6
FEM Modeling: Introduction

TABLE OF CONTENTS

Page

§6.1. **FEM Terminology** 6–3

§6.2. **Idealization** 6–4
- §6.2.1. Models . . . 6–5
- §6.2.2. Mathematical Models . . . 6–5
- §6.2.3. Implicit vs. Explicit Modeling . . . 6–6

§6.3. **Discretization** 6–6
- §6.3.1. Decisions . . . 6–6
- §6.3.2. Error Sources and Approximation . . . 6–7
- §6.3.3. Other Discretization Methods . . . 6–7

§6.4. **The Finite Element Method** 6–8
- §6.4.1. Interpretation . . . 6–8
- §6.4.2. Element Attributes . . . 6–8

§6.5. **Classification of Mechanical Elements** 6–10
- §6.5.1. Primitive Structural Elements . . . 6–10
- §6.5.2. Continuum Elements . . . 6–10
- §6.5.3. Special Elements . . . 6–10
- §6.5.4. Macroelements . . . 6–11
- §6.5.5. Substructures . . . 6–11

§6.6. **Assembly** 6–12

§6.7. **Boundary Conditions** 6–12
- §6.7.1. Essential and Natural B.C. . . . 6–12
- §6.7.2. Boundary Conditions in Structural Problems . . . 6–12

§6. **Notes and Bibliography** . . . 6–13

§6. **References** . . . 6–13

Chapters 2 through 5 cover material technically known as Matrix Structural Analysis or MSA. This is a subject that historically preceded the Finite Element Method (FEM), as chronicled in Appendix H. Here we begin the coverage of the FEM proper. This is technically distinguished from MSA by the more dominant role of continuum and variational mechanics.

This Chapter introduces terminology used in FEM modeling, and surveys the attributes and types of finite elements used in structural mechanics. The next Chapter gives more specific rules for defining meshes, forces and boundary conditions.

§6.1. FEM Terminology

The ubiquitous term "degrees of freedom," often abbreviated to either freedom or DOF, has figured prominently in the preceding Chapters. This term, as well as "stiffness matrix" and "force vector," originated in structural mechanics, the application for which FEM was invented. These names have carried over to non-structural applications. This "terminology overspill" is discussed next.

Classical analytical mechanics is that invented by Euler and Lagrange in the XVIII century and further developed by Hamilton and Jacobi as a systematic formulation of Newtonian mechanics. Its objects of attention are models of mechanical systems ranging from material particles composed of sufficiently large number of molecules, through airplanes, to the Solar System.[1] The spatial configuration of any such system is described by its *degrees of freedom*. These are also called *generalized coordinates*. The terms *state variables* and *primary variables* are also used, particularly in mathematically oriented treatments.

If the number of degrees of freedom is finite, the model is called *discrete*, and *continuous* otherwise. Because FEM is a discretization method, the number of degrees of freedom of a FEM model is necessarily finite. The freedoms are collected in a column vector called $\mathbf{u}$. This vector is called the *DOF vector* or *state vector*. The term *nodal displacement vector* for $\mathbf{u}$ is reserved to mechanical applications.

In analytical mechanics, each degree of freedom has a corresponding "conjugate" or "dual" term, which represents a generalized force.[2] In non-mechanical applications, there is a similar set of conjugate quantities, which for want of a better term are also called *forces* or *forcing terms*. They are the agents of change. These forces are collected in a column vector called $\mathbf{f}$. The inner product $\mathbf{f}^T\mathbf{u}$ has the meaning of external energy or work.

Just as in the truss problem, the relation between $\mathbf{u}$ and $\mathbf{f}$ is assumed to be of linear and homogeneous. The last assumption means that if $\mathbf{u}$ vanishes so does $\mathbf{f}$. The relation is then expressed by the master stiffness equations:

$$\boxed{\mathbf{K}\mathbf{u} = \mathbf{f}.} \tag{6.1}$$

$\mathbf{K}$ is universally called the *stiffness matrix* even in non-structural applications because no consensus has emerged on different names.

The physical significance of the vectors $\mathbf{u}$ and $\mathbf{f}$ varies according to the application being modeled, as illustrated in Table 6.1.

---

[1] For cosmological scales, such as galaxy clusters, the general theory of relativity is necessary. For the atomic and sub-particle world, quantum mechanics is appropriate.

