

# Introduction to Optimization

## 1.1 INTRODUCTION

Optimization is the act of obtaining the best result under given circumstances. In design, construction, and maintenance of any engineering system, engineers have to take many technological and managerial decisions at several stages. The ultimate goal of all such decisions is either to minimize the effort required or to maximize the desired benefit. Since the effort required or the benefit desired in any practical situation can be expressed as a function of certain decision variables, *optimization* can be defined as the process of finding the conditions that give the maximum or minimum value of a function. It can be seen from Fig. 1.1 that if a point  $x^*$  corresponds to the minimum value of function  $f(x)$ , the same point also corresponds to the maximum value of the negative of the function,  $-f(x)$ . Thus, without loss of generality, optimization can be taken to mean minimization since the maximum of a function can be found by seeking the minimum of the negative of the same function. There is no single method available for solving all optimization problems efficiently. Hence, a number of optimization methods have been developed for solving different types of optimization problems.

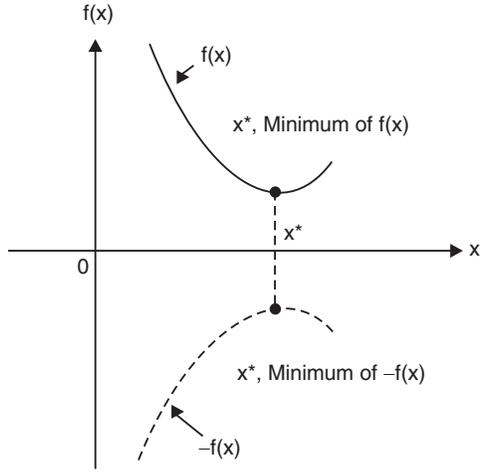


Figure 1.1 Minimum of  $f(x)$  is same as maximum of  $-f(x)$ .

The optimum seeking methods are also known as *mathematical programming techniques* and are generally studied as a part of operations research. *Operations research* is a branch of mathematics

concerned with the application of scientific methods and techniques to decision making problems and with establishing the best or optimal solutions. Table 1.1 lists various mathematical programming techniques together with other well-defined areas of operations research. The classification given in Table 1.1 is not unique; it is given mainly for convenience.

Mathematical programming techniques are useful in finding the minimum of a function of several variables under a prescribed set of constraints. Stochastic process techniques can be used to analyze problems described by a set of random variables having known probability distributions. Statistical methods enable one to analyze the experimental data and build empirical models to obtain the most accurate representation of the physical situation. This book deals with the theory and application of mathematical programming techniques suitable for the solution of engineering design problems.

**Table 1.1** Methods of Operations Research

<i>Mathematical Programming Techniques</i>	<i>Stochastic Process Techniques</i>	<i>Statistical Methods</i>
Calculus methods	Statistical decision theory	Regression analysis
Calculus of variations	Markov processes	Cluster analysis, pattern recognition
Nonlinear programming	Queueing theory	Design of experiments
Geometric programming	Renewal theory	Discriminate analysis
Quadratic programming	Simulation methods	(factor analysis)
Linear programming	Reliability theory	
Dynamic programming		
Integer programming		
Stochastic programming		
Separable programming		
Multiobjective programming		
Network methods: CPM and PERT		
Game theory		
Simulated annealing		
Genetic algorithms		
Neural networks		

## 1.2 HISTORICAL DEVELOPMENT

The existence of optimization methods can be traced to the days of Newton, Lagrange, and Cauchy. The development of differential calculus methods of optimization was possible because of the contributions of Newton and Leibnitz to calculus. The foundations of calculus of variations, which deals with the minimization of functionals, were laid by Bernoulli, Euler, Lagrange, and Weirstrass. The method of optimization for constrained problems, which involves the addition of unknown multipliers, became known by the name of its inventor, Lagrange. Cauchy made the first application of the steepest descent method to solve unconstrained minimization problems. Despite these early contributions, very little progress was made until the middle of the twentieth century, when high-speed digital computers made implementation of the optimization procedures possible and stimulated further research on new methods. Spectacular advances followed, producing a massive literature on optimization

techniques. This advancement also resulted in the emergence of several well-defined new areas in optimization theory.

