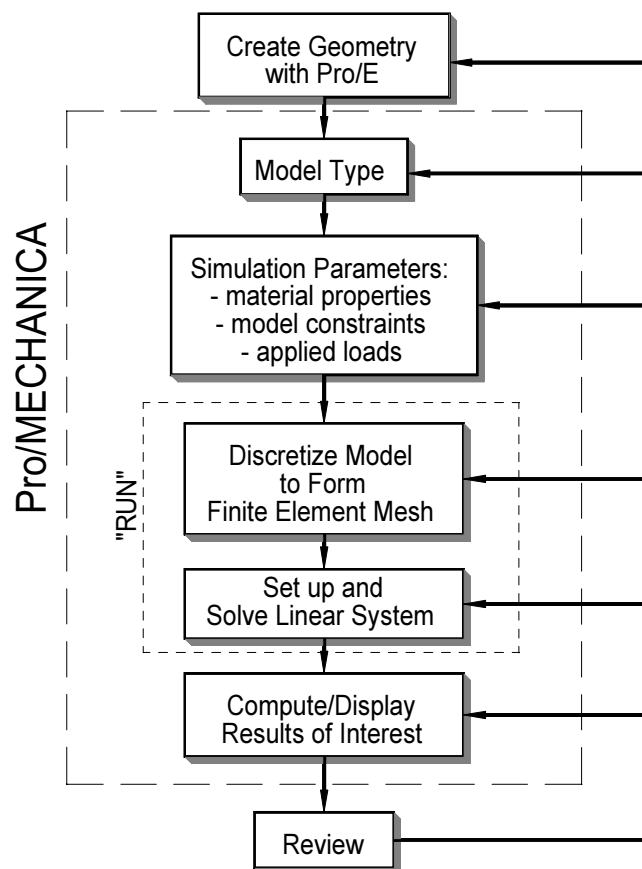
Starting from the simplified geometric model, there are generally several steps to be followed in the analysis. These are:

1. identify the model type
2. specify the material properties, model constraints, and applied loads
3. discretize the geometry to produce a finite element mesh
4. solve the system of linear equations
5. compute items of interest from the solution variables
6. display and critically review results and, if necessary, repeat the analysis

The overall procedure is illustrated in Figure 6. Some additional detail on each of these steps is given below. The major steps must be executed in order, and each must be done correctly before proceeding to the next step. When a problem is to be re-analyzed (for example, if a stress analysis is to be performed for the same geometry but different loads), it will not usually be necessary to return all the way to the beginning. The available re-entry points will become clear as you move through these tutorials.

The steps shown in the figure are:

1. The geometric model of the part/system is created using Pro/ENGINEER.
2. On entry to Pro/M, the model type must be identified. The default is a solid model.
3.
  - A) Specify material properties for the model. It is not necessary that all the elements have the same properties. In an assembly, for example, different parts can be made of different materials. For stress analysis the required properties are Young's modulus and Poisson's ratio. Most FEA packages contain built-in libraries containing properties of common materials (steel, iron, aluminum, etc.).
  - B) Identify the constraints on the solution. In stress analysis, these could be fixed points, points of specified displacement, or points free to move in specified directions only.
  - C) Specify the applied loads on the model (point loads, uniform edge loads, pressure on surfaces, etc.).



**Figure 6** Overall steps in FEA Solution

4. Once you are satisfied with your model, you set up and run a processor that actually performs the solution to the posed FEA problem. This starts with the automatic creation of the finite element mesh from the geometric model by a subprogram within Pro/M called AutoGEM. Pro/M will trap some modeling errors here. The processor will produce a summary file of output messages which can be consulted if something goes wrong - for example, a model that is not sufficiently constrained by boundary conditions.
5. FEA produces immense volumes of output data. The only feasible way of examining this is graphically. Pro/M has very powerful graphics capabilities to examine the results of the FEA - displaced shape, stress distributions, mode shapes, etc. Hard copy of the results file and screen display is easy to obtain.
6. Finally, the results must be reviewed critically. In the first instance, the results should agree with our modeling intent. For example, if we look at an animated view of the deformation, we can easily see if our boundary constraints have been implemented properly. The results should also satisfy our intuition about the solution (stress concentration around a hole, for example). If there is any cause for concern, it may be advisable to revisit some aspects of the model and perform the analysis again.

---

## P-Elements versus H-Elements

Not all discretized finite elements are created equal! Here is where a major difference arises between MECHANICA and most other FEA programs.

