

# 5

---

## Computer-Aided Engineering Analysis and Prototyping

Engineering design starts with identifying customer requirements and developing the most promising conceptual product architecture to satisfy the need at hand (Chap. 2). This stage is often followed with a finer decision making process on issues such as product modularity as well as initial parametric design of the product, including its subassemblies and parts (Chaps. 3 and 4). The concluding phase of design is engineering analysis and prototyping facilitated through the use of computing software tools. Engineering students spend the majority of their time during their undergraduate education in preparation for carrying engineering analysis tasks for this phase of design, for example, ranging from mechanical stress analysis to heat transfer and fluid flow analyses in the mechanical engineering field. Students are taught many analytical tools for solving closed-form engineering analysis problems as well as numerical techniques for solving problems that lack closed-form solution models. They are, however, often reminded that the analysis of most engineering products requires approximate solutions and furthermore frequently need physical prototyping and testing under real operating conditions owing to our inability to model analytically all physical phenomena.

The objective of engineering analysis and prototyping can therefore be noted as the optimization of the design at hand. The objective function of the optimization problem would be maximizing performance and/or minimizing

cost. The constraints would be those set by the customer and translated into engineering specifications and/or by the manufacturing processes to be employed. These would, normally, be set as inequalities, such as a minimum life expectancy or a maximum acceptable mechanical stress. The variables of the optimization problem are the geometric parameters of the product (dimensions, tolerances, etc.) as well as material properties. As discussed in [Chap. 3](#), a careful design-of-experiments process must be followed, regardless whether the analysis and prototyping process is to be carried out via numerical simulation or physical testing, in order to determine a minimal set of optimization variables. The last step in setting the analysis stage of design is selection of an algorithmic search technique that would logically vary the values of the variables in search of their optimal values. The search technique to be chosen would be either of a combinatoric nature for discrete variables or one that deals with continuous variables.

In this chapter, we will review the most common engineering analysis tool used in the mechanical engineering field, finite-element modeling and analysis, and we will subsequently discuss several optimization techniques. However, as a preamble to both topics, we will first discuss below *prototyping* in general and clarify the terminology commonly used in the mechanical engineering literature in regard to this topic.

## 5.1 PROTOTYPING

A prototype of a product is expected to exhibit the identical (or very close to) properties of the product when tested (operated) under identical physical conditions. Prototypes can, however, be required to exhibit identical behavior only for a limited set of product features according to the analysis objectives at hand. For example, analysis of airflow around an airplane wing requires only an approximate shell structure of the wing. Thus one can define the prototyping process as a time-phased process in which the need for prototyping can range from “see and feel” at the conceptual design stage to physical testing of all components at the last alpha (or even beta) stage of fabrication prior to the final production and unrestricted sale of the product.

### 5.1.1 Virtual Prototyping

Virtual (analytical) prototyping refers to the computer-aided engineering (CAE) analysis and optimization of a product carried out completely within a computer (i.e., in virtual space). This process would naturally rely on the existence of suitable software that can help the designer to model the part (via solid modeling, [Chap. 4](#)) as well as to simulate a variety of physical phenomena that the part will be subjected to (commonly, via finite-element

analysis, Sec. 5.2 below). In the past two decades, significant progress has been reported in the area of numerical modeling and simulation of physical phenomena, which however require extensive computing resources: computational fluid dynamics (CFD) is one of the fields that rely on such modeling and simulation tools.

The two primary advantages of virtual prototyping are significant engineering cost savings (as well reduced time to market) and ability to carry out distributed design. The latter advantage refers to a company's ability to carry out design in multiple locations, where design data is shared over the company's (and their suppliers') intranets. The design of the Boeing 777 airplane, in virtual space, has been the most visible and talked about virtual prototyping process.

## Boeing 777

The Boeing company is the world's largest manufacturer of commercial jetliners and military aircraft. Total company revenues for 1999 were \$58 billion. Boeing has employees in more than 60 countries and together with its subsidiaries they employ more than 189,000 people. Boeing's main commercial product line includes the 717, 737, 747, 757, 767, and 777 families of jetliners, of which there exist more than 11,000 planes in service worldwide. The Boeing fighter/attack aircraft products and programs include the F/A-18E/F Super Hornet, F/A-18 Hornet, F-15 Eagle, F-22 Raptor, and AV-8B Harrier. Other military airplanes include the C-17 Globemaster III, T-45 Goshawk, and 767 AWACS.

The Boeing 777 jetliner has been recognized as the first airplane to be 100% digitally designed and preassembled in a computer. Its virtual design eliminated the need for a costly three-stage full-scale mock-up development process that normally spans from the use of plywood and foam to hand-made full-scale airplane structures of almost identical materials to the proposed final product.

The 777 program, during the period of 1989 to 1995, established and utilized 238 design/build teams (each having 10 to 20 people) to develop each element of the plane's frame (main body and wings), which includes more than 100,000 unique parts (excluding the engines). The engines have almost 50,000 parts each and are manufactured by GE, Rolls-Royce, or Pratt and Whitney and installed on the 777 according to specific customer demand.

Under this revolutionary product design team approach, Boeing designers and manufacturing and tooling engineers, working concurrently with Boeing's suppliers and customers, created all the airplane's parts and systems. Several thousands of workstations around the world were linked to

eight IBM mainframe computers. The CATIA (computer-aided three-dimensional interactive application) and ELFINI (finite element analysis system), both developed by Dassault Systems of France, and EPIC (electronic preassembly integration on CATIA) were used for geometric modeling and computer-aided engineering analysis.

As a side note, it is worth mentioning that the 777's flight deck and the passenger cabin received the Industrial Designers Society of America Design Excellence Award. This was the first time any airplane was recognized by the society.

### 5.1.2 Virtual Reality for Virtual Prototyping

Virtual reality (VR) could be used as part of the virtual-prototyping process, in order to evaluate human-machine interfaces, for example, ease of operability of a device. The primary challenge in employing VR is to provide the user with a realistic visual sensation of the environment, normally achieved via head-mounted displays capable of generating stereoscopic images. The secondary challenge is to manipulate the environment through input devices, such as three-dimensional mice (also known as spaceballs) and intelligent gloves for simulating a one-way haptic interface (Fig. 1). However, no VR system can be fully useful if it cannot provide the user of the “virtual product” with haptic feedback—for example, a user must feel the effort required in opening a car door or lifting and placing luggage into a car's trunk.

The beginning of VR can be traced to I. Sutherland's work in the late 1960s on head-mounted display (Sutherland is also the designer and developer of the first known CAD system, Sketchpad, discussed in [Chap. 4](#)). However, VR significantly developed only more than a decade later with the introduction of high-definition graphic display hardware and surface-modeling software, as well as a variety of commercial interface devices (especially those developed for the entertainment industry) and flight-

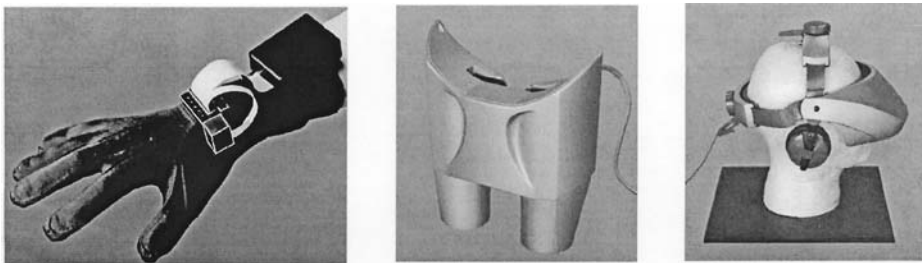


FIGURE 1 VR input/output devices. (Images courtesy of [www.5DT.com](http://www.5DT.com).)

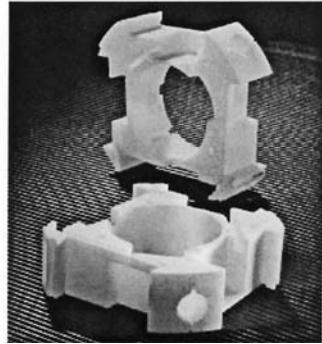
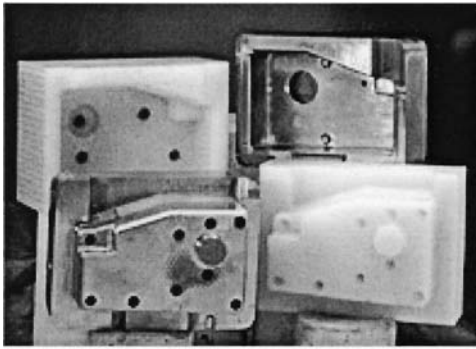
simulation applications. Naturally, not all CAD software packages provide easy interface to VR environments: CATIA with its SIMPLIFY module is one the few that not only can simplify geometric models for real-time manipulation but also can increase the quality of surface representations. VR users need to develop (nontrivial) interface programs for accessing CAD data stored by most other commercial packages, such as ADAMS/Car by Volvo, Renault, BMW, and Audi.

The automotive industry is the most common user of virtual reality in the design of commercial vehicles. Companies such as Chrysler, Ford, and Volkswagen utilize the CAD models of their vehicles to provide engineers with an immersive VR environment, for example, means of visualizing different dashboard configurations for visibility and reachability. Some have also experimented with VR to evaluate assembly (of door locks, window regulators, etc.) as well as disassembly (of tail lights, etc.) for maintainability. However, in almost all cases, users have been provided with only visual feedback and no force feedback. In numerous instances, integrated sensors have helped these users in detecting their head and hand movements and adjust the display of the virtual environment accordingly. It has been claimed that these users could evaluate the goodness of assembly plans, the suitability of tolerances, and the potential collisions with the environment.