It is interesting to note that the major developments in the area of numerical methods of unconstrained optimization have been made in the United Kingdom only in the 1960s. The development of the simplex method by Dantzig in 1947 for linear programming problems and the announcement of the principle of optimality in 1957 by Bellman for dynamic programming problems paved the way for development of the methods of constrained optimization. Work by Kuhn and Tucker in 1951 on the necessary and sufficiency conditions for the optimal solution of programming problems laid the foundations for a great deal of later research in nonlinear programming. The contributions of Zoutendijk and Rosen to nonlinear programming during the early 1960s have been very significant. Although no single technique has been found to be universally applicable for nonlinear programming problems, work of Carroll and Fiacco and McCormick allowed many difficult problems to be solved by using the well-known techniques of unconstrained optimization. Geometric programming was developed in the 1960s by Duffin, Zener, and Peterson. Gomory did pioneering work in integer programming, one of the most exciting and rapidly developing areas of optimization. The reason for this is that most real-world applications fall under this category of problems. Dantzig and Charnes and Cooper developed stochastic programming techniques and solved problems by assuming design parameters to be independent and normally distributed.

The desire to optimize more than one objective or goal while satisfying the physical limitations led to the development of multiobjective programming methods. Goal programming is a well-known technique for solving specific types of multiobjective optimization problems. The goal programming was originally proposed for linear problems by Charnes and Cooper in 1961. The foundations of game theory were laid by von Neumann in 1928 and since then the technique has been applied to solve several mathematical economics and military problems. Only during the last few years has game theory been applied to solve engineering design problems.

Simulated annealing, genetic algorithms, and neural network methods represent a new class of mathematical programming techniques that have come into prominence during the last decade. Simulated annealing is analogous to the physical process of annealing of solids. The genetic algorithms are search techniques based on the mechanics of natural selection and natural genetics. Neural network methods are based on solving the problem using the efficient computing power of the network of interconnected “neuron” processors.

### 1.3 ENGINEERING APPLICATIONS OF OPTIMIZATION

Optimization, in its broadest sense, can be applied to solve any engineering problem. To indicate the wide scope of the subject, some typical applications from different engineering disciplines are given below.

1. Design of aircraft and aerospace structures for minimum weight
2. Finding the optimal trajectories of space vehicles
3. Design of civil engineering structures such as frames, foundations, bridges, towers, chimneys, and dams for minimum cost
4. Minimum-weight design of structures for earthquake, wind, and other types of random loading
5. Design of water resources systems for maximum benefit
6. Optimal plastic design of structures
7. Optimum design of linkages, cams, gears, machine tools, and other mechanical components
8. Selection of machining conditions in metal-cutting processes for minimum production cost

9. Design of material handling equipment such as conveyors, trucks, and cranes for minimum cost
10. Design of pumps, turbines, and heat transfer equipment for maximum efficiency
11. Optimum design of electrical machinery such as motors, generators, and transformers
12. Optimum design of electrical networks
13. Shortest route taken by a salesperson visiting various cities during one tour
14. Optimal production planning, controlling, and scheduling
15. Analysis of statistical data and building empirical models from experimental results to obtain the most accurate representation of the physical phenomenon
16. Optimum design of chemical processing equipment and plants
17. Design of optimum pipeline networks for process industries
18. Selection of a site for an industry
19. Planning of maintenance and replacement of equipment to reduce operating costs
20. Inventory control
21. Allocation of resources or services among several activities to maximize the benefit
22. Controlling the waiting and idle times and queueing in production lines to reduce the costs
23. Planning the best strategy to obtain maximum profit in the presence of a competitor
24. Optimum design of control systems