### Convergence of H-elements (the “classic” approach)

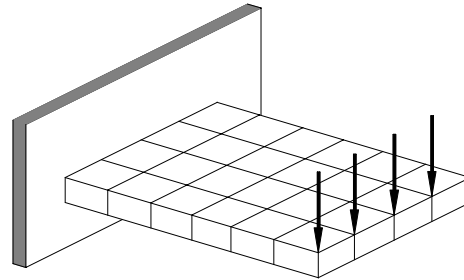
Following the classic approach, other programs often use low order interpolating polynomials in each element. This has significant ramifications, especially in stress analysis. As mentioned above, in stress analysis the primary solution variables are the displacements of the nodes. The interpolating functions are typically linear (first order) within each element. Strain is obtained by taking the derivatives of the displacement field and the stress is computed from the material strain. For a first order interpolating polynomial within the element, this means that the strain and therefore the stress components within the element are constant everywhere. The situation is depicted in Figure 7, which shows the computed Von Mises stress in each of the elements surrounding a hole in a thin plate under tension. Such discontinuity in the stress field between elements is, of course, unrealistic and will lead to inaccurate values for the maximum stress. Low order elements lead to the greatest inaccuracy precisely in the regions of greatest interest, typically where there are large gradients within the real object.

An even more disastrous situation is shown in Figure 8. This is a solid cantilever beam with a uniform transverse load modeled using solid brick elements. With only a single first-order element through the thickness, the computed stress will be the same on the top and bottom of the beam. This is clearly wrong, yet the FEA literature and product demonstrations abound with examples similar to this.





**Figure 7** Von Mises stress in 1/4 model of thin plate under tension using first order elements

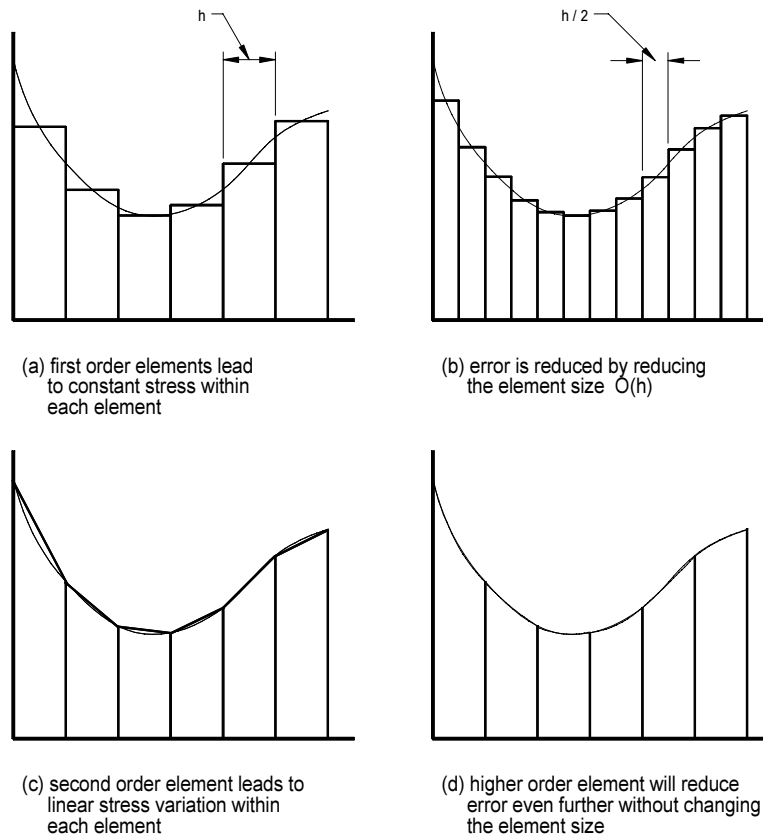


**Figure 8** A disaster waiting to happen using first order elements

This situation is often masked by the post-processing capabilities of the software being used, which will sometimes average or interpolate contour values within the mesh or perform other “smoothing” functions strictly for visual appearance. This is strictly a post-processing step, and may bear no resemblance at all to what is actually going on in the model or the real object.

Using first order elements, then, in order to get a more accurate estimate of the stress, it is necessary to use much smaller elements, a process called *mesh refinement*. It may not always be possible to easily identify regions where mesh refinement is required, and quite often the entire mesh is modified. The process of mesh refinement continues until further mesh division and refinement does not lead to significant changes in the obtained solution. The process of continued mesh refinement leading to a “good” solution is called *convergence analysis*. Of course, in the process of mesh refinement, the size of the computational problem becomes larger and larger and we may reach a limit for practical problems (due to time and/or memory limits) before we have successfully converged to an acceptable solution.

The use of mesh refinement for convergence analysis leads to the *h-element* class of FEA methods. This “h” is borrowed from the field of numerical analysis, where it denotes the fact that convergence and accuracy are related (sometimes proportional to) the step size used in the solution, usually denoted by  $h$ . In FEA, the  $h$  refers to the size of the elements. The elements, always of low order, are referred to as *h-elements*, and the mesh refinement procedure is called *h-convergence*. This situation is depicted in parts (a) and (b) of Figure 9, where a series of constant-height steps is used to approximate a smooth continuous function. The narrower the steps, the more closely we can approximate the smooth function. Note also that where the gradient of the function is large (such as near the left edge of the figure), then mesh refinement will always produce increasingly higher maximum values.