[2] In variational mathematics this is called a duality pairing.

Table 6.1. Significance of **u** and **f** in Miscellaneous FEM Applications

Application Problem	*State (DOF) vector* **u** *represents*	*Conjugate vector* **f** *represents*
Structures and solid mechanics	Displacement	Mechanical force
Heat conduction	Temperature	Heat flux
Acoustic fluid	Displacement potential	Particle velocity
Potential flows	Pressure	Particle velocity
General flows	Velocity	Fluxes
Electrostatics	Electric potential	Charge density
Magnetostatics	Magnetic potential	Magnetic intensity

If the relation between forces and displacements is linear but not homogeneous, equation (6.1) generalizes to

$$\boxed{\mathbf{K}\mathbf{u} = \mathbf{f}_M + \mathbf{f}_I.} \tag{6.2}$$

Here $\mathbf{f}_I$ is the initial node force vector introduced in Chapter 29 for effects such as temperature changes, and $\mathbf{f}_M$ is the vector of mechanical forces.

The basic steps of FEM are discussed below in more generality. Although attention is focused on structural problems, most of the steps translate to other applications problems as noted above. The role of FEM in numerical simulation is schematized in Figure 6.1, which is a merged simplification of Figures 1.2 and 1.3. Although this diagram oversimplifies the way FEM is actually used, it serves to illustrate terminology. The three key simulation steps shown are: *idealization*, *discretization* and *solution*. Each step is a source of errors. For example, the discretization error is the discrepancy that appears when the discrete solution is substituted in the mathematical model. The reverse steps: continuification and realization, are far more difficult and ill-posed problems.

The idealization and discretization steps, briefly mentioned in Chapter 1, deserve further discussion. The solution step is dealt with in more detail in Part III of this book.

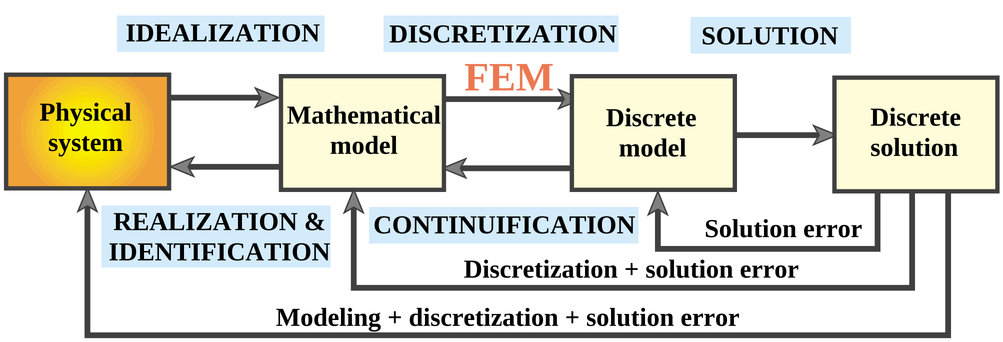

FIGURE 6.1. A simplified view of the physical simulation process, reproduced to illustrate modeling terminology.

§6.2. Idealization

Idealization passes from the physical system to a mathematical model. This is the most important step in engineering practice, because it cannot be "canned." It must be done by a human.

§6.2.1. Models

The word "model" has the traditional meaning of a scaled copy or representation of an object. And that is precisely how most dictionaries define it. We use here the term in a more modern sense, which has become increasingly common since the advent of computers:

$$\boxed{\text{A model is a symbolic device built to simulate and predict aspects of behavior of a system.}} \tag{6.3}$$

Note the distinction made between *behavior* and *aspects of behavior*. To predict everything, in all physical scales, you must deal with the actual system. A model *abstracts* aspects of interest to the modeler. The qualifier *symbolic* means that a model represents a system in terms of the symbols and language of another discipline. For example, engineering systems may be (and are) modeled with the symbols of mathematics and/or computer sciences.[3]

§6.2.2. Mathematical Models

Mathematical modeling, or *idealization*, is a process by which an engineer or scientist passes from the actual physical system under study, to a *mathematical model* of the system, where the term *model* is understood in the sense of (6.3).