#### 1.4 STATEMENT OF AN OPTIMIZATION PROBLEM

An optimization or a mathematical programming problem can be stated as follows:

$$\text{Find } \mathbf{X} = \left\{ \begin{array}{c} x_1 \\ x_2 \\ \vdots \\ x_n \end{array} \right\} \text{ which minimizes } f(\mathbf{X})$$

subject to the constraints  $g_j(\mathbf{X}) \leq 0, j = 1, 2, \dots, m$  ... (1.1)

$$l_j(\mathbf{X}) = 0, j = 1, 2, \dots, p$$

where  $\mathbf{X}$  is an  $n$ -dimensional vector called the *design vector*;  $f(\mathbf{X})$  is termed the *objective function*, and  $g_j(\mathbf{X})$  and  $l_j(\mathbf{X})$  are known as *inequality* and *equality* constraints, respectively. The number of variables  $n$  and the number of constraints  $m$  and/or  $p$  need not be related in any way. The problem stated in Eq. (1.1) is called a *constrained optimization problem*.<sup>†</sup> Some optimization problems do not involve any constraints and can be stated as:

$$\text{Find } \mathbf{X} = \left\{ \begin{array}{c} x_1 \\ x_2 \\ \vdots \\ x_n \end{array} \right\} \text{ which minimizes } f(\mathbf{X}) \quad \dots(1.2)$$

Such problems are called *unconstrained optimization problems*.

<sup>†</sup> In the mathematical programming literature, the equality constraints  $l_j(\mathbf{X}) = 0, j = 1, 2, \dots, p$  are often neglected, for simplicity, in the statement of a constrained optimization problem, although several methods are available for handling problems with equality constraints.

### 1.4.1 Design Vector

Any engineering system or component is defined by a set of quantities some of which are viewed as variables during the design process. In general, certain quantities are usually fixed at the outset and these are called *preassigned parameters*. All the other quantities are treated as variables in the design process and are called *design* or *decision variables*  $x_i$ ,  $i = 1, 2, \dots, n$ . The design variables are collec-

tively represented as a design vector  $\mathbf{X} = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{Bmatrix}$ . As an example, consider the design of the gear pair

shown in Fig. 1.2, characterized by its face width  $b$ , number of teeth  $T_1$  and  $T_2$ , center distance  $d$ , pressure angle  $\psi$ , tooth profile, and material. If center distance  $d$ , pressure angle  $\psi$ , tooth profile, and material of the gears are fixed in advance, these quantities can be called *preassigned parameters*. The

remaining quantities can be collectively represented by a design vector  $\mathbf{X} = \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{Bmatrix} b \\ T_1 \\ T_2 \end{Bmatrix}$ . If there are

no restrictions on the choice of  $b$ ,  $T_1$ , and  $T_2$ , any set of three numbers will constitute a design for the gear pair. If an  $n$ -dimensional Cartesian space with each coordinate axis representing a design variable  $x_i$  ( $i = 1, 2, \dots, n$ ) is considered, the space is called the *design variable space* or simply, *design space*. Each point in the  $n$ -dimensional design space is called a *design point* and represents either a possible or an impossible solution to the design problem. In the case of the design of a gear pair, the design point

$\begin{Bmatrix} 1.0 \\ 20 \\ 40 \end{Bmatrix}$ , for example, represents a possible solution, whereas the design point  $\begin{Bmatrix} 1.0 \\ -20 \\ 40.5 \end{Bmatrix}$  represents an impossible solution since it is not possible to have either a negative value or a fractional value for the number of teeth.

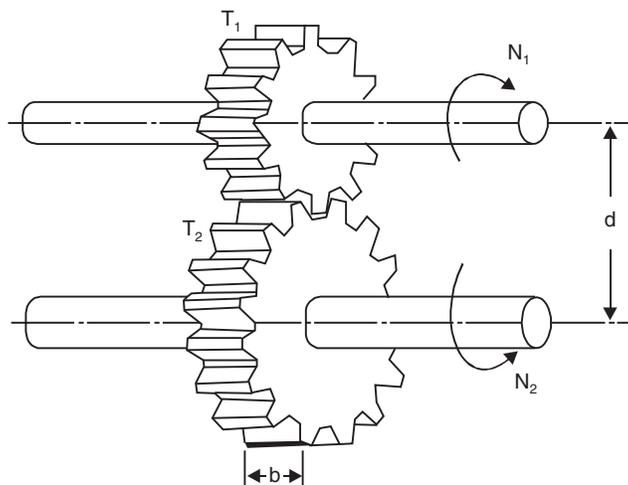


Figure 1.2 Gear pair in mesh.