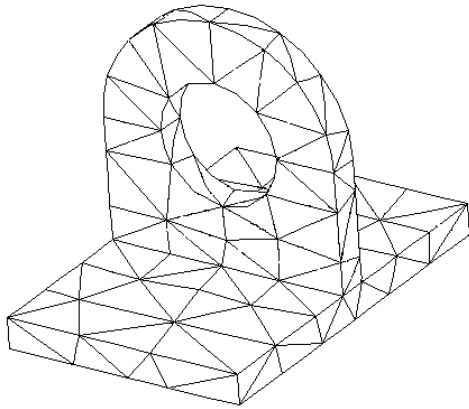
**Figure 9** Approximation of stress function in a model

The major outcome of using h-elements is the need for meshes of relatively small elements. Furthermore, h-elements are not very tolerant of shape extremes in terms of skewness, rapid size variation through the mesh, large aspect ratio, and so on. This further increases the number of elements required for an acceptable mesh, and this, of course, greatly increases the computational cost of the solution.

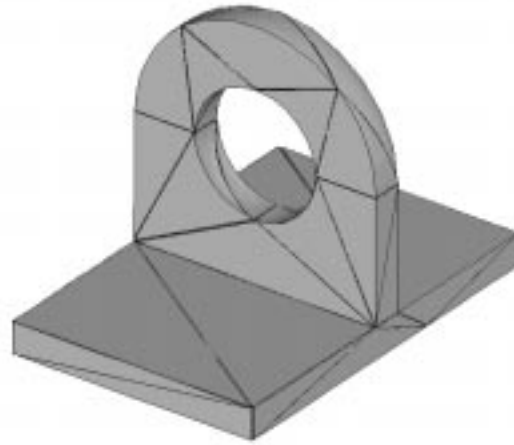
### Convergence of P-elements (the Pro/MECHANICA approach)

Now, the major difference incorporated in MECHANICA is the following: instead of constantly refining and recreating finer and finer meshes, convergence is obtained by *increasing the order of the interpolating polynomials on each element*. The mesh stays the same for every iteration, called a *p-loop pass*. The use of higher order interpolating polynomials for convergence analysis leads to the *p-element* class of FEA methods, where the “p” denotes polynomial. This method is depicted in parts (c) and (d) of the Figure 9. Only elements in regions of high gradients are bumped up to higher order polynomials. Furthermore, by examining the effects of going to higher order polynomials, MECHANICA can monitor the expected error in the solution, and automatically increase the polynomial order only on those elements where it is required. Thus, the convergence analysis is performed quite automatically, with the solution proceeding until an accuracy limit (set by the user) has been satisfied. With MECHANICA, the limit for the polynomial order is 9. In theory, it would be possible to go to higher orders than this, but the computational cost starts to rise too quickly. If the solution cannot converge even with these 9<sup>th</sup>

order polynomials, it may be necessary to recreate the mesh at a slightly higher density so that lower order polynomials will be sufficient. This is a very rare occurrence.



**Figure 10** A mesh of solid tetrahedral (4 node) h-elements



**Figure 11** A mesh of tetrahedral p-elements produced by MECHANICA.

The use of p-elements has a number of features/advantages:

- ▶ The same mesh can be used throughout the convergence analysis, rather than recreating meshes or local mesh refinement required by h-codes.
- ▶ The mesh is virtually always more coarse and contains fewer elements than h-codes. Compare the meshes in Figures 10 and 11, and note that the mesh of h-elements in Figure 10 would probably not produce very good results, depending on the loads and constraints applied. The reduced number of elements in Pro/M (which can be a couple of orders of magnitude smaller) initially reduces the computational load, but as the order of the polynomials gets higher, this advantage is somewhat diminished.
- ▶ The restrictions on element size and shape are not nearly as stringent for p-elements as they are for h-elements (where concerns of aspect ratio, skewness, and so on often arise).
- ▶ Automatic mesh generators, which can produce very poor meshes for h-elements, are much more effective with p-elements, due to the reduced requirements and limitations on mesh geometry.
- ▶ Since the same mesh is used throughout the analysis, this mesh can be tied directly to the geometry. This is the key reason why MECHANICA is able to perform sensitivity and optimization studies during which the geometric parameters of a body can change, but the program does not need to be constantly re-meshing the part.