### **5.1.3 Physical Prototyping**

Despite intensive CAE and VR efforts and successes, as noted above, problems do arise both in the exact modeling of a product and in its (virtual) analysis process. It is thus common, and in most cases mandatory owing to governmental regulations, to manufacture physical product prototypes and test them under over-stressed or accelerated conditions (to mimic long-term usage or unusual circumstances). Such physical prototyping, however, should be restricted to the functional testing of the final optimized product or the fine-tuning of design parameters. It would be costly to use physical prototypes during the parameter-optimization phase, especially if tests require the destruction of the product under duress.

In response to lengthy physical-prototyping processes, since the late 1980s, numerous technologies have been developed and commercialized for “rapid prototyping” (RP). The common objective of these techniques has been the fabrication of physical prototypes, directly from their geometric solid models, in a time-optimal manner i.e., faster than existing conventional manufacturing techniques (Fig. 2). In most cases, however, prototypes fabricated using these material-additive and layered techniques can only exhibit a very limited number of a product’s features, primarily because of



**FIGURE 2** Layered manufactured parts.

material restrictions. A very successful use of RP technologies had been the generation of part models for the fabrication of sand-casting and investment-casting dies. Current research on RP concentrates on the development of new fabrication techniques that would yield functional prototypes with increased numbers of physical characteristics identical with (or very similar to) those of the real product itself. (Several RP technologies will be detailed in [Chap. 9](#).)

## **5.2 FINITE-ELEMENT MODELING AND ANALYSIS**

The finite-element method provides engineers with an approximate behavior of a physical phenomenon in the absence of a closed-form analytical model. The quality of the approximation can be substantially increased by spending high levels of computational effort (CPU time and memory). In

this method, a continuum or an object geometry is represented as a collection of (finite) elements that are connected to each other at nodal points (nodes). Variations within each element are approximated by simple functions to analyze variables, such as displacement, temperature, velocity. Once the individual variable values are determined for all the nodes, they are assembled by the approximating functions throughout the field of interest.

Although approximate mathematical solutions to complex problems have been utilized for a long time (several centuries), the finite-element method (as it is known today) dates only back several decades—it can be traced to the earlier works of R. Courant in the 1940s and the later works of other aerospace scientists in the early 1950s. The first attempts at using the finite-element method were for the analysis of aircraft structures. In the past several decades, however, the method has been used in numerous engineering disciplines to solve many complex problems:

*Mechanical engineering:* Stress analysis of components (including composite materials); fracture and crack propagation; vibration analysis (including natural frequency and stability of components and linkages); steady-state and transient heat flow and temperature distributions in solids and liquids; and steady-state and transient fluid flow and velocity and pressure distributions in Newtonian and non-Newtonian (viscous) fluids.

*Aerospace engineering:* Stress analysis of aircraft and space vehicles (including wings, fuselage, and fins); vibration analysis; and aerodynamic (flow) analysis.

*Electrical engineering:* Electromagnetic (field) analysis of currents in electrical and electromechanical systems.

*Biomedical engineering:* Stress analysis of replacement bones, hips and teeth; fluid-flow analysis in blood vessels; and impact analysis on skull and other bones.

The finite-element modeling and analysis for the above-mentioned and other problems is a sequential procedure comprising the following primary steps:

1. *Discretization of the problem:* The object geometry or the field of interest is subdivided into a finite number of elements—the number, type, and size of the elements are closely related to the required level of approximation and should take into account existing symmetries and loading and boundary conditions.
2. *Selection of the approximating (interpolation) function:* The distribution of the unknown variable through each element is

approximated using an interpolation function—normally chosen in a polynomial form. The accuracy of the analysis can be improved by choosing higher-order (polynomial) representations, though at the expense of computational effort.

3. *Derivation of the basic element equations:* Based on the physical phenomenon examined (e.g., stress analysis), the equations that describe the behavior of the elements are derived (e.g., stiffness matrices and load vectors).
4. *Calculation of the system equations:* Individual element equations are assembled into an overall system model, and the boundary conditions are incorporated into this model.
5. *Solution of the system equations:* The system model is solved for the variable values at individual nodes (e.g., displacement).

In most cases, it is expected that an object model considered for finite-element analysis (FEA) would be developed in a CAD environment and imported using a preprocessor in the FEA software package (for example, one that interprets an IGES file). Similarly, the results of the FEA would be displayed to the user through a postprocessor in the FEA or CAD system.

In the following subsections, the above five-step process will be presented in greater detail. Mechanical stress, fluid flow, and heat transfer analysis problems will also be briefly addressed.

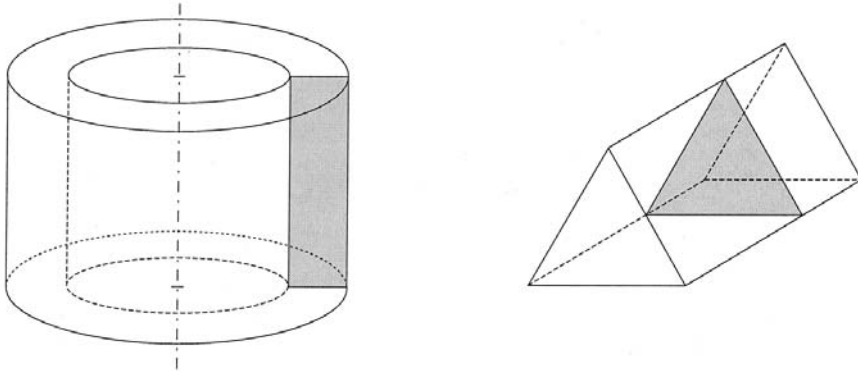
### **5.2.1 Discretization**

The first step in FEA is the discretization of the domain (region of interest) into a finite number of elements according to the approximation level required. Over the years, numerous automatic mesh generators have been developed in order to facilitate the task of discretization, which is normally carried out manually by FEA specialists. If the domain to be examined is symmetrical, the complexity of the computations can be significantly reduced, for example, by considering the problem only in 2-D or even analyzing only a half or a quarter of the solid model (Fig. 3).

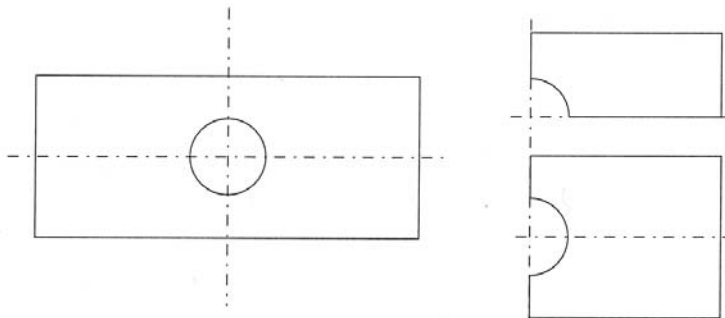
The shapes, sizes, and numbers of elements, as well as the location of the nodes, dictate the complexity of the finite-element model and greatly impact on the level of a solution's accuracy. Elements can be one-, two-, or three-dimensional (line, area, volume) (Fig. 4). The choice of the element type naturally depends on the domain to be analyzed: truss structures utilize line elements, two dimensional heat-transfer problems utilize area elements, and solid (nonsymmetrical) objects require volume elements. For area and



2D representation of a 3D problem.



Analyzing a  $\frac{1}{4}$  or a  $\frac{1}{2}$  of a problem.

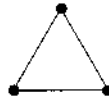


**FIGURE 3** Reduction in finite element representation.

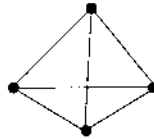
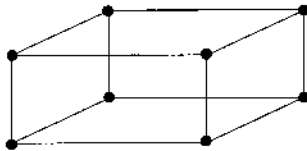
## Line



## Area



## Volume



**FIGURE 4** Basic element shapes.

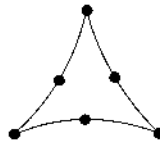
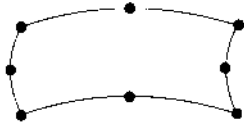
volume elements the boundary edges do not need to be linear. They can be curves (Fig. 5—*isoparametric representation*).

The size of the elements influences the accuracy of FEA—the smaller the size, the larger the number, the more accurate the solution will be, at the expense of computational effort. One can, however, choose different element sizes at different subregions of interest within the object (domain) (Fig. 6), i.e., a finer mesh, where a rapid change in the value of the variable is expected. It is also recommended that nodes be carefully placed, especially at discontinuity points and loading locations.