The process is called *idealization* because the mathematical model is necessarily an abstraction of the physical reality — note the phrase *aspects of behavior* in (6.3). The analytical or numerical results produced by the mathematical model are physically re-interpreted only for those aspects.[4]

To give an example on the choices an engineer may face, suppose that the structure is a flat plate structure subjected to transverse loading. Here is a non-exhaustive list of four possible mathematical models:

1. A *very thin* plate model based on Von Karman's coupled membrane-bending theory.
2. A *thin* plate model, such as the classical Kirchhoff's plate theory.
3. A *moderately thick* plate model, for example Mindlin-Reissner plate theory.
4. A *very thick* plate model based on three-dimensional elasticity.

The person responsible for this kind of decision is supposed to be familiar with the advantages, disadvantages, and range of applicability of each model. Furthermore the decision may be different in static analysis than in dynamics.

Why is the mathematical model an abstraction of reality? Engineering systems, particularly in Aerospace and Mechanical, tend to be highly complex. For simulation it is necessary to reduce that

---

[3] A problem-definition input file, a digitized earthquake record, or a stress plot are examples of the latter.

[4] Whereas idealization can be reasonably taught in advanced design courses, the converse process of "realization" or "identification" — see Figure 6.1 — generally requires considerable physical understanding and maturity that can only be gained through professional experience.

complexity to manageable proportions. Mathematical modeling is an abstraction tool by which complexity can be controlled.

This is achieved by "filtering out" physical details that are not relevant to the analysis process. For example, a continuum material model filters out the aggregate, crystal, molecular and atomic levels of matter. Engineers are typically interested in a few integrated quantities, such as the maximum deflection of a bridge or the fundamental periods of an airplane. Although to a physicist this is the result of the interaction of billions and billions of molecules, such details are weeded out by the modeling process. Consequently, picking a mathematical model is equivalent to choosing an information filter.

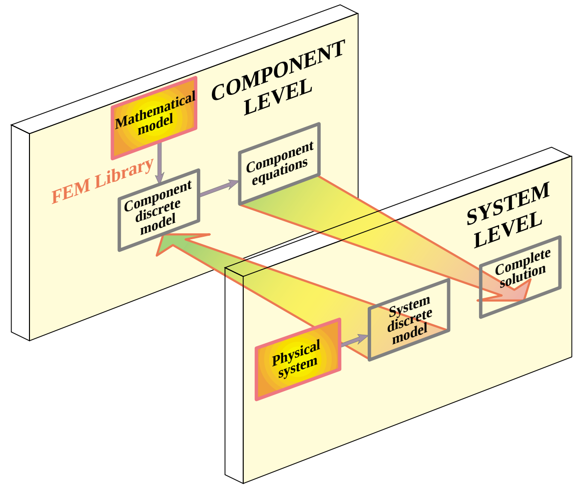

FIGURE 6.2. A reproduction of Figure 1.5 with some relabeling. Illustrates implicit modeling: picking elements from an existing FEM code consents to an idealization.

§6.2.3. Implicit vs. Explicit Modeling

As noted the diagram of Figure 6.1 is an oversimplification of engineering practice. The more common scenario is that pictured in Figures 1.2, 1.4 and 1.5. The latter is reproduced in Figure 6.2 for convenience.

A common scenario in industry is: you have to analyze a structure or a substructure, and at your disposal is a "black box" general-purpose finite element program. Those programs offer a *catalog* of element types; for example, bars, beams, plates, shells, axisymmetric solids, general 3D solids, and so on. The moment you choose specific elements from the catalog you automatically accept the mathematical models on which the elements are based. This is *implicit modeling*. Ideally you should be fully aware of the implications of your choice. Providing such "finite element literacy" is one of the objective of this book. Unfortunately many users of commercial programs are unaware of the implied-consent aspect of implicit modeling and their legal implications.

The other extreme happens when you select a mathematical model of the physical problem with your eyes wide open and *then* either shop around for a finite element program that implements that model, or write the program yourself. This is *explicit modeling*. It requires far more technical expertise, resources, experience and maturity than implicit modeling. But for problems that fall out of the ordinary it could be the right thing to do.

In practice a combination of implicit and explicit modeling is common. The physical problem to be simulated is broken down into subproblems. Those subproblems that are conventional and fit available programs may be treated with implicit modeling, whereas those that require special handling may only submit to explicit modeling.