### 1.4.2 Design Constraints

In many practical problems, the design variables cannot be chosen arbitrarily; rather, they have to satisfy certain specified functional and other requirements. The restrictions that must be satisfied to produce an acceptable design are collectively called *design constraints*. Constraints that represent limitations on the behavior or performance of the system are termed *behavior* or *functional constraints*. Constraints that represent physical limitations on design variables such as availability, fabricability, and transportability are known as *geometric* or *side constraints*. For example, for the gear pair shown in Fig. 1.2, the face width  $b$  cannot be taken smaller than a certain value, due to strength requirements. Similarly, the ratio of the numbers of teeth,  $T_1/T_2$ , is dictated by the speeds of the input and output shafts,  $N_1$  and  $N_2$ . Since these constraints depend on the performance of the gear pair, they are called behavior constraints. The values of  $T_1$  and  $T_2$  cannot be any real numbers but can only be integers. Further, there can be upper and lower bounds on  $T_1$  and  $T_2$  due to manufacturing limitations. Since these constraints depend on the physical limitations, they are called side constraints.

### 1.4.3 Constraint Surface

For illustration, consider an optimization problem with only inequality constraints  $g_j(\mathbf{X}) \leq 0$ . The set of values of  $\mathbf{X}$  that satisfy the equation  $g_j(\mathbf{X}) = 0$  forms a hypersurface in the design space and is called a *constraint surface*. Note that this is an  $(n - 1)$ -dimensional subspace, where  $n$  is the number of design variables. The constraint surface divides the design space into two regions: one in which  $g_j(\mathbf{X}) < 0$  and the other in which  $g_j(\mathbf{X}) > 0$ . Thus the points lying on the hypersurface will satisfy the constraint  $g_j(\mathbf{X})$  critically, whereas the points lying in the region where  $g_j(\mathbf{X}) > 0$  are infeasible or unacceptable, and the points lying in the region where  $g_j(\mathbf{X}) < 0$  are feasible or acceptable. The collection of all the constraint surfaces  $g_j(\mathbf{X}) = 0$ ,  $j = 1, 2, \dots, m$ , which separates the acceptable region is called the *composite constraint surface*.

Figure 1.3 shows a hypothetical two-dimensional design space where the infeasible region is indicated by hatched lines. A design point that lies on one or more than one constraint surface is called a *bound point*, and the associated constraint is called an *active constraint*. Design points that do not lie

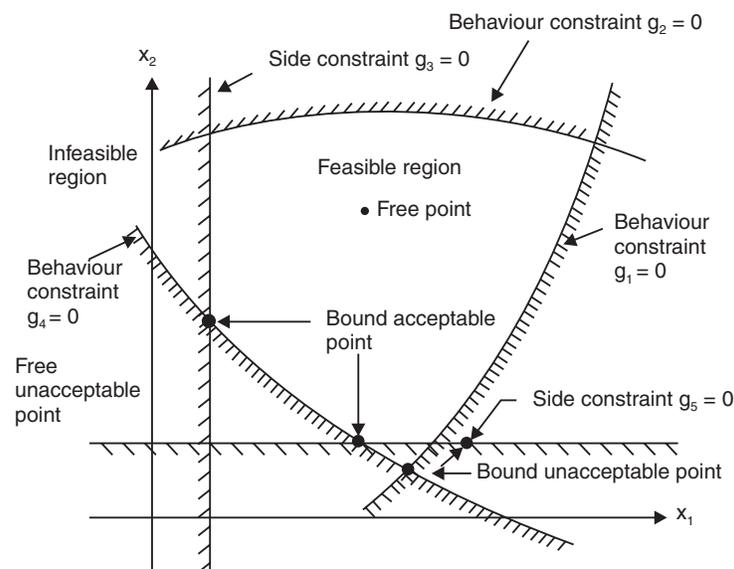


Figure 1.3 Constraint surfaces in a hypothetical two-dimensional design space.

on any constraint surface are known as *free points*. Depending on whether a particular design point belongs to the acceptable or unacceptable region, it can be identified as one of the following four types:

1. Free and acceptable point
2. Free and unacceptable point
3. Bound and acceptable point
4. Bound and unacceptable point

All four types of points are shown in Fig. 1.3.