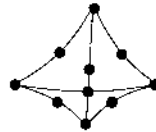
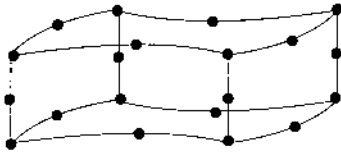
### 5.2.2 Interpolation

Finite-element modeling and analysis requires piecewise solution of the problem (for each element) through the use of an adopted interpolation function representing the behavior of the variable within each element. Polynomial approximation is the most commonly used method for this

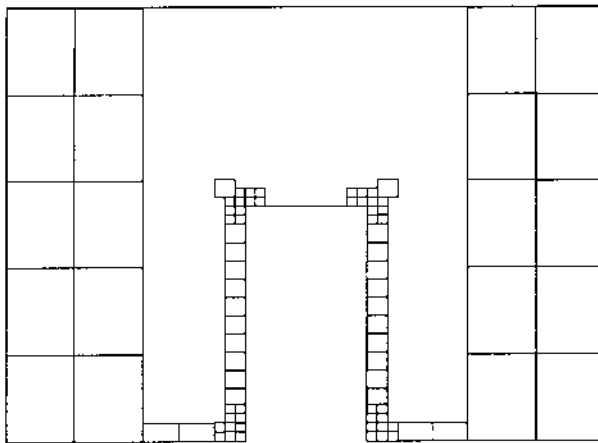
## Area



## Volume



**FIGURE 5** Curved elements.



**FIGURE 6** Elements of different size.

purpose. Let us, for example, consider a triangular (area) element, where the variable value can be expressed as a function of the Cartesian coordinates using different-order polynomial functions (Fig. 7): linear,

$$\phi(x, y) = \alpha_1 + \alpha_2x + \alpha_3y \quad (5.1)$$

and quadratic,

$$\phi(x, y) = \alpha_1 + \alpha_2x + \alpha_3y + \alpha_4x^2 + \alpha_5y^2 + \alpha_6xy \quad (5.2)$$

One would expect that as the element size decreases and the polynomial order increases, the solution would converge to the true solution at the limit. However, one should not attempt to achieve unreasonable accuracies that would not be needed by the designers/and engineers, who would normally interpret the results of the FEA and use them as part of their overall design parameter optimization process (satisfying a set of constraints and/or maximizing/minimizing an objective function). It is thus common to find simplex (first-order) or complex (second-order) elements in most FEA solutions in the manufacturing industry, and not higher orders.

For the two-dimensional simplex element given in Fig. 7 and defined by Eq. (5.1), the variable's nodal values (e.g.,  $i = 1, j = 2, k = 3$ ) are defined as

$$\begin{aligned} \phi_i &= \alpha_1 + \alpha_2x_i + \alpha_3y_i \\ \phi_j &= \alpha_1 + \alpha_2x_j + \alpha_3y_j \\ \phi_k &= \alpha_1 + \alpha_2x_k + \alpha_3y_k \end{aligned} \quad (5.3)$$

where ( $\alpha_1, \alpha_2$ , and  $\alpha_3$ ) are the coefficients of the first-order polynomial. These coefficients can be solved for, using the above system of equations

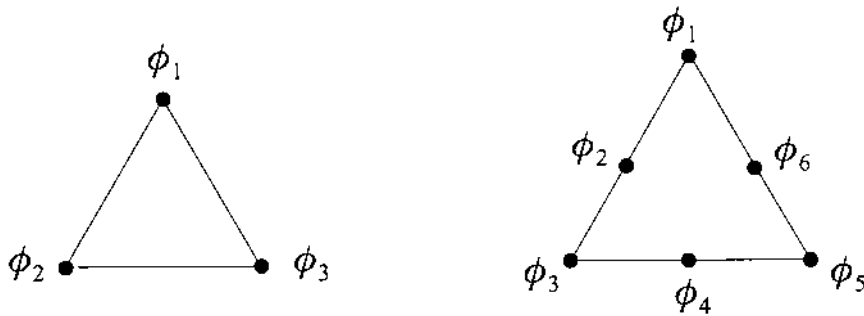


FIGURE 7 Two-dimensional element.

(i.e., three equations and three unknowns), in terms of the nodal coordinates and the function values at these nodes. Equation (5.1) can thus be rewritten as a function of the above nodal values as

$$\begin{aligned}\phi(x, y) &= N_i\phi_i + N_j\phi_j + N_k\phi_k, \\ &= [N]\{\phi\}\end{aligned}\quad (5.4)$$

where the elements of  $[N]$ , ( $N_i$ ,  $N_j$ , and  $N_k$ ), are functions of the  $(x, y)$  coordinate values of the three nodes,

$$\begin{aligned}N_i &= \frac{1}{2A}(a_i + b_ix + c_iy) \\ N_j &= \frac{1}{2A}(a_j + b_jx + c_jy)\end{aligned}\quad (5.5)$$

$$\begin{aligned}N_k &= \frac{1}{2A}(a_k + b_kx + c_ky) \\ A &= \frac{1}{2}(x_iy_j + x_jy_k + x_ky_i - x_iy_k - x_jy_i - x_ky_j)\end{aligned}\quad (5.6)$$

and

$$\begin{aligned}a_i &= x_jy_k - x_ky_j & a_j &= x_ky_i - x_iy_k & a_k &= x_iy_j - x_jy_i \\ b_i &= y_j - y_k & b_j &= y_k - y_i & b_k &= y_i - y_j \\ c_i &= x_k - x_j & c_j &= x_i - x_k & c_k &= x_j - x_i\end{aligned}\quad (5.7)$$

The value of  $\phi(x, y)$  at any point  $(x, y)$  is assumed to be scalar in Eq. (5.4) (e.g., temperature). However, in most engineering problems, the variable at a node would be vectorial in nature (e.g., displacement along  $x$  and  $y$ ). Thus the interpolation polynomial must also be defined accordingly in multidimensional space. For the simplex element above, let us assume that the variable  $\phi$  will have two components  $u$  and  $v$ , along the  $x$  and  $y$  directions, respectively (Fig. 8). Then, based on Eq. (5.4),

$$\begin{aligned}u(x, y) &= N_i\phi_{2i-1} + N_j\phi_{2j-1} + N_k\phi_{2k-1} \\ v(x, y) &= N_i\phi_{2i} + N_j\phi_{2j} + N_k\phi_{2k}\end{aligned}\quad (5.8)$$

where  $N_i$ ,  $N_j$ , and  $N_k$  are defined by Eq. (5.5), and the nodal values are defined as  $u_i = \phi_{2i-1}$ ,  $v_i = \phi_{2i}$ , etc.

### 5.2.3 Element Equations and Their Assembly

Derivation of the element equations depends on the application at hand and can be carried out using a number of different methods. Since (mechanical) stress analysis is the most common (mechanical) engineering analysis

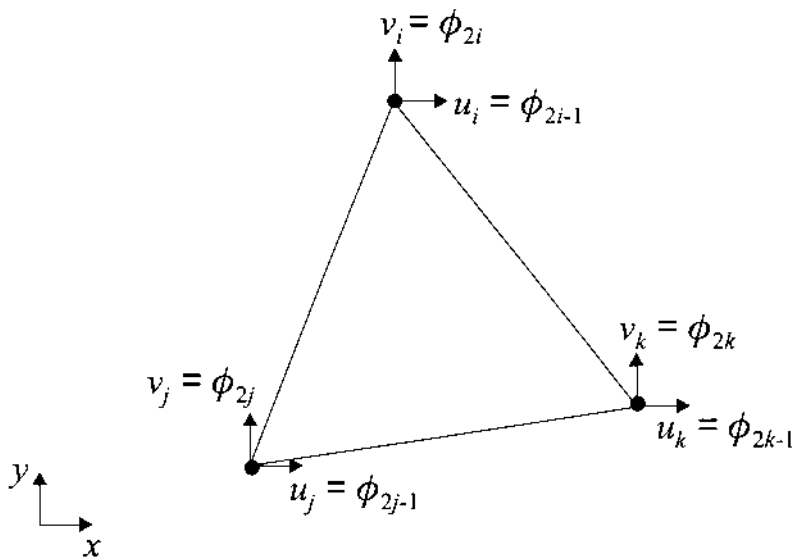


FIGURE 8 Two-dimensional simplex element.

problem, it will be utilized here as an example case study for the derivation of element equations. Other analysis problems will also be addressed in Sec. 5.2.5.

The three common modeling approaches used for elasticity analysis (i.e., stress analysis in the elastic domain) using finite elements are

*The Direct Approach:* Direct physical reasoning is utilized to derive the relationships for the variables considered. (This method is normally restricted to simple one-dimensional representations).

*The Variational Approach:* Calculus of variations is utilized for solving problems formulated in variational forms. It leads to approximate solutions of problems that cannot be formulated using the direct approach.

*The Weighted Residual Approach:* The governing differential equations of the problem are utilized for the derivation of the element's equations. (This method could be useful for problems such as fluid flow and mass transport, where we could readily have the governing differential equations and boundary conditions.)

## The Variational Approach for Stress Analysis

Let us consider a two-dimensional stress–strain relationship:

$$\{\varepsilon\} = \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{xy} \end{Bmatrix} = [C]\{\sigma\} + \{\varepsilon_0\} = [C] \begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{Bmatrix} + \begin{Bmatrix} \varepsilon_{xx0} \\ \varepsilon_{yy0} \\ \varepsilon_{xy0} \end{Bmatrix} \quad (5.9)$$

where  $[C]$  is a matrix of elastic coefficients,

$$[C] = \frac{1}{E} \begin{bmatrix} 1 & -\nu & 0 \\ -\nu & 1 & 0 \\ 0 & 0 & 2(1+\nu) \end{bmatrix} \quad (5.10)$$

and  $\{\varepsilon_0\}$  is the vector of initial strains.  $E$  is Young's modulus, and  $\nu$  is the Poisson ratio. Equation (5.9) can also be written as

$$\{\sigma\} = [D]\{\varepsilon\} - [D]\{\varepsilon_0\} \quad (5.11)$$

where, for plane strain,

$$[D] = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & \frac{1}{2}(1-\nu) \end{bmatrix} \quad (5.12)$$

The strain–displacement relationships are correspondingly defined as

$$\varepsilon_{xx} = \frac{\partial u}{\partial x} \quad \varepsilon_{yy} = \frac{\partial v}{\partial y} \quad \varepsilon_{xy} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \quad (5.13)$$

where  $u$  and  $v$  are displacements along the  $(x, y)$  directions, respectively, each of which are functions of the coordinates  $(x, y)$ .