§6.3. Discretization

Mathematical modeling is a simplifying step. But models of physical systems are not necessarily simple to solve. They often involve coupled partial differential equations in space and time subject to boundary and/or interface conditions. Such models have an *infinite* number of degrees of freedom.

§6.3.1. Decisions

At this point one faces the choice of going for analytical or numerical solutions. Analytical solutions, also called "closed form solutions," are more intellectually satisfying, particularly if they apply to a wide class of problems, so that particular instances may be obtained by substituting the values of free parameters. Unfortunately they tend to be restricted to regular geometries and simple boundary conditions. Moreover some closed-form solutions, expressed for example as inverses of integral transforms, often have to be numerically evaluated to be useful.

Most problems faced by the engineer either do not yield to analytical treatment or doing so would require a disproportionate amount of effort.[5] The practical way out is numerical simulation. Here is where finite element methods enter the scene.

To make numerical simulations practical it is necessary to reduce the number of degrees of freedom to a *finite* number. The reduction is called *discretization*. The product of the discretization process is the *discrete model*. As discussed in Chapter 1, for complex engineering systems this model is the product of a multilevel decomposition.

Discretization can proceed in space dimensions as well as in the time dimension. Because the present book deals only with static problems, we need not consider the time dimension and are free to concentrate on *spatial discretization*.

§6.3.2. Error Sources and Approximation

Figure 6.1 tries to convey graphically that each simulation step introduces a source of error. In engineering practice modeling errors are by far the most important. But they are difficult and expensive to evaluate, because *model validation* requires access to and comparison with experimental results. These may be either scarce, or unavailable in the case of a new product in the design stage.

Next in order of importance is the *discretization error*. Even if solution errors are ignored — and usually they can — the computed solution of the discrete model is in general only an approximation in some sense to the exact solution of the mathematical model. A quantitative measurement of this discrepancy is called the *discretization error*. The characterization and study of this error is addressed by a branch of numerical mathematics called approximation theory.

Intuitively one might suspect that the accuracy of the discrete model solution would improve as the number of degrees of freedom is increased, and that the discretization error goes to zero as that number goes to infinity. This loosely worded statement describes the *convergence* requirement of discrete approximations. One of the key goals of approximation theory is to make the statement as precise as it can be expected from a branch of mathematics.[6]

---

[5] This statement has to be tempered in two respects. First, the wider availability and growing power of computer algebra systems, outlined in Chapter 4, has widened the realm of analytical solutions than can be obtained within a practical time frame. Second, a combination of analytical and numerical techniques is often effective to reduce the dimensionality of the problem and to facilitate parameter studies. Important examples are provided by Fourier analysis, perturbation and boundary-element methods.

[6] The discretization error is often overhyped in the FEM literature, since it provides an inexhaustible source of publishable poppycock. If the mathematical model is way off reducing the discretization error buys nothing; only a more accurate answer to the wrong problem.

§6.3.3. Other Discretization Methods

It was stated in Chapter 1 that the most popular discretization techniques in structural mechanics are finite element methods and boundary element methods. The finite element method (FEM) is by far the most widely used. The boundary element method (BEM) has gained in popularity for special types of problems, particularly those involving infinite domains, but remains a distant second, and seems to have reached its natural limits.

In non-structural application areas such as fluid mechanics and electromagnetics, the finite element method is gradually making up ground but faces stiff competition from both the classical and energy-based *finite difference* methods. Finite difference and finite volume methods are particularly well entrenched in computational fluid dynamics spanning moderate to high Reynolds numbers.

§6.4. The Finite Element Method

§6.4.1. Interpretation

The finite element method (FEM) is the dominant discretization technique in structural mechanics. As discussed in Chapter 1, the FEM can be interpreted from either a physical or mathematical standpoint. The treatment has so far emphasized the former.

The basic concept in the physical FEM is the subdivision of the mathematical model into disjoint (non-overlapping) components of simple geometry called *finite elements* or *elements* for short. The response of each element is expressed in terms of a finite number of degrees of freedom characterized as the value of an unknown function, or functions, at a set of nodal points. The response of the mathematical model is then considered to be approximated by that of the discrete model obtained by connecting or assembling the collection of all elements.

The disconnection-assembly concept occurs naturally when examining many artificial and natural systems. For example, it is easy to visualize an engine, bridge, building, airplane, or skeleton as fabricated from simpler components.