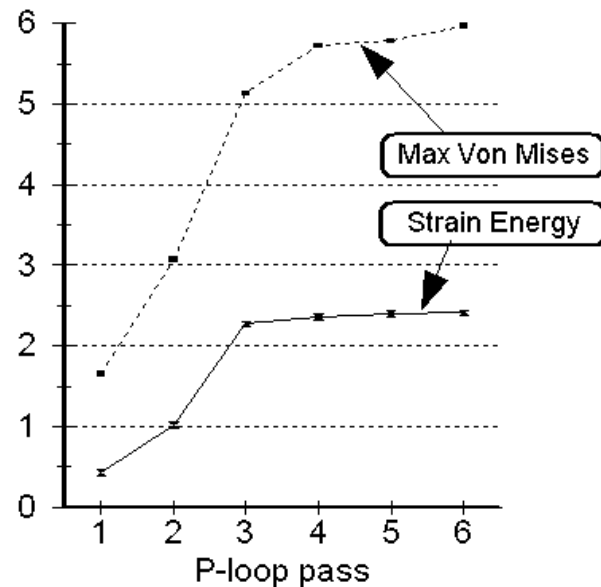
## Convergence and Accuracy in the Solution

It should be apparent that, due to the number of simplifying assumptions necessary to obtain results with FEA, we should be quite cautious about the results obtained. No FEA solution

should be accepted unless the convergence properties have been examined.

For h-elements, this generally means doing the problem several times with successively smaller elements and monitoring the change in the solutions. When decreasing the element size results in a negligible (or acceptably small) change in the solution, then we are generally satisfied that the FEA has wrung all the information out of the model that it can.

As mentioned above, with p-elements, the convergence analysis is built in to the program. Since the geometry of the mesh does not change, no remeshing is required. Rather, each successive solution (called a *p-loop pass*) is performed with increasing orders of polynomials (only on elements where this is required) until the change between iterations is “small enough”. Figure 12 shows the convergence behavior of two common measures used to monitor convergence in MECHANICA. These are the maximum Von Mises stress and the total strain energy. Note that the Von Mises stress will generally always increase during the convergence test, but can behave quite erratically as we will see later. Because Von Mises stress is a local measure, the strain energy is probably a better measure to use to control convergence.



**Figure 12** Two common convergence measures using p-elements.

## Sources of Error

Error enters into the FEA process in a number of ways:

- ◆ **errors in problem definition** - are the geometry, loads, and constraints known and implemented accurately? Is the correct analysis being performed? Are the material properties correct and/or appropriate?
- ◆ **errors in creating the physical model** - can we really use symmetry? Is the material isotropic and homogeneous, as assumed? Are the physical constants known? Does the material behave linearly?
- ◆ **errors in creating the mathematical model** - is the model complete enough to capture the effects we wish to observe? Is the model overly complex? Does the mathematical model correctly express the physics of the problem?
- ◆ **errors in discretization** - is the mesh too coarse or too fine? Have we left accidental “holes” in the model? If using shell elements, are there tears or rips (free edges) between elements where there shouldn’t be?
- ◆ **errors in the numerical solution** - when dealing with very large computational problems,

we must always be concerned about the effects of accumulated round-off error. Can this error be estimated? How trustworthy is the answer going to be?

- ◆ **errors in interpretation of the results** - are we looking at the results in the right way to see what we want and need to see? Are the limitations of the program understood<sup>2</sup>? Has the possible misuse of a purely graphical or display tool obscured or hidden a critical result?

You will be able to answer most of these questions by the time you complete this tutorial. The answers to others will be problem dependent and will require some experience and further exposure before you are a confident and competent FEA user.

---

## A CAD Model is *NOT* an FEA Model!

One of the common misconceptions within the engineering community is the equivalence of a CAD solid model with a model used for FEA. These are, in fact, not the same despite proclamations of the CAD vendors that their solid models can be “seamlessly” ported to one or another FEA program. In fact, this is probably quite undesirable! It should not be surprising that CAD and FEA models are different, since the two models are developed for different purposes.

The CAD model is usually developed to provide a data base for manufacturing. Thus, dimensions must be fully specified (including tolerances), all minor features (such as fillets, rounds, holes) must be included, processing steps and surface finishes are indicated, threads are specified, and so on. Figure 13 shows a CAD solid model of a hypothetical piping component, complete with bolt holes, flanges, o-ring grooves, chamfered edges, and carrying lugs. Not visible in the figure are the dimensions, tolerances, and welding instructions for fabrication which are all part of the CAD model.