Referring to finite-element displacement equations of a simplex, Eq. (5.8),

$$\begin{aligned} u(x, y) &= N_i u_i + N_j u_j + N_k u_k \\ v(x, y) &= N_i v_i + N_j v_j + N_k v_k \end{aligned} \quad (5.14)$$

or in the alternate notation for the nodal displacements, as in Eq. (5.8), Fig. 8,

$$\begin{Bmatrix} u \\ v \end{Bmatrix} = \begin{bmatrix} N_i & 0 & N_j & 0 & N_k & 0 \\ 0 & N_i & 0 & N_j & 0 & N_k \end{bmatrix} \begin{Bmatrix} u_{2i-1} \\ u_{2i} \\ u_{2j-1} \\ u_{2j} \\ u_{2k-1} \\ u_{2k} \end{Bmatrix} = [N]\{U\} \quad (5.15)$$

Using Eqs. (5.5), (5.13), and (5.15),

$$\begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{xy} \end{Bmatrix} = \frac{1}{2A} \begin{bmatrix} b_i & 0 & b_j & 0 & b_k & 0 \\ 0 & c_i & 0 & c_j & 0 & c_k \\ c_i & b_i & c_j & b_j & c_k & b_k \end{bmatrix} \{U\} = [B]\{U\} \quad (5.16)$$

The stiffness matrix for the (two-dimensional) simplex element is then defined by

$$[k] = \int_V [B]^T [D] [B] dV = [B]^T [D] [B] \int_V dV \quad (5.17)$$

where the volumetric integral in the above equation can be replaced with  $(tA)$ .  $t$  is the constant thickness of the element and  $A$  is the cross-sectional area.

Similarly, the element load vector due to initial strains,  $\{P_i\}$ , is defined as

$$\{P_i\} = \int_V [B]^T [D] \{\varepsilon_0\} dV = [B]^T [D] \{\varepsilon_0\} tA \quad (5.18)$$



and the element load vector due to body forces,  $\{P_b\}$ , is defined as

$$\{P_b\} = \int_V [N]^T \begin{Bmatrix} F_x \\ F_y \end{Bmatrix} dV = \frac{tA}{3} \begin{Bmatrix} F_x \\ F_y \\ F_x \\ F_y \\ F_x \\ F_y \end{Bmatrix} \quad (5.19)$$

where the vector  $\{F_x \ F_y\}^T$  is the body-force vector per unit volume.

Equations (5.17) (5.18) to (5.19) and the concentrated forces vector,  $\{P_c\}$ , can be combined to complete the derivation of the element equations (excluding pressures applied on the element) summed over the entire domain (all the elements,  $e = 1$  to  $E$ ).

$$[K]\{U\} = \{P\} \quad (5.20)$$

where

$$\{P\} = \sum_{e=1}^E (\{P_i\} + \{P_b\})^e + \{P_c\} \quad (5.21)$$

and

$$[K] = \sum_{e=1}^E [k]^e \quad (5.22)$$

As shown above, the assembly of element equations, Eq. (5.20), is the combination of the element stiffness matrices into one global stiffness matrix, summing all the force vector components into one global force vector. The compatibility requirement must be met during this assembly process, that is, the values of the nodal parameters are the same for nodes that are shared by multiple elements. If the element matrices and vectors were calculated in local coordinates, it would be necessary to transfer them to a global (world) coordinate system. (Naturally, in a computer-aided analysis environment all above-mentioned transactions would be carried out automatically by the appropriate software module.) One must, finally, add the boundary conditions (geometric/essential and free/natural) onto the system's (assembled) model.

## 5.2.4 Solution

The finite-element method is a numerical technique providing an approximate solution to the continuous problem that has been discretized. The solution process can be carried out utilizing different techniques that solve the equilibrium equations of the assembled system. Direct methods yield exact solutions after a finite number of operations. However, one must be aware of potential round-off and truncation errors when using such methods. Iterative methods, on the other hand, are normally robust to round-off errors and lead to better approximations after every iteration (when the process converges). Common solution methods include

*The Gaussian-Elimination “Direct” Method*, which is based on the triangularization of the system of equations (the coefficient matrices) and the calculation of the variable values by back-substitution.

*The Choleski Method*, which is a direct method for solving a linear system by decomposing the (normally symmetric) positive definite FEA matrices into lower and upper triangular matrices and calculation of the variable values by back-substitution.

*The Gauss–Seidel Method*, which is an iterative method primarily targeted for large systems, in which the system of equations is solved one equation at a time to determine a better approximation of the variable at hand based on the latest values of all other variables.

For solving eigenvalue problems, FEA solution methods include the power, Rayleigh–Ritz, Jacobi, Givens, and Householder techniques; while for propagation problems, solutions include the Runge–Kutta, Adams–Moulton, and Hamming methods.

## 5.2.5 Fluid Flow and Heat Transfer Problems

In heat transfer problems, determination of temperature distribution within a conducting body is paramount to our understanding of heat dissipation and potential development of significant thermal stresses. The basic governing equation for heat transfer problems is

$$\begin{aligned} &\text{Heat inflow during } dt \\ &= (\text{Heat outflow} + \text{Change in internal body energy}) \text{ during } dt \end{aligned}$$

Both heat conduction and heat convection phenomena can be modeled and analyzed using a finite-element method. As in the (mechanical) stress analysis case, the first task at hand is the selection of the element type and division of the domain of interest into  $E$  elements. The next task is the choice

of a temperature (variation) function within each element and to express it as a function of Cartesian coordinates and time. Next, the element conduction (or convection) matrix and equations can be developed using the variational approach. The last step in the formulation of the FEA problem is the assembly of the element equations and the incorporation of the boundary conditions to yield

$$[K]\{T\} = \{P\} \quad (5.23)$$

where  $[K]$  is the overall conduction (or convection) matrix,  $\{T\}$  is the nodal temperature vector, and  $\{P\}$  is the heat-source vector.

In fluid mechanics, FEA has been widely applied in the past two decades to laminar as well as turbulent flows of Newtonian fluids (whose viscosity is not a function of velocity). Recently, however, FEA has been also applied to non-Newtonian fluids, especially by users of polymers. FEA for fluid and heat flows are similar—the process starts with the meshing of the domain; choice of a potential function and derivation of the element equations follows this step; the element equations are, then, assembled to yield

$$[K]\{\phi\} = \{P\} \quad (5.24)$$

where  $\{\phi\}$  is the nodal velocity potential vector and  $\{P\}$  is the input potential vector; the definition of the stiffness matrix  $[K]$  is the same as in the cases of stress analysis and heat transfer analysis equations. Equation (5.24) can be solved, using any one of the methods mentioned in Sec. 5.2.4 for determining fluid velocity.

### 5.2.6 Commercial FEA Software

Commercial finite-element modeling and analysis packages can be categorized into comprehensive packages that provide FEA for several engineering fields, such as ANSYS, ALGOR, and MISC/NASTRAN, physical-phenomenon-specific packages that provide FEA for specific physical problems, such as FLUENT for computational fluid dynamics (CFD) problems and application-specific packages that specialize in unique engineering problems, such as MOLDFOW for injection-molding-related problems. All these CAE packages have been developed over the years to run on microcomputers (such as SUN) and lately on personal computers (mainly Windows-based platforms) as their CPU speeds become faster and RAM storage capability increases.

Although most FEA packages have evolved over the years in terms of the friendliness of their graphical user interfaces (GUIs) for domain modeling, it would be advisable to utilize the original CAD solid models of objects as our starting point and not to attempt to redefine these models within a

FEA package. At the opposite end of the spectrum, CAD packages have also significantly evolved in terms of their engineering analysis capabilities—SDRC (I-DEAS), for example, allows designers to run FEA on solid models for mechanical stress and heat transfer analyses. However, for complex problems (complex geometry, layered materials, two-phase flows, etc.), it would be advisable to utilize specialized FEA packages.

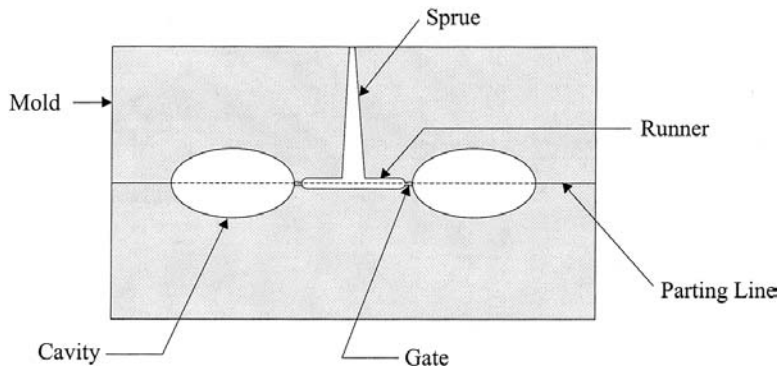
As discussed earlier, effective mesh generation is the precursor to any accurate FEA analysis. This step can be carried out on a CAD workstation. The outcome (domain model) can be transferred to a commercial FEA package using an available data-exchange standard (IGES or STEP) and prepared for analysis by being processed through a preprocessor. At this stage, the user is expected to add onto the geometric model the necessary boundary conditions (including loads) as well as material properties. Preprocessors are expected to verify the finite-element model by checking for distorted elements and modeling errors.

Once the solution of the problem has been obtained, a postprocessor can be run to examine the results (preferably graphically) via the GUI of a CAD system that would allow us to manipulate the output effectively—view it from different angles, cross-section it, etc. It is important to remember that the outcome of the FEA analysis is primarily a metric to be fed into an optimization algorithm that would search for the best design parameters.