Unlike finite difference models, finite elements *do not overlap* in space. In the mathematical interpretation of the FEM, this property goes by the name *disjoint support* or *local support*.

§6.4.2. Element Attributes

Just like members in the truss example, one can take finite elements of any kind one at a time. Their local properties can be developed by considering them in isolation, as individual entities. This is the key to the modular programming of element libraries.

In the Direct Stiffness Method, elements are isolated by the disconnection and localization steps, which were described for the truss example in Chapter 2. The procedure involves the separation of elements from their neighbors by disconnecting the nodes, followed by referral of the element to a convenient local coordinate system.[7] After that we can consider *generic* elements: a bar element, a beam element, and so on. From the standpoint of the computer implementation, it means that you can write one subroutine or module that constructs, by suitable parametrization, all elements of one type, instead of writing one module for each element instance.

[7] Both steps are only carried out in the modeler's mind. They are placed as part of the DSM for instructional convenience. In practice the analysis begins directly at the element level.

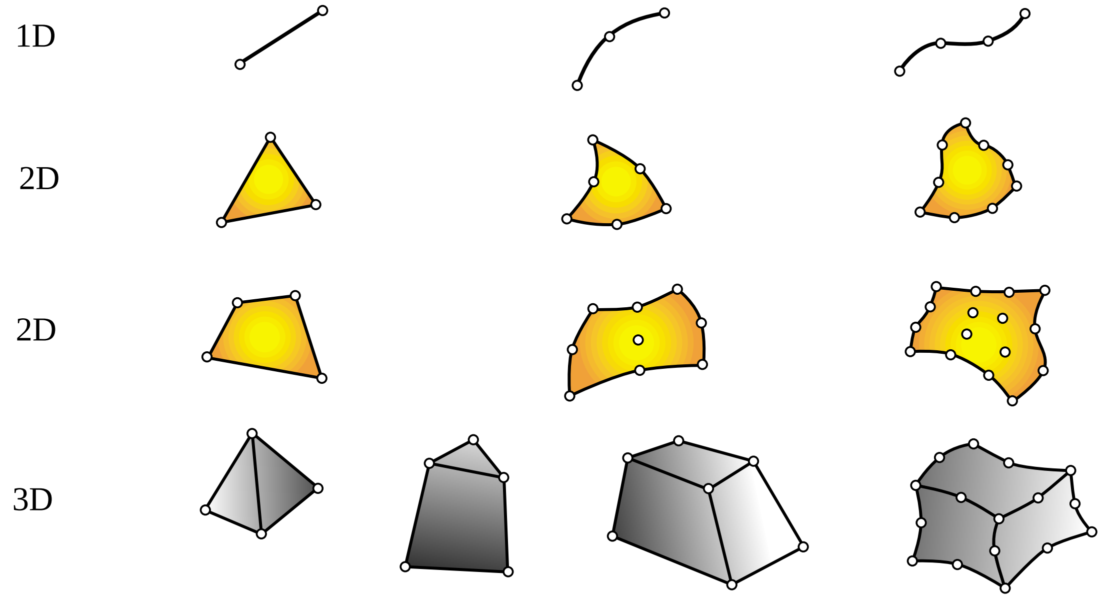

FIGURE 6.3. Typical finite element geometries in one through three dimensions.

Following is a summary of the data associated with an individual finite element. This data is used in finite element programs to carry out element level calculations.

Intrinsic Dimensionality. Elements can have one, two or three space dimensions.[8] There are also special elements with zero dimensionality, such as lumped springs or point masses.

Nodal points. Each element possesses a set of distinguishing points called *nodal points* or *nodes* for short. Nodes serve a dual purpos: definition of element geometry, and home for degrees of freedom. They are usually located at the corners or end points of elements, as illustrated in Figure 6.3. In the so-called refined or higher-order elements nodes are also placed on sides or faces, as well as perhaps the interior of the element.

Geometry. The geometry of the element is defined by the placement of the nodal points. Most elements used in practice have fairly simple geometries. In one-dimension, elements are usually straight lines or curved segments. In two dimensions they are of triangular or quadrilateral shape. In three dimensions the most common shapes are tetrahedra, pentahedra (also called wedges or prisms), and hexahedra (also called cuboids or "bricks"). See Figure 6.3.