#### 1.4.4 Objective Function

The conventional design procedures aim at finding an acceptable or adequate design which merely satisfies the functional and other requirements of the problem. In general, there will be more than one acceptable design, and the purpose of optimization is to choose the best one of the many acceptable designs available. Thus a criterion has to be chosen for comparing the different alternative acceptable designs and for selecting the best one. The criterion with respect to which the design is optimized, when expressed as a function of the design variables, is known as the *criterion* or *merit* or *objective function*. The choice of objective function is governed by the nature of problem. The objective function for minimization is generally taken as weight in aircraft and aerospace structural design problems. In civil engineering structural designs, the objective is usually taken as the minimization of cost. The maximization of mechanical efficiency is the obvious choice of an objective in mechanical engineering systems design. Thus the choice of the objective function appears to be straightforward in most design problems. However, there may be cases where the optimization with respect to a particular criterion may lead to results that may not be satisfactory with respect to another criterion. For example, in mechanical design, a gearbox transmitting the maximum power may not have the minimum weight. Similarly, in structural design, the minimum-weight design may not correspond to minimum stress design, and the minimum stress design, again, may not correspond to maximum frequency design. Thus the selection of the objective function can be one of the most important decisions in the whole optimum design process.

In some situations, there may be more than one criterion to be satisfied simultaneously. For example, a gear pair may have to be designed for minimum weight and maximum efficiency while transmitting a specified horse-power. An optimization problem involving multiple objective functions is known as a *multiobjective programming problem*. With multiple objectives there arises a possibility of conflict, and one simple way to handle the problem is to construct an overall objective function as a linear combination of the conflicting multiple objective functions. Thus if  $f_1(\mathbf{X})$  and  $f_2(\mathbf{X})$  denote two objective functions, construct a new (overall) objective function for optimization as

$$f(\mathbf{X}) = \alpha_1 f_1(\mathbf{X}) + \alpha_2 f_2(\mathbf{X}) \quad \dots(1.3)$$

where  $\alpha_1$  and  $\alpha_2$  are constants whose values indicate the relative importance of one objective function relative to the other.

#### 1.4.5 Objective Function Surfaces

The locus of all points satisfying  $f(\mathbf{X}) = c = \text{constant}$  forms a hypersurface in the design space, and for each value of  $c$  there corresponds a different member of a family of surfaces. These surfaces, called *objective function surfaces*, are shown in a hypothetical two-dimensional design space in Fig. 1.4.

Once the objective function surfaces are drawn along with the constraint surfaces, the optimum point can be determined without much difficulty. But the main problem is that as the number of design

variables exceeds two or three, the constraint and objective function surfaces become complex even for visualization and the problem has to be solved purely as a mathematical problem. The following example illustrates the graphical optimization procedure.