### **5.2.7 An Example—Computer-Aided Injection Molding Analysis**

Injection molding is a common plastics-processing technique used for the manufacture of containers, toys, electronic packaging, and automotive products. As simple as the process may be thought at first glance (i.e., filling of mold cavities with liquid polymer by injection at high speeds and pressures), the design of the mold is quite complex owing to the concurrent existence of several physical phenomena: flow of non-Newtonian fluid, heat transfer, and thermal stresses. A good mold design can significantly benefit from the usage of a FEA-based computer software package for the analysis of all the mentioned physical phenomena. Some of the design issues are discussed below, prior to a discussion of available commercial packages in the analysis of mold filling, cooling, and warpage issues (Fig. 9):

*Cavities:* Although the number of cavities on a mold base may be treated as a purely economic issue, their locations and arrangement affect injection pressure and clamping force. Furthermore, as discussed earlier in Chap. 3, part features (such as draft angles, sharp edges, the geometry of ribs) affect the flow of the molten material, the cooling time of the part, and its warpage.



**FIGURE 9** Mold-filling elements.

*Gates:* The type, geometry, location, and number of gates affect flow patterns during filling.

*Sprue and runners:* A mold-filling objective is minimization of the distance traveled by the molten material before it reaches the cavities. Other conflicting considerations include prolonged cycle times owing to excessive sizes of the sprue and runners and creation of undesirable flow patterns owing to insufficient diameters of the runners, etc.

*Cooling:* Effective cooling provides short cycle times and prevents defects such as warpage, poor surface quality, or even burn marks.

The injection molding process starts with the filling of the cavities with molten (normally thermoplastic) polymer and some additional melt to compensate for shrinkage. The fluid flow during the filling process is predominantly of the shear-flow type that is driven by pressure to overcome the melt's resistance to flow. Naturally, fluid temperature is an important factor, as the mobility of the polymer chains increases with increased temperature. Using FEA, the flow of fluid through the runner/gate/cavity assembly can be analyzed as a function of time, using a solid model of the overall system generated (and automatically meshed) on a CAD system. The two leading commercial FEA packages that can be used for this purpose are MOLDFLOW (Australia) and CMOLD (U.S.A.). Both packages can carry out automatic mesh generation and simulate mold filling.

During the mold filling analysis process, one can also examine the heat transfer characteristics of the mold configuration at hand (i.e., a mold design with specific locations and geometries for the sprue/runners/gates/cavities),

concurrently with the fluid flow analysis mentioned above. Heat loss occurs through the circulating coolant (in the cooling channels) as well as through the mold surroundings. A considerable amount of cycle time is spent on cooling the molded parts. Thus one must examine temperature distributions during the filling process as well as during the postfilling period for different mold configurations and filling parameters. However, one must realize that mold cooling is a complicated problem and nonuniform mold cooling results in undesirable part warpage (during ejection) due to nonuniform residual stresses. Another important factor in part warpage is, of course, variations in shrinkage (due to flow orientation, differential pressure, etc.) Both the commercial FEA packages mentioned above provide users with corresponding modules for thermal analysis and warpage determination capabilities—(MF/COOL and MF/WARP by MOLDFLOW, and C-COOL and C-WARP by CMOLD).

Over the past two decades, many researchers have developed optimal mold design techniques that utilize the above-mentioned (and other) finite-element-based mold flow analysis tools, in order to relieve dependence on expert opinions and other heuristics. It should be mentioned here that most mold makers still heavily depend on human judgment rather than utilizing analytical methods in optimizing mold designs.

### 5.3 OPTIMIZATION

Engineering design is an iterative process, in which the outcome of the analysis phase is fed back to the synthesis phase for the determination of optimal design parameter values. That is, the parametric design stage is carried out under the auspices of a search algorithm whose objective is to optimize (through CAE analysis) an objective function (e.g., performance, cost, weight) by varying the product design parameters at hand. Most optimization problems encountered in engineering design are of the constrained type. An optimal solution (“best” parameter values) is selected among all feasible designs, subject to limits imposed on the variable design parameters. The variables are, normally, of a continuous type—i.e., they can be assigned any one of the infinite possible values. For example, the thickness of a vessel is a geometric (dimensional) continuous variable and can be assigned any (floating-point) value within a given range ( $t_{\min}$  to  $t_{\max}$ ), while attempting to optimize a desired objective function.

As discussed above, a typical optimization problem aims at maximizing/minimizing an overall objective function  $Z$ , which is a function of a number of variables,  $x_i$ ,  $i = 1$  to  $n$  subject to ( $j$ ) equality and ( $k$ ) inequality

constraints placed on the variables, whose optimal values we are trying to determine:

$$\min Z = Z(x_1, x_2, \dots, x_n) \quad (5.25)$$

subject to  $\phi_j(x_1, \dots, x_n) = 0$  and  $\psi_k(x_1, \dots, x_n) \leq \psi_k^{\max}$

In most engineering design cases, the design team must decide what to optimize (i.e., what to choose as an objective function) and formulate the other desired specifications as equality and inequality constraints. However, in numerous cases, the team may be faced with a situation in which multiple objectives (sometimes in conflict with each other) must be optimized. Two common solutions to this problem are (1) to prioritize the objective functions and formulate a multilevel (nested) optimization problem, and (2) to combine the functions into a single weighted sum (overall) objective function. In the former case, a priority could be to reduce the number of fasteners used, for example, followed by determining the optimal geometrical parameters for each fastener. Thus one could achieve a required attachment strength by increasing the number of fasteners or by increasing their dimensions. At any iteration, for a given number of fasteners considered by the outer level of a two-loop search, the inner loop would select the (best) parameter values that would maximize fastening strength. Once determined, the search would return to the outer loop and check whether the number of fasteners could be further reduced. Otherwise, the optimal solution is considered to be reached.

For the latter multiobjective function case, an example task could be to attempt to maximize component life while minimizing the manufacturing cost:

$$\min Z = w_1 \left( \frac{1}{L_n(x_1, \dots, x_n)} \right) + w_2 (C_n(x_1, \dots, x_n)) \quad (5.26)$$

where  $L_n$  is the estimated (normalized) product life,  $C_n$  is the estimated (normalized) product cost, both functions of the variables  $x_1$  to  $x_n$ , and  $w_1$  to  $w_n$  are weighting coefficients. The choice of the weighting coefficients is application dependent.

In the above optimization problems, whether a single- or a multi-objective formulation, one must carefully examine the variables as well. Although in most design cases the variables would be of the continuous type, as mentioned in the above example, they could also be of a discrete or integer type. An objective function could have both types of variables or only one type. Solution techniques proposed in the literature, some of which are to be discussed herein, would be sensitive to the types of the variables.

Other factors that strongly affect the choice of a solution (search) method would include the expected behavior of the objective function—whether it has one or multiple extrema (single-mode, multimode functions); the order of the function (linear versus nonlinear) and whether its derivative can be calculated; and lastly the restrictions on the search domain—whether the problem is constrained or not constrained.

### 5.3.1 Overview of Optimization Techniques

Optimization procedures are widely applied in engineering, spanning from design to planning and to control. In this section, although we will overview a number of existing optimization solution techniques, our focus will be on those that are most useful in the engineering design cycle of synthesis → analysis → synthesis. Furthermore, among the most pertinent techniques, only a few will be detailed—it is expected that users of optimization will have to review carefully the complete existing spectrum on available search techniques.

It is important to acknowledge here that the field of numerical optimization reached recognition only after the 1940s and has been widely researched concurrently with the significant developments in computing hardware and software. The pioneers in the field (during the 1950s to the early 1980s) were W. C. Davidon, M. J. Powell, R. Fletcher, P. E. Gill, L. A. Wolsey, and G. L. Nemhauser, to mention a few. They and others classified optimization methods broadly into two main categories: continuous versus integer and combinatoric. In this section, our focus will be on the first category; the latter category deals with “process” problems, such as sequencing and network-flow analysis, in the context of planning for manufacturing.

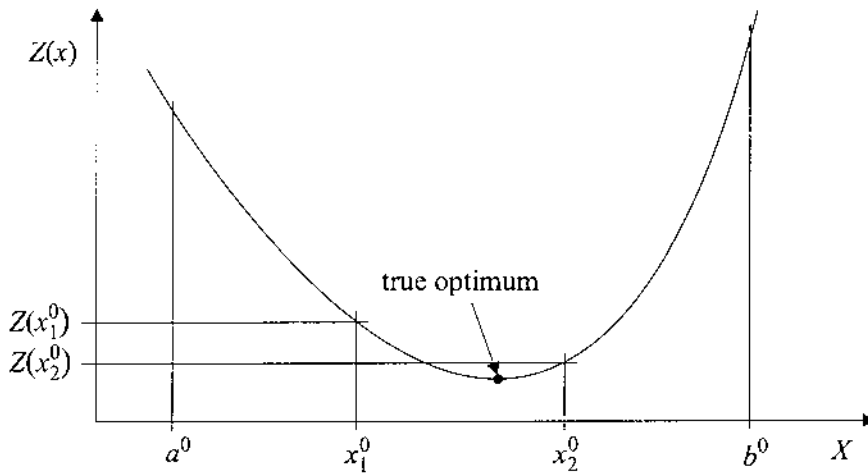
### 5.3.2 Single-Variable Functions—Numerical Methods

Let us consider a simple case: a product’s characteristic is a function of one design variable,  $Z(x)$ . Let us further assume that  $Z(x)$  is a continuous function and can only be evaluated through a numerical simulation, such as FEA, and that derivatives of the function cannot be obtained. Based on experience (or preliminary investigation), we also know that  $Z(x)$  is a single-mode function (one extremum). The problem at hand is to determine the optimal  $x$  value that would minimize the objective function in the range  $[a, b]$  for  $x$ :

$$\min Z = Z(x)$$

subject to  $a-x \leq 0$  and  $x-b \geq 0$ .