Degrees of freedom. The degrees of freedom (DOF) specify the *state* of the element. They also function as "handles" through which adjacent elements are connected. DOFs are defined as the values (and possibly derivatives) of a primary field variable at nodal points. The actual selection depends on criteria studied at length in Part II. Here we simply note that the key factor is the way in which the primary variable appears in the mathematical model. For mechanical elements, the primary variable is the displacement field and the DOF for many (but not all) elements are the displacement components at the nodes.

Nodal forces. There is always a set of nodal forces in a one-to-one correspondence with degrees of

[8] In dynamic analysis, time appears as an additional dimension.

freedom. In mechanical elements the correspondence is established through energy arguments.

Constitutive properties. For a mechanical element these are relations that specify the material behavior. For example, in a linear elastic bar element it is sufficient to specify the elastic modulus E and the thermal coefficient of expansion α.

Fabrication properties. For mechanical elements these are fabrication properties which have been integrated out from the element dimensionality. Examples are cross sectional properties of MoM elements such as bars, beams and shafts, as well as the thickness of a plate or shell element.

This data is used by the element generation subroutines to compute element stiffness relations in the local system.

§6.5. Classification of Mechanical Elements

The following classification of finite elements in structural mechanics is loosely based on the "closeness" of the element with respect to the original physical structure.

It is given here because it clarifies points that recur in subsequent sections, as well as providing insight into advanced modeling techniques such as hierarchical breakdown and global-local analysis.

§6.5.1. Primitive Structural Elements

These resemble fabricated structural components, and are often drawn as such; see Figure 6.4. The qualifier *primitive* is used to distinguish them from macroelements, which is another element class described below. Primitive means that they are not decomposable into simpler elements.

These elements are usually derived from Mechanics-of-Materials simplified theories and are better understood from a physical, rather than mathematical, standpoint. Examples are the elements discussed in Chapter 5: bars, cables, beams, shafts, spars.

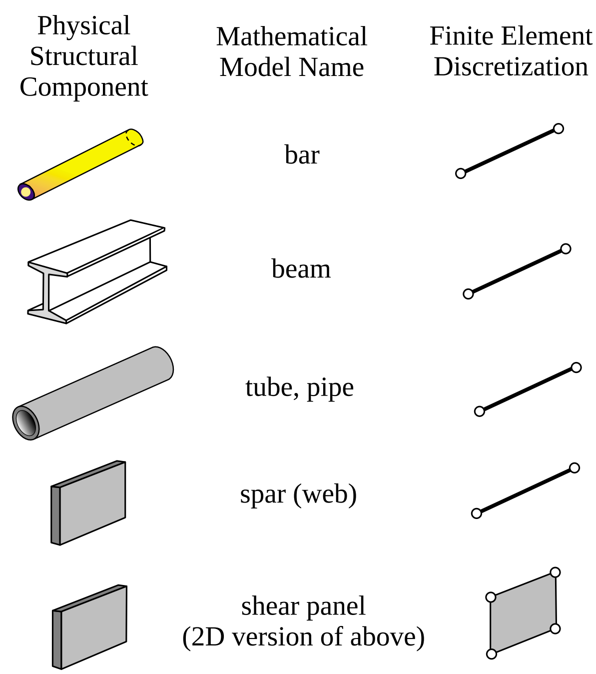

FIGURE 6.4. Examples of primitive structural elements.

§6.5.2. Continuum Elements

These do not resemble fabricated structural components at all. They result from the subdivision of "blobs" of continua, or of structural components viewed as continua.

Unlike structural elements, continuum elements are better understood in terms of their mathematical interpretation. Examples: plates, slices, shells, axisymmetric solids, general solids. See Figure 6.5.