**Figure 13** A hypothetical 3D solid model of a piping junction

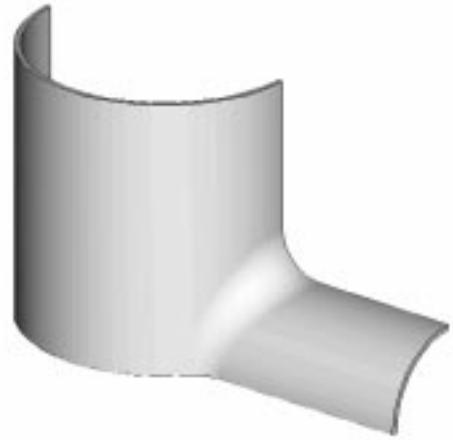
FEA is usually directed at finding out other information about a proposed design. To do this *efficiently*, the FEA model can (and often needs to) be quite different from the CAD model. A simple example of this is that the symmetry of an object is often exploited in the preparation of the FEA model. In one of the exercises we will do later, we will model a thin tapered plate with a couple of large holes. The plate has a plane of symmetry so that we only need to do FEA of one-half of the plate. It is also

---

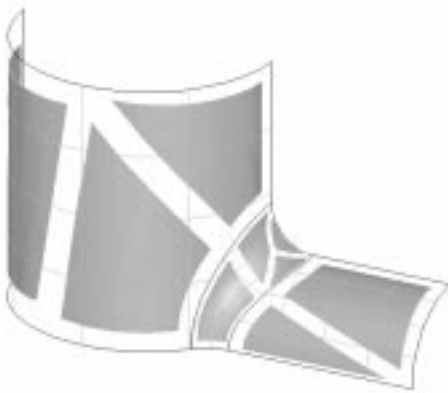
<sup>2</sup> The author once had a student who was rightly concerned about the very large deflections in a truss computed using a simple FEM program. It turned out that the program was performing a linear analysis, and was computing stresses in some members several orders of magnitude higher than the yield strength of the material. It took some time to explain that the FEM software knew nothing about failure of the material. It turned out that a simple data entry error had reduced the cross sectional area of the members in the truss.

quite common in FEA to ignore minor features like rounds, fillets, chamfers, holes, minor changes in surface profile, and other cosmetic features unless these features will have a large effect on the measures of interest in the model. Most frequently, they do not, and can be ignored.

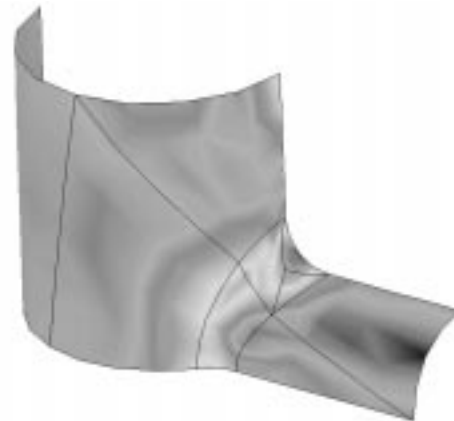
Figure 14 shows an FEA model of the piping component created to determine the maximum Von Mises stress in the vicinity of the filleted connection between the two pipes. The differences between the two models shown in Figures 13 and 14 are immediately obvious. Figures 15 and 16 show the mesh of shell elements created from the surface, and the computed Von Mises stress.



**Figure 14** The 3D solid model of pipe junction



**Figure 15** Shell elements of specified thickness created from 3D model



**Figure 16** Von Mises stress in the FEA model

In summary, the stated goal of FEA (the “Golden Rule”, if you like) might be expressed as:

***Use the simplest model possible that will yield sufficiently reliable results of interest at the lowest computational cost.***

You can easily see how this might be at odds with the requirements of a CAD model. For further discussion of this, see the excellent book **Building Better Products with Finite Element Analysis** by Vince Adams and Abraham Askenazi, Onword Press, 1998.

---

## Overview of Pro/MECHANICA Structure

### Basic Operation

We are going to start using Pro/M in the next chapter. Before we dive in, it will be useful to have an overall look at the function and organization of the software. This will help to explain some of the Pro/M terminology and see how the program relates to the ideas presented in this chapter's overview of FEA.

We can divide the operation and functionality of Pro/M Structure according to the rows in Table I below. These entries are further elaborated in the next few pages. In the process of setting up and running a solution, you will basically need to pick one option from each row in the table. The top-down organization of the table is roughly in the order that these decisions must be made. Other issues such as creation of the model geometry and post-processing and display of final results will be left to subsequent chapters.

**TABLE I - An Overall View of Pro/M Capability and Function**

	<b>MECHANICA Options</b>	<b>Description</b>
<b>Mode of Operation</b>	Independent Integrated	how Pro/M is operated with respect to Pro/ENGINEER
<b>Type of Model</b>	3D Plane Stress Plane Strain Axisymmetric	basic structure of the model
<b>Type of Elements</b>	Shell Beam Solid Spring Mass	element types that can be used in a model
<b>Analysis Methods</b>	Static Modal Buckling Pre-stress modal Pre-stress buckling	the fundamental solution being sought for the model
<b>Convergence Methods</b>	Quick Check Single Pass Adaptive Multi-Pass Adaptive	method of monitoring convergence in the solution
<b>Design Studies</b>	Standard Sensitivity Optimization	high level methods to organize essentially repetitive computations

## Modes of Operation

A discussion of the full details of operating modes gets pretty confusing, so only the main points are presented here. These are:

1. Pro/M can operate in two modes<sup>3</sup>, in relation to its cousin application Pro/ENGINEER. These are: **independent** and **integrated**. A special license is required to run the independent version. In the student edition, only integrated mode is possible.
2. The user interface is determined by the mode:
  - ◆ integrated mode - Pro/ENGINEER interface
  - ◆ independent mode - Pro/MECHANICA interface
3. If you start out in Pro/ENGINEER to create the part (or assembly) geometry and call up MECHANICA, you will initially be running in integrated mode. You can then switch to independent mode if desired (and if your license allows it), as illustrated here (note that the arrow is a one-way transfer - you can't get back again!):

Integrated    →    Independent  
Mode                    Mode

4. If you switch to independent mode, the connection with Pro/ENGINEER will be severed. Any changes in design parameters (for example following an optimization) must be manually transferred back into the Pro/E model.
5. In integrated mode, a few Pro/M commands and result displays are not available. However the tight integration with Pro/E makes it very easy to perform design modification and quick FEA.
6. In integrated mode, the user interface is the same as Pro/E. Only one set of controls to learn! The independent mode user interface is quite different.
7. The full set of Pro/M commands and functions are available in independent mode (for example: display of some types of results such as element p-levels, manual and semi-automatic mesh generation for difficult models).
8. Although independent mode gives access to the complete range of MECHANICA functionality, the benefits of feature-based geometry creation/modification are lost.

A condensed comparison of these operating modes is shown in Table II on the next page. As mentioned above, all the tutorials in this manual are meant to be run in integrated mode.

---

<sup>3</sup> A third mode, called *linked*, was available up until Release 2000i, but has been removed.



TABLE II - Pro/MECHANICA Modes of Operation

Integrated Mode	Independent Mode
Pro/E interface	Pro/M interface
all analyses available	all analyses available
2D and 3D models	2D and 3D models
some measures of results not available	all measures available
some analysis options not available (eg excluding elements)	all options available
all elements generated automatically	element creation manual or automatic
sensitivity and optimization using Pro/E parameters only	sensitivity and optimization uses Pro/M variables

## Types of Models

This is fairly self-explanatory. In addition to 3D solid, shell, and beam models, Pro/M in both modes can treat 2D models (plane stress, plane strain, or axisymmetric). Note that all geometry and model entities (loads and constraints) for all 2D model types must be defined in the XY plane of a selected coordinate system. Also, a very thin plate might be modeled as a 2D shell, but if it is loaded with any force components normal to the plate, then it becomes a 3D problem.

Independent Pro/M contains a good set of tools to create both 2D and 3D geometry. Complicated 3D geometry of parts would be easier to make in Pro/E or some other CAD package, and brought into Pro/M in integrated mode. The model geometry is generally created entirely in Pro/E. It is possible to create some (non-solid) simulation features while in Pro/M, such as datum points and curves.

## Types of Elements

The various types of elements that can be used in Pro/M are listed in Table I. It is possible to use different types of elements in the same model (e.g. combining solid + beam + spring elements), but we will discuss only a couple of models of this degree of complexity in these tutorials. At first glance, this seems like a limited list of element types. H-element programs typically have large libraries of different element types, but these are often necessary to overcome the limitations of low order simple h-elements. In Pro/M, we do not have this problem and you can do practically anything with the elements available.

## Analysis Methods

For a given model, several different analysis types are possible. For example, the *static* analysis will compute the stresses and deformations within the model, while the *modal* analysis will compute the mode shapes and natural frequencies. *Buckling* analysis will compute the buckling loads on the body, and so. Other analysis methods are available but in this manual, we will only look at static stress and modal analysis.

## Convergence Methods

As discussed above, using the p-code method allows Pro/M to monitor the solution and modify the polynomial edge order until a solution has been achieved to a specified accuracy. This is implemented with three options:

- **Quick Check** - This actually isn't a convergence method since the model is run only for a single fixed (low, usually 3) polynomial order. **The results of a Quick Check should never be trusted.** What a Quick Check is for is to quickly run the model through the solver in order to pick up any errors that may have been made, for example in the constraints. A quick review of the results will also indicate whether any gross modeling errors have been made and possibly to point out potential problem areas in the model.
- **Single Pass Adaptive** - More than a Quick Check, but less than a complete convergence run, the single pass adaptive method performs one pass at a low polynomial order, assesses the accuracy of the solution, modifies the p-level of "problem elements", and does a final pass with some elements raised to an order that should provide reasonable results. Unless the model is very computationally intensive and/or is very well behaved and understood, avoid this method. The Single Pass Adaptive analysis is available for most model types.
- **Multi-Pass Adaptive** - The ultimate in convergence analysis. Multiple "p-loop" passes are made through the solver, with edge orders of "problem elements" being increased with each pass. This iterative approach continues until either the solution converges to a specified accuracy or the maximum specified edge order (default 6, maximum 9) is reached. At the conclusion of the run, the convergence measures may be examined. These are typically the Von Mises stress and the total strain energy, as shown in Figure 12. Unless you have a very good reason not to, always base your final conclusions on the results obtained using this convergence method.