**FIGURE 10** Golden section search—an example. First iteration.

The most popular numerical technique that can be used for the solution of the above optimization problem is known as the golden section search technique. It successively divides the available search range, specified as  $[a^i, b^i]$  at every iteration, into two sections proportioned approximately as (0.319 and 0.681) and discards the one that does not contain the minimum. The number 0.681 has been discovered as the most efficient way for internal division by numerous mathematicians (whose derivation can be found in optimization books, such as the one by J. Kowalik and M. R. Osborne). The golden section search starts by choosing two  $x$  values,  $x_1^0$  and  $x_2^0$ , which divide the interval  $[a^0, b^0]$  into three thirds (Fig. 10), and proceeds to the evaluation of the function at these points,  $Z(x_1^0)$  and  $Z(x_2^0)$  (for example, through FEA), respectively.

The golden section iterative process compares the two function values, evaluated at  $x_1^i$  and  $x_2^i$  in Step  $i$ , and narrows the search domain accordingly:

- (1) If  $Z(x_1^i) > Z(x_2^i)$   
 $a^{i+1} = x_1^i, b^{i+1} = b^i$   
 $x_1^{i+1} = x_2^i$       and       $x_2^{i+1} = b^{i+1} - 0.319(b^{i+1} - a^{i+1})$       (5.27)
- (2) If  $Z(x_2^i) > Z(x_1^i)$   
 $a^{i+1} = a^i, b^{i+1} = x_2^i$

$$x_1^{i+1} = a^{i+1} + 0.319 (b^{i+1} - a^{i+1}) \quad \text{and} \quad x_2^{i+1} = x_1^i \quad (5.28)$$

The above search is normally terminated based on the size of the latest interval as a percentage of the initial interval,

$$\frac{(a^{i+1} - b^{i+1})}{(a^0 - b^0)} \leq \varepsilon \quad (5.29)$$

where  $\varepsilon$  is denoted as the convergence threshold.

A competing search method is the Fibonacci search technique that utilizes a number set named after the mathematician Leonardo of Pisa (also known as Fibonacci) who lived from 1180 to 1225. The Fibonacci numbers are defined as follows:

$$F_0 = F_1 = 1, \quad F_i = F_{i-1} + F_{i-2} \quad \text{for } i > 1 \quad (5.30)$$

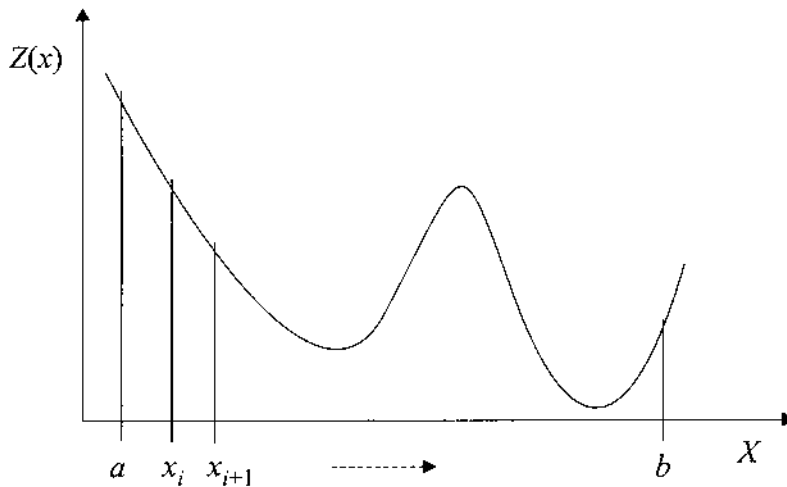
The search divides the search domain of length  $L = a - b$  into three sections by a proportion defined by

$$\Delta_i = L_{i-1} \frac{F_{i-2}}{F_i} \quad (5.31)$$

Either of the two outlying sections is eliminated based on the function values at  $x_1^i$  and  $x_2^i$  as in the golden section search method.

Although the Fibonacci method has been shown to have a slight advantage over the golden section search technique, the former requires an advanced knowledge of the size of the Fibonacci set (based on the desired  $\varepsilon$ ). However, neither can cope with functions that may have multiple extrema. In such cases, one may have to search the entire domain, starting at one end and proceeding to the other at fixed increments in order to determine all the extrema and choose the variable value corresponding to the global extremum (Fig. 11). Over the years, numerous supplementary algorithms have been proposed in order to accelerate such brute force searches based on the availability of additional function values, normally obtained using a random search. Such supplementary algorithms allow the user to increase the size of the increments when it is suspected that the search is a distance away from an extremum (global or local).

Once can, naturally, argue the benefit of using any search technique in determining the minimum of a one-dimensional function at a time when it appears that we have “infinite” computing power, as opposed to using an exhaustive (brute force) method, when we test many, many  $x$  values. The counter-argument to the use of a brute force method would be that although the function may have only one variable, the function evaluations using, for example, FEA can consume enormous amounts of time if the search is



**FIGURE 11** Multimode functions.

carried out in an ad hoc or random manner. The computation time problem would rapidly worsen for multivariable functions.

### 5.3.3 Multivariable Functions—Numerical Methods

Let us consider a product characteristic that is a function of multiple variables,  $Z(x_1, x_2, \dots, x_n)$ . Let us further assume that  $Z(x)$  is a continuous function and can only be evaluated through a numerical simulation (e.g., FEA), and that derivatives of the function cannot be obtained. The function is known to be single mode (the case of the existence of multiple extrema will be discussed at the end of this subsection), and there exist no restrictions on the variables. This optimization problem is called multivariable, single-mode, and unconstrained. However, despite all these simplifications, the “curse of dimensionality” increases the difficulty of solving the problem (compared to a single-variable function) hyperexponentially.

Although many solution techniques have been proposed over the years for the above problem, there does not exist a clear measure of efficiency in their comparison. Thus engineers are recommended to test several methods for their specific problem in regard to efficiency of convergence and choose the most suitable one. Most recommended search techniques vary the values of the variables simultaneously (in contrast to one at a time) and select the

next point of evaluation based on past functional data. The general steps of a sequential search technique can be noted as follows:

1. Select one (or several) feasible point(s)—a point is a vertex of all variables,  $\{x\}_0$  or  $\{x\}_1, \{x\}_2, \dots, \{x\}_k$ . (If the function has multiple extrema, these points should be widely separated.)
2. Evaluate the objective function at the initial point(s).
3. Based on the search technique utilized, choose the next feasible point.
4. Evaluate the objective function at this new point.
5. Compare the newest function value with earlier values and return to Step (3) if the search has not yet converged to the optimal solution,  $\{x\}_{\text{opt}}$ .

The specific search method reviewed in this section is the simplex method developed by J. A. Nelder and R. Mead, which lends itself to be adopted for constrained problems. The method has been often referred to as the flexible polyhedron search technique. As the name implies, the search utilizes a polyhedron in the hyperspace of the (multiple) variables. The simplex starts with four feasible vertices\* labeled as follows:

$\mathbf{x}_h$  is the (multivariable) vertex that corresponds to  $f(\mathbf{x}_h) = \max_i f(\mathbf{x}_i)$ , i.e., the highest function value, for  $i$  vertices considered.

$\mathbf{x}_l$  is the vertex that corresponds to  $f(\mathbf{x}_l) = \min_i f(\mathbf{x}_i)$ , i.e., the lowest function value, for  $i$  vertices considered.

$\mathbf{x}_s$  is the vertex that corresponds to  $f(\mathbf{x}_s) = \max_i f(\mathbf{x}_i), i \neq h$ , i.e., the second-highest function value, for  $i$  vertices considered.

$\mathbf{x}_o$  is the centroid vertex of all  $\mathbf{x}_i, i \neq h$ .

$$\mathbf{x}_o = \frac{1}{k} \sum_{\substack{i=1 \\ i \neq h}}^{k+1} \mathbf{x}_i \quad (5.32)$$

As mentioned above, we will consider the simplex at hand for determining the next “point” (vertex) in our quest for the optimal variable values  $\mathbf{x}_{\text{opt}}$ . Once the initial simplex is constructed,

1. We first try a “reflection” operation to determine the next point as

$$\mathbf{x}_r = (1 + \alpha)\mathbf{x}_o - \alpha\mathbf{x}_h \quad (5.33)$$

where  $\alpha > 0$  is a user chosen reflection coefficient.

---

\* Each vertex is the set of all the variables  $(x_1, \dots, x_n)$  for the multivariable function considered. For clarity, bold lettering,  $\mathbf{x}$ , is used for  $\{x\}$  in the description of the algorithm.

2. If  $f(\mathbf{x}_s) > f(\mathbf{x}_r) > f(\mathbf{x}_l)$ , we set  $\mathbf{x}_h = \mathbf{x}_r$  and return to Step 1;  
 If  $f(\mathbf{x}_r) < f(\mathbf{x}_l)$ , we may expect the discovery of an even better point in the direction of  $\mathbf{x}_r - \mathbf{x}_o$  and thus proceed to Step 3;  
 If  $f(\mathbf{x}_h) > f(\mathbf{x}_r) > f(\mathbf{x}_s)$ , we set  $\mathbf{x}_h = \mathbf{x}_l$  and carry out a “contraction” by proceeding to Step 4.  
 If  $f(\mathbf{x}_r) > f(\mathbf{x}_h)$ , we contract without replacement, Step 5.
3. We “expand” as,

$$\mathbf{x}_e = \gamma \mathbf{x}_r + (1 - \gamma) \mathbf{x}_o \quad (5.34)$$

where  $\gamma > 1$  is a user chosen expansion coefficient.