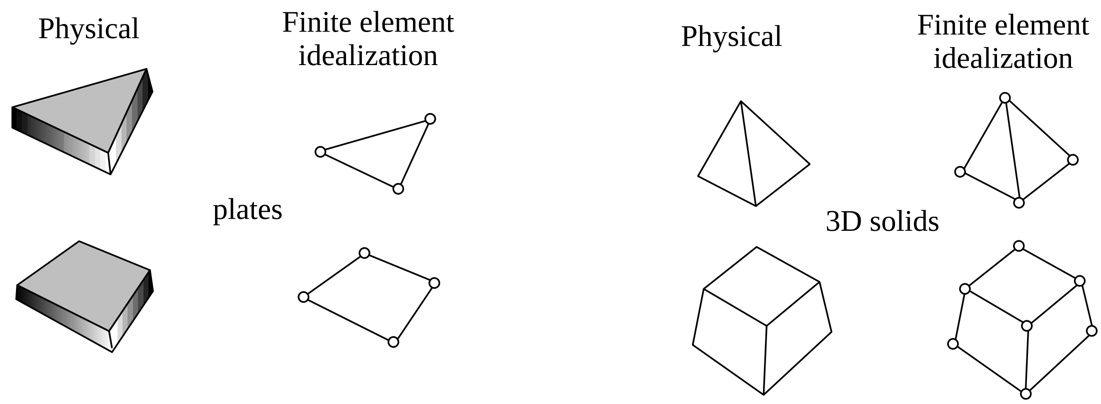

FIGURE 6.5. Continuum element examples.

§6.5.3. Special Elements

Special elements partake of the characteristics of structural and continuum elements. They are derived from a continuum mechanics standpoint but include features closely related to the physics of the problem. Examples: crack elements for fracture mechanics applications, shear panels, infinite and semi-infinite elements, contact and penalty elements, rigid-body elements. See Figure 6.6.

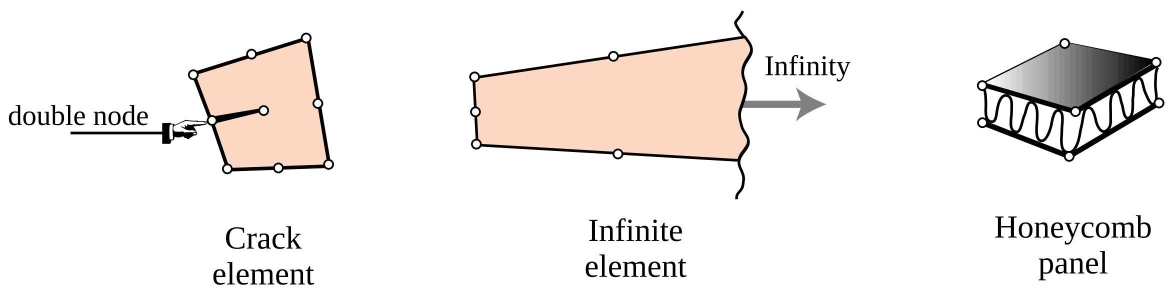

FIGURE 6.6. Special element examples.

§6.5.4. Macroelements

Macroelements are also called mesh units and superelements, although the latter term overlaps with substructures (defined below). These often resemble structural components, but are fabricated with simpler elements. See Figure 6.7.

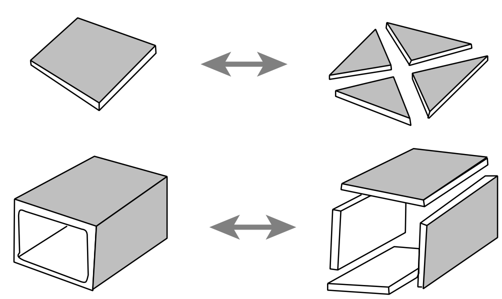

FIGURE 6.7. Macroelement examples.

The main reason for introducing macroelements is to simplify preprocessing tasks. For example, it may be simpler to define a regular 2D mesh using quadrilaterals rather than triangles. The fact that the quadrilateral is really a macroelement may not be important for the majority of users.

Similarly a box macroelement can save modeling times for structures that are built by such components; for example box-girder bridges.

§6.5.5. Substructures

Also called structural modules and superelements. These are sets of elements with a well defined structural function, typically obtained by cutting the complete structure into functional components. Examples: the wings and fuselage in an airplane, the deck and cables in a suspension bridge.

The distinction between substructures and macroelements is not clear-cut. The main conceptual distinction is that substructures are defined "top down" as parts of a complete structure, whereas macroelements are built "bottom up" from primitive elements. The term *superelement* is often used in a collective sense to embrace all element groupings. This topic is further covered in Chapter 10.