## Design Studies

A Design Study is a problem or set of problems that you define for a particular model. When you ultimately press the **Run** button on Pro/M, what will execute is a design study - it is the top-most level of organization in Pro/M. There are three types of design studies:

- A **Standard** design study is the most basic and simple. It will include at least one but possibly several analyses (for example a static analysis plus a modal analysis). For this

study, you need to specify the geometry, create the elements, assign material properties, set up loads and constraints, determine the analysis and convergence types, and then display and review the final results. The Standard design study is what most people would consider “Finite Element Analysis.”

- A **sensitivity** design study can be set up so that results are computed for several different values of designated design variables or material properties. In addition to the standard model, you need to designate the design variables and the range over which you want them to vary. You can use a sensitivity study to determine, for example, which design variables will have the most effect on a particular measure of performance of the design like the maximum stress or total mass.
- Finally, the most powerful design study is an **optimization**. For this, you start with a basic FEA model. You then specify a desired goal (such as minimum mass of the body), geometric constraints (such as dimensions or locations of geometric entities), material constraints (such as maximum allowed stress) and one or more design variables which can vary over specified ranges. Pro/M will then search through the space of the design variables and determine the best design that satisfies your constraints. Amazing!

---

## A Brief Note about Units

It is crucial to use a consistent set of units throughout your Pro/M activities. The program itself has no default set of units (other than those brought in with the model from Pro/E), and only uses the numerical values provided by you. Thus, if your geometry is created with a particular linear unit like mm or inches in mind, you must make sure that any other data supplied, such as loads (force, pressure) and material properties (density, Young’s modulus, and so on) are defined consistently. The built-in material libraries offer properties for common materials in four sets of units (all at room temperature):

- inch - pound - second
- foot - pound - second
- meter - Newton - second
- millimeter - Newton - second

Note that the weight of the material is obtained by multiplying the mass density property by the acceleration of gravity expressed in the appropriate unit system.

If you require or wish to use a different system of units, you can enter your own material properties, but must look after consistency yourself. Table III outlines the common units in the various systems including how some common results will be reported by MECHANICA. For further information on units, consult the on-line help page “Unit Conversion Tables.”

TABLE III - Common unit systems in Pro/MECHANICA

Quantity	System and Units			
	SI MNS	Metric mm-N-s	English FPS ft-lb-sec	English IPS in-lb-sec
length	m	mm	ft	in
time	s	s	sec	sec
mass	kg	tonne (1000 kg)	slug	lbf-sec <sup>2</sup> / in
density	kg/m <sup>3</sup>	tonne/mm <sup>3</sup>	slug/ft <sup>3</sup>	lbf-sec <sup>2</sup> / in <sup>4</sup>
gravity, g	9.81 m/s <sup>2</sup>	9810 mm/s <sup>2</sup>	32.2 ft/sec <sup>2</sup>	386.4 in/sec <sup>2</sup>
force	N	N	lbf	lbf
stress, pressure, Young's modulus	N/m <sup>2</sup> = Pa	N/mm <sup>2</sup> = MPa	lbf/ft <sup>2</sup>	lbf/in <sup>2</sup> = psi

## Files and Directories Produced by Pro/MECHANICA

Since you will be working in integrated mode in this book, note that your entire simulation model is stored in the Pro/E part file. You do not need to store a special copy of this. Simulation entities like loads and constraints will appear when you transfer into Pro/M from Pro/E.

Pro/M produces a bewildering array of files and directories. Unless you specify otherwise (or specified in your default system configuration), all of these will be created in the Pro/E working directory. It is therefore wise to create a new subdirectory for each model, make it your working directory, and store the part file there. Locations for temporary and output files can be changed at appropriate points in the program. For example, when you set up to run a design study, you can designate the location for the subdirectory which Pro/M will create for the output files.

The important files and directories are indicated in the Table IV. In the table, the symbol ① represents the directory specified in the **Run > Settings** dialog box for output files, and ② represents the directory specified in the same dialog box for temporary files. Unless the run terminates abnormally, all temporary files are deleted on completion of a run. The names *model*, *study*, and *filename* are supplied by you during execution of the program. Note that many of these files are stored in a binary format and are not readable by normal file editors.