4. If  $f(\mathbf{x}_l) > f(\mathbf{x}_e)$ , we set  $\mathbf{x}_h = \mathbf{x}_e$  and we return to Step 1.
- Otherwise, we set  $\mathbf{x}_h = \mathbf{x}_r$  and we return to Step 1.

5. We “contract” as,

$$\mathbf{x}_c = \beta \mathbf{x}_h + (1 - \beta) \mathbf{x}_o \quad (5.35)$$

where  $0 < \beta < 1$  is a user chosen contraction coefficient.

6. If  $f(\mathbf{x}_h) > f(\mathbf{x}_c)$ , we set  $\mathbf{x}_h = \mathbf{x}_c$  and we return to Step 1.
- Otherwise, the simplex is “shrunk” as in Step 7.

7. We “shrink” the simplex as,

$$\mathbf{x}_i = \frac{1}{2} (\mathbf{x}_i + \mathbf{x}_l) \quad (5.36)$$

where  $i = h, l$ , and  $s$ , and return to Step 1.

In the above algorithm, after each new function evaluation, the convergence criterion, as given below, must be checked:

$$\left\{ \frac{1}{k} \sum_{i=1}^{k+1} (f(\mathbf{x}_i) - f(\mathbf{x}_o))^2 \right\}^{1/2} < \varepsilon \quad (5.37)$$

where  $\varepsilon$  is the convergence threshold.

Nelder and Mead propose to have the initial polyhedron (simplex) be configured as widely spread as possible over the domain, in order not to be stalled at a local optimum. However, as with all other multivariable search techniques, there exists no convergence guarantee to the global optimum, even when the procedure is run several times starting with a different initial simplex.

The next problem to consider is having a constrained optimization problem, where the variables are restricted by numerous inequality constraints. A technique specifically designed for the Nelder–Mead method is the flexible tolerance method. A primary characteristic of this method is that if, during a reflection process, the new point lies within the infeasible region, it is not immediately dismissed as it would be with other constrained

methods that build firewalls around the feasible domains and use penalty functions. (A penalty function would artificially increase the function value of the new infeasible point and thus severely discourage a search in the direction considered.) The flexible tolerance method, however, uses a more tolerant firewall approach: the firewall is initially placed further than its true location and moved back progressively as the search continues. That is, infeasible solutions are tolerated at lesser degrees as the search continues.

There are many other techniques that can be utilized for multivariable functions, especially if function derivatives can be evaluated (or approximated). The most effective methods are the Davidon–Fletcher–Powell method and the Rosenbrock method. As mentioned before, it is the application at hand that determines which method to use and how and when to stop it (i.e., declare convergence).

### **5.3.4 Integer and Combinatorial Optimization in Brief**

Throughout this section, our primary focus has been on optimization problems in which the variables were allowed to be assigned any value in the continuous domain. In some design (and many manufacturing planning) instances, however, the variables can only be assigned an integer value (e.g., number of fasteners). One may attempt to use solutions developed for continuous optimization in these cases as well, and round the values to the nearest integer, though the solutions may no longer be optimal or even feasible. Thus it is strongly recommended that engineers should use suitable techniques for such problems—labeled as integer programming and combinatorics.

#### **Gomory’s Algorithm**

This method solves an integer (linear programming) problem by using a constraint during the linear programming solution of the problem, which only allows integer solutions. The constraint mentioned is often called as the “Gomory’s cutting plane.”

#### **Branch and Bound Method**

Also known as the truncated enumeration method, this technique can be utilized to solve any bounded integer optimization problem. In such a method, the total set of solutions is systematically subdivided into smaller subsets; as the search progresses, some are discarded without examining any solutions within the subsets.

## Dynamic Programming

This technique can be used to solve optimization problems with continuous as well as discrete (integer) variables. A problem of  $n$  variables is converted into  $n$  one-variable problems, which are then solved sequentially, yielding an optimal solution for the overall problem. Not all problems, however, lend themselves to being formulated as multiple subproblems. The three basic conditions for such a formulation can be summarized as follows:

The overall problem is divisible into  $n$  subproblems (stages), each of which (preferably) involves only one variable.

Relationships can be established between each pair of successive subproblems ( $i-1$  and  $i$ ).

The solution of the  $i^{\text{th}}$  stage (subproblem) depends only on the solution of the  $(i-1)^{\text{th}}$  stage.

## Simulated Annealing

This method is a stochastic (probabilistic) method with a characteristic of occasionally allowing the choice of a sequence (in a combinatoric sequencing problem) that does not improve on the current minimum value of the objective function in order to prevent stalling at a local optimum. As in the flexible tolerance method for constrained (continuous) nonlinear problems, however, this acceptance of inferior sequences is reduced as the number of iterations increases.

## Genetic Algorithms

Such algorithms attempt to mimic the natural selection process, where, in a population, the fittest is allowed to survive and reproduce. The optimization starts with a random generation of a population of genomes (different sequences of integer values for each of the variables) and their fitness evaluation (i.e., objective function value). “Offsprings” (new solutions) are generated by pairing two existing genomes that have the highest probability of yielding a better solution. (Different genetic algorithms (GAs) use different methods for generating offsprings.) Thus GAs can be classified as stochastic-type methods suitable for the sequencing of integer variables.

In conclusion to this subsection, an engineer once again should not assume that, since integer programming problems deal with finite spaces of solutions, one could use a brute force approach of simply testing every combination. The two primary counter-arguments are (1) the curse of dimensionality and (2) time spent in evaluating each solution.

## REVIEW QUESTIONS

1. Define *engineering analysis*. How would computer hardware and software increase the efficiency of engineering analysis (i.e., computer-aided engineering analysis – CAE)?
2. What is a prototype of an engineering product?
3. How does the virtual prototyping process differ from physical prototyping? Discuss advantages/disadvantages of each.
4. How could CAE facilitate the prototyping process of engineering products?
5. What is the role of virtual reality in the prototyping process of engineering products?
6. Why would an engineer use finite element modeling and analysis (FEM/A) to obtain an approximate numerical solution?
7. Describe briefly the sequential process of FEM/A.
8. Would you recommend the use of FEM/A in manufacturing environments by engineers who do not have strong backgrounds in the field but have access to commercial software packages with FEM/A capability? Discuss both sides of the argument.
9. What is the role of optimization in the design process? In a typical engineering optimization process, what would the objective function, variables, and constraints represent, respectively?
10. What is the role of computers in an optimization process? Why should a design engineer attempt to choose the most efficient search method for the optimization of the variable values, as opposed to testing every possible combination? Consider problems with continuous and discrete variables.
11. In constrained optimization, strict firewalls could lengthen the search time significantly or may even prevent convergence. Discuss the use of strict firewalls versus tolerant, dynamic firewalls.

## DISCUSSION QUESTIONS

1. Computers have often been considered as machines that automatically process information, versus (automated) production machines that process materials. Explain.
2. Discuss the role of computers in the different stages of the (iterative) design process: concept development, synthesis, and analysis.
3. In the mid part of the 20th century, design, planning, and control of manufacturing processes were argued to be activities that had more to do with the application of experience/knowledge gathered by experts in



contrast to the use of any mathematical models or systematic analysis techniques. As a contrary argument, discuss the basic physical phenomena (e.g., non-Newtonian fluid mechanics and heat transfer) that govern common manufacturing processes (e.g., casting of metals and molding of plastics) and their use in the design, planning, and control of these processes. Furthermore, discuss the role of computer-aided engineering (CAE) in these analyses.

4. Geometric modeling is a design task that can be efficiently implemented using a commercial CAD package. Discuss the importance of geometric modeling for the engineering analysis stage of design. Address issues such as surface modeling versus solid modeling.
5. A prototype of a product must meet some or all of its engineering specifications. These specifications can be geometric or functional. In this context, a large number of material-additive techniques were developed during the period 1980 to 2000 and classified as rapid prototyping (RP) processes. Justify through examples (production techniques and product features) why such layered manufacturing techniques may indeed be used to yield prototypes.
6. Discuss the concept of progressively increasing *cost of changes* to a product as it moves from the design stage to full production and distribution. How could virtual prototyping minimize potential future changes to a product, especially for those products that have very short development cycles?
7. Finite element modeling and analysis (FEM/A) methods have been developed to cope with the engineering analysis of complex product geometries and/or physical phenomena. Although these techniques would yield near-true results, the efficiency and correctness of software code may only be verified for simple scenarios. In this context, discuss the use of FEM/A during the (iterative) design process (i.e., synthesis–analysis) for the determination of optimal design parameter values. Furthermore, discuss the potential need for verifying the FEA results with physical tests.
8. Many conflicting interests affect the design of a product (i.e., its geometry, materials, configuration, etc.) as well as its fabrication/assembly techniques: profit, quality, environmental concerns, etc. As an engineer, when analyzing potential solutions, you may address the design process as a typical multivariable, multiobjective, constrained optimization problem. Discuss the utilization of such a mathematical approach to the determination of an optimal design versus utilizing an ad hoc approach.
9. Process planning in assembly (in its limited definition) refers to the optimal selection of an assembly sequence of the components. For

example, solving the traveling salesperson problem in the population of electronics boards. Although there are a number of search techniques for the solution of such problems, they could all benefit from the existence of a good initial (guess) solution. Discuss the role of group technology (GT) on the identification of such initial (guesses) sequences of assembly.