§6.6. Assembly

The assembly procedure of the Direct Stiffness Method for a general finite element model follows rules identical in principle to those discussed for the truss example. As in that case the processs involves two basic steps:

Globalization. The element equations are transformed to a common *global* coordinate system.

Merge. The element stiffness equations are merged into the master stiffness equations by appropriate indexing and matrix-entry addition.

The hand calculations for the example truss conceal the implementation complexity. The master stiffness equations in practical cases may involve thousands (or even millions) of freedoms. The use of sparse matrix techniques, and possibly peripheral storage, becomes essential. But this inevitably increases the programming complexity. The topic is elaborated upon in Part III.

§6.7. Boundary Conditions

A key strength of the FEM is the ease and elegance with which it handles arbitrary boundary and interface conditions. This power, however, has a down side. A big hurdle faced by FEM newcomers is the understanding and proper handling of boundary conditions. Below is a simple recipe for treating boundary conditions. The following Chapter provides specific rules and examples.

§6.7.1. Essential and Natural B.C.

The key thing to remember is that boundary conditions (BCs) come in two basic flavors: essential and natural.

Essential BCs directly affect DOFs, and are imposed on the left-hand side vector $\mathbf{u}$.

Natural BCs do not directly affect DOFs and are imposed on the right-hand side vector $\mathbf{f}$.

The mathematical justification for this distinction requires use of concepts from variational calculus, and is consequently relegated to Part II. For the moment, the basic recipe is:

1. If a boundary condition involves one or more degrees of freedom in a *direct* way, it is essential. An example is a prescribed node displacement.
2. Otherwise it is natural.

The term "direct" is meant to exclude derivatives of the primary function, unless those derivatives also appear as degrees of freedom, such as rotations in beams and plates.

§6.7.2. Boundary Conditions in Structural Problems

Essential boundary conditions in mechanical problems involve *displacements* (but not strain-type displacement derivatives). Support conditions for a building or bridge problem furnish a particularly simple example. But there are more general boundary conditions that occur in practice. A structural engineer must be familiar with displacement B.C. of the following types.

Ground or support constraints. Directly restraint the structure against rigid body motions.

Symmetry conditions. To impose symmetry or antisymmetry restraints at certain points, lines or planes of structural symmetry. This allows the discretization to proceed only over part of the structure with a consequent savings in modeling effort and number of equations to be solved.

Ignorable freedoms. To suppress displacements that are irrelevant to the problem.[9] Even experienced users of finite element programs are sometimes baffled by this kind. An example are rotational degrees of freedom normal to smooth shell surfaces.

Connection constraints. To provide connectivity to adjoining structures or substructures, or to specify relations between degrees of freedom. Many conditions of this type can be subsumed under the label *multipoint constraints* or *multifreedom constraints*. These can be notoriously difficult to handle from a numerical standpoint, and are covered in Chapters 8–9.

Notes and Bibliography

Most FEM textbooks do not provide a systematic treatment of modeling. This is no accident: few academic authors have experience with complex engineering systems. Good engineers are too busy (and in demand) to have time for writing books. This gap has been particularly acute since FEM came on the scene because of generational gaps: "real engineers" tend to mistrust the computer, and often for good reason. The notion of explicit versus implicit modeling, which has deep legal and professional implications, is rarely mentioned.

FEM terminology is by now standard, and so is a majority of the notation. But that is not so in early publications. E.g. **K** is universally used[10] for stiffness matrix in virtually all post-1960 books. There are a few exceptions: Przemieniecki [140] uses **S**. There is less unanimity on **u** and **f** for node displacement and force vectors, respectively; some books such as Zienkiewicz and Taylor [195] still use different symbols.

The element classification given here attempts to systematize dispersed references. In particular, the distinction between macroelements, substructures and superelements is an ongoing source of confusion, particularly since massively parallel computation popularized the notion of "domain decomposition" in the computer science community. The all-encompassing term "superelement" emerged in Norway by 1968 as part of the implementation of the computer program SESAM; additional historical details are provided in Chapter 10.

The topic of BC classification and handling is a crucial one in practice. More modeling mistakes are done in this aspect of FEM application than anywhere else.

References

Referenced items have been moved to Appendix R.

---

[9] In classical dynamics these are called *ignorable coordinates*.

[10] A symbol derived from the "spring constant" k that measures the stiffness of a mechanical spring.