Table IV - Some Files Produced by Pro/MECHANICA

File Type	File/Directory Name	Comments
<b>Model Files</b>	<i>model.mdb</i> <i>model.mbk</i>	the <b>mdb</b> file contains the last-saved model database. <b>mbk</b> is a backup that can be used if the <b>mdb</b> file is lost or corrupted
<b>Engine Files</b>	①/ <i>study/study.mdb</i>	contains the entire model database at the time a design study is started
	①/ <i>study/study.cnv</i> ①/ <i>study/study.hst</i> ①/ <i>study/study.res</i> ①/ <i>study/study.rpt</i>	Engine output files: - convergence information - model updates during optimization - measures at each pass - output report for a design study (also accessible with the <b>Run &gt; Summary</b> command)
<b>Exchange Files</b>	<i>filename.dxf</i> <i>filename.igs</i>	file formats used for import/export of geometry information
<b>Temporary Files</b>	②/ <i>study.tmp/* .tmp</i> ②/ <i>study.tmp/* .bas</i>	should delete automatically on completion of design study
<b>Results Files</b>	<i>filename.rwd</i>	result window definitions stored with Save in the Result Windows dialog box
<b>AutoGEM Files</b>	<i>model.agm</i>	information about the most recent AutoGEM operation. If the model has not yet been named, this file is <b>untitled.agm</b>
<b>Miscellaneous Files</b>	<b>mechevnt</b>	a complete history of the most recent Pro/M session (every command, mouse click, and data entry). Automatically overwritten with next session.

## On-line Documentation

For further details on any of these functions or operating commands, consult the on-line documentation available with MECHANICA. See your local system administrator for

information on how to access these files.

## Summary

This chapter has introduced the background to FEA. In particular, the difference between h-code and p-code methods have been discussed. The general procedure involved in performing an analysis was described. Finally, an overview of MECHANICA has been presented to give you a view of the forest before we start looking at the individual trees!

You are strongly urged to have a look at the articles written by Dr. Paul Kurowski that are listed in the References at the end of this chapter. These offer an in-depth look at common errors made in FEA, the concept of convergence, a comparison of h- and p-elements, and more comments on the difference between CAD and FEA.

In the next Chapter, we will start to look at the basic tools within MECHANICA. We will produce a simple model and go through the process of setting up a standard design study for static analysis of a simple 3D solid model. We will also take a first look at the methods for viewing the results of the analysis.

## References

“*Avoiding Pitfalls in FEA*,” Paul Kurowski, *Machine Design*, November 1994.

“*When good engineers deliver bad FEA*,” Paul Kurowski, *Machine Design*, November, 1995.

“*Good Solid Modeling, Bad FEA*,” Paul Kurowski, *Machine Design*, November, 1996.

Finite Element Methods for Engineers, Roger T. Fenner, Macmillan, 1975.

Building Better Products with Finite Element Analysis, Vince Adams and Abraham Askenazi, Onword Press, 1998.

The Finite Element Method in Mechanical Design, Charles E. Knight, Jr., PWS-Kent, 1993.

CAD/CAM Theory and Practice, Ibrahim Zeid, McGraw-Hill, 1991.

The Finite Element Method, T.J.R. Hughes, Prentice Hall, 1987.

Computer-Assisted Mechanical Design, J.Ed Akin, Prentice Hall, 1990.

## Questions for Review

1. What is the purpose of interpolating polynomials in FEA?
2. What is a “model?” What are some different types of models and how do these relate to the real world?
3. Is it ever possible for a FEM solution to be “exact?” Why or why not?
4. What is the primary source of error when using first-order h-elements for stress analysis?
5. Give an outline of the necessary steps in performing FEA.
6. Why is it probably not a good idea to use a CAD model directly in an FEA solution?
7. What is the “Golden Rule” of FEA?
8. How is convergence of the solution obtained using h-code and p-code methods?
9. Does mesh refinement always yield higher maximum stresses?
10. What is the maximum edge order available in Pro/M? In the (unlikely) event that the solution will not converge, what needs to be done?
11. What measures are typically used in Pro/M to monitor convergence?
12. How will error enter into an FEA?
13. What is a *design study*? What types are available in Pro/M?
14. What are the three methods of convergence analysis? When would each be appropriate?
15. What types of 2D models can be created? In what operating modes? What restrictions are there on 2D models?
16. What types of analyses can be performed on a model?
17. How can you gain access to the on-line help on your system?
18. Compare the advantages and disadvantages of *integrated* and *independent* modes of operation.
19. What is the maximum edge order available in Pro/M? In the (unlikely) event that the solution will not converge, what needs to be done?
20. What measures are typically used in Pro/M to monitor convergence?
21. What are the steps required to perform a complete FEA with Pro/MECHANICA?
22. Where and how do you set up the units for the Mechanica model?

## Exercises

1. Consult a numerical methods textbook and find out what algorithms are used to solve very large linear systems. What effect does round-off error have, and can this be quantified? Are some methods more susceptible to round-off than others?
2. Locate some product brochures for FEM software, and look for the kind of modeling errors discussed in this chapter. Compare the models to the “real thing” and comment on any differences you notice.