10. Design of experiments (DOE) is a statistical approach that can be used in the design of physical or simulation-based experiments for the determination of optimal variable values. Such factorial based experiments help engineers in the narrowing of the field of search to those parameters that have the greatest impact on the performance of the product as well as limiting the combinatoric number of variations of these variables. Discuss the role of DOE in the engineering analysis of a product.
11. Analysis of a production process via computer-aided modeling and simulation can lead to an optimal process plan with significant savings in production time and cost. Discuss the issue of time and resources spent on obtaining an optimal plan and the actual (absolute) savings obtained due to this optimization: for example, spending several hours in planning to reduce production time from 2 minutes to 1 minute. Present your analysis as a comparison of one-of-a-kind production versus mass production.
12. The use of design features has long been considered as improving the overall synthesis and analysis stages of products due to the potential of encapsulating additional nongeometric data, such as process plans, in the definition of such features. Discuss feature-based design, in which the user, through some recognition/extraction process, can access and retrieve individual similar or identical features on earlier product designs and utilize them for the design and engineering analysis of the product at hand.
13. Computer-aided engineering (CAE) analysis has been practiced as long as there have been computers in existence (since the early 1950s). Current commercial CAD software packages, which have been in existence only since the late 1970s, complement (older and new) CAE software with powerful graphical user interfaces (GUI) for the modeling of parts/products and graphical display of analysis results. In this context, discuss the complementary and potentially future competitive use of CAE and CAD packages in the design (synthesis and analysis) of engineering products.
14. In a majority of companies that do not employ engineers, and in some that do, decision making for design and/or production tasks relies on existing in-house expertise (human experts and/or written knowledge).

Discuss the diminishing role of experts in decision making as we improve our ability to model and analyze systems/phenomena and reach optimal solutions based on analytical/heuristic search techniques running on ever more powerful computers. Would such a trend lead to the disappearance of experience/knowledge-based “experts” and create a dependence on professionals (engineers/scientists with different backgrounds) who can develop and run these automated decision making systems?

15. The period 1980–2000 has witnessed the dismantling of the vertical integration of many large manufacturers and rapid movement towards supply-chain relationships. Discuss the impact of recent technological and management developments on this movement: short product lives, concurrent engineering carried out in the virtual domain (i.e., distributed design), minimization of in-process inventories, etc.

## BIBLIOGRAPHY

- Argyris, J. H., Kelsey, S. (Oct. 1954 to May 1955). Energy theorems and structural analysis. *Aircraft Engineering* Vols. 26 and 27.
- Benzley, S. E., Merkley, K., Blacker, T. D., Schoof, L. (May 1995). Pre- and post-processing for the finite element method. *Journal of Finite Elements in Analysis and Design* 19(4):243–260.
- Bernhardt, Ernest., ed. (1983). *CAE: Computer Aided Engineering for Injection Molding*. Munich, Germany: Hanser.
- Beveridge, Gordon S. G., Schechter, Robert S. (1970). *Optimization: Theory and Practice*. New York: McGraw-Hill.
- Boeing Co. Boeing News Release: Boeing 777 Digital Design Process Earns Technology Award. [http://www.boeing.com/news/releases/1995/news.release\\_950614-a.html](http://www.boeing.com/news/releases/1995/news.release_950614-a.html), and <http://www.boeing.com/commercial/777family/cdfacts.html>
- Cooper, Leon N., Steinberg, David (1970). *Introduction to Methods of Optimization*. Philadelphia: Saunders.
- Courant, R. (1943). Variational methods for the solution of problems of equilibrium and vibrations. *Bulletin of the American Mathematical Society* 49:1–23.
- Dai, Fan (1998). *Virtual Reality for Industrial Applications*. New York: Springer-Verlag.
- Dieter, George Ellwood (2000). *Engineering Design: A Materials and Processing Approach*. New York: McGraw-Hill.
- Edney, R. C. (1991). *Computer Aided Engineering for Mechanical Engineers*. New York: Prentice Hall.
- Fagan, M. J. (1992). *Finite Element Analysis: Theory and Practice*. New York: John Wiley.
- Fletcher, Roger., ed. (1969). *Optimization: 1968 Symposium of the Institute of Mathematics and Its Applications*. London, England: London Academic Press.

- Fletcher, Roger (1987). *Practical Methods of Optimization*. New York: John Wiley.
- Gen, Mitsuo (2000). *Genetic Algorithms and Engineering Optimization*. New York: John Wiley.
- Gomes de Sá, A., Zachmann, G. (June 1999). Virtual reality as a tool for verification of assembly and maintenance processes. *J. of Computers and Graphics* Vol. 23(3):389–403.
- Gomorooy, R.E. (1958). Outline of an algorithm for integer solutions to linear programs. *Bull. Amer. Math. Soc.* 64:275–278.
- Huang, Hou-Cheng, Usmani, Asif S. (1994). *Finite Element Analysis for Heat Transfer: Theory and Software*. New York: Springer-Verlag.
- Irani, R. K., Kim, B. H., Dixon, J. R. (Feb. 1995). Towards automated design of the feed system on injection molds BY integrating CAE, iterative redesign and features. *ASME Trans., Journal of Engineering for Industry* 117(1): 72–77.
- Jacobs, Paul F. (1992). *Rapid Prototyping and Manufacturing: Fundamentals of Stereolithography*. Dearborn, MI: Society of Manufacturing Engineers.
- Kobayashi, Shiro, Oh, Soo-Ik, Altan, Taylan (1989). *Metal Forming and the Finite-Element Method*. New York: Oxford University Press.
- Kowalik, J., Osborne, M. R (1968). *Methods for Unconstrained Optimization Problems*. New York: Elsevier.
- Lee, K., Gadh, R. (Oct. 1998). Destructive disassembly to support virtual prototyping. *IIE Transactions* 30(10):959–972.
- Lepi, Steven M. (1998). *Practical Guide to Finite Elements: A Solid Mechanics Approach*. New York: Marcel Dekker.
- Logan, Daryl L. (1997). *A First Course in The Finite Element Method Using ALGOR*. Boston: PWS.
- Machover, Carl., ed. (1996). *The CAD/CAM Handbook*. New York: McGraw-Hill.
- Nedler, J. A., Mead, R. (1965). A simplex method for function minimization. *Computer Journal*, (No. 7):308.
- Nee, A. Y. C., Ong, S. K., Wang, Y. G. (1999). *Computer Applications in Near Net-Shape Operations*. New York: Springer-Verlag.
- Nemhauser, George L., Wolsey, Laurence A. (1988). *Integer and Combinatorial Optimization*. New York: John Wiley.
- Onalir, B., Kaftanoglu, B., Balkan, T. (June 1997). Application of computer-aided injection molding simulation for thermoplastic materials. CIRP, International Conference on Design and Production of Dies and Molds, Istanbul, Turkey, pp. 57–64.
- Paviani, D. A., Himmelblau, D. M. (1969). Constrained nonlinear optimization by heuristic programming. *J. of Operations Research* 17:872–882.
- Pironneau, Olivier (1989). *The Finite Element Methods for Fluids*. New York: John Wiley.
- Sabbagh, Karl (1996). *21st Century Jet: The Making and Marketing of the Boeing 777*. New York: Scribner.
- SAE (1998). *Virtual Prototyping: Computer Modeling, and Computer Simulation*. Warrendale, PA: Society of Automotive Engineers.
- Schmitz, B. (May 1998). Automotive design races ahead. *J. of Computer-Aided Engineering* 17(5):38–46.

- Siddique, Z., Rosen, D. W. (Dec. 1997). A virtual prototyping approach to product disassembly reasoning. *J. of Computer-Aided Design* 29(12): 847–860.
- Singiresu, S. Rao (1999). *The Finite Element Method in Engineering*. Boston: Butterworth Heinemann.
- Sutherland, I. (Dec. 1968). A head-mounted three dimensional display. *AFIPS, Proc. of Fall Joint Conference on Computers* 33:757–764 (San Francisco, CA).
- Turner, M. J., Clough, R. W., Martin, H. C., Topp, L. J. (1957). Stiffness and deflection analysis of complex structures. *J. of Aeronautical Sciences* 23: 805–842.
- Ulrich, K. T., Eppinger, S. D. (2000). *Product Design and Development*. Boston: McGraw-Hill.
- Vose, Michael D. (1999). *The Simple Genetic Algorithm: Foundations and Theory*. Cambridge, MA: MIT Press.
- Wang, K. K., Himasekhar, K., Chiang, H. H., Jong, W. R., Wang, V. W. Integrated CAE of injection molding using a three-layer approach. Soc. of Plastics Engineers, Proceedings of ANTEC, Vol. 37., Brookfield, CT, pp. 267–273.
- Warwick, Kevin, Gray, John, Roberts, David (1993). *Virtual Reality in Engineering*. London: Institution of Electrical Engineers.
- Wilson, John R. (1996). *Virtual Reality for Industrial Application: Opportunities and Limitations*. Nottingham, UK: Nottingham University Press.
- Yevko, V., Park, C. B., Zak, G., Coyle, T. W., Benhabib, B. (Dec. 1998). Cladding formation in laser-beam fusion of metal powder. *J. of Rapid Prototyping* 4(4):168–184.
- Zeid, Ibrahim (1991). *CAD/CAM Theory and Practice*. New York: McGraw-Hill.
- Zou, Q., Horney, J. R. (1994). Minimize injection molded plastic part warpage by applying warpage analysis in mold design. Soc. of Plastics Engineers, Proceedings of ANTEC, Brookfield, CT, pp. 1104–1110.