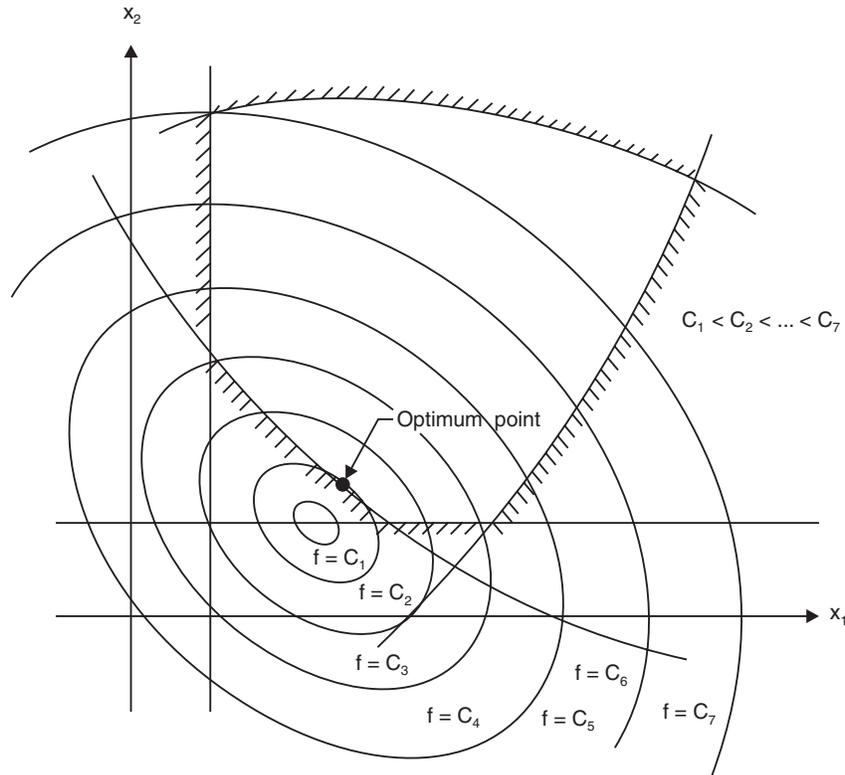


Figure 1.4 Contours of the objective function.

**Example 1.1:** Design a uniform column of tubular section (Fig. 1.5) to carry a compressive load  $P = 2500 \text{ kg}_f$  for minimum cost. The column is made up of a material that has a yield stress ( $\sigma_y$ ) of  $500 \text{ kg}_f/\text{cm}^2$ , modulus of elasticity ( $E$ ) of  $0.85 \times 10^6 \text{ kg}_f/\text{cm}^2$ , and density ( $\rho$ ) of  $0.0025 \text{ kg}_f/\text{cm}^3$ . The length of the column is 250 cm. The stress induced in the column should be less than the buckling stress as well as the yield stress. The mean diameter of the column is restricted to lie between 2 and 14 cm, and columns with thicknesses outside the range 0.2 to 0.8 cm are not available in the market. The cost of the column includes material and construction costs and can be taken as  $5W + 2d$ , where  $W$  is the weight in kilograms force and  $d$  is the mean diameter of the column in centimeters.

**Solution:** The design variables are the mean diameter ( $d$ ) and tube thickness ( $t$ ):

$$\mathbf{X} = \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} = \begin{Bmatrix} d \\ t \end{Bmatrix} \quad \dots(\text{E}_1)$$

The objective function to be minimized is given by

$$f(\mathbf{X}) = 5W + 2d = 5\rho l\pi dt + 2d = 9.82x_1x_2 + 2x_1 \quad \dots(\text{E}_2)$$

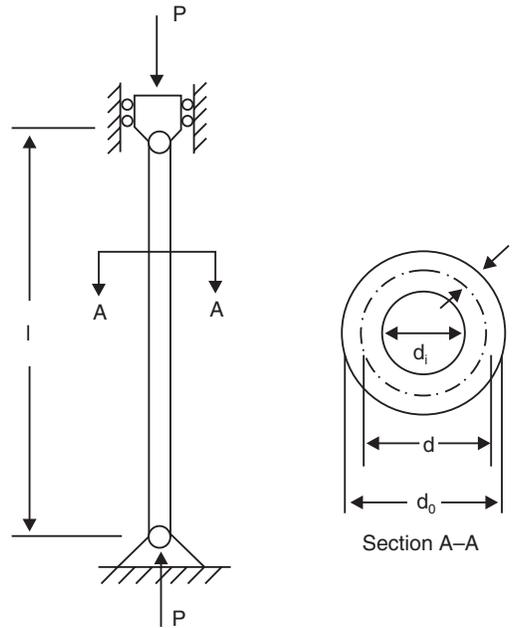


Figure 1.5 Tubular column under compression.

The behaviour constraints can be expressed as  
 stress induced  $\leq$  yield stress  
 stress induced  $\leq$  buckling stress

The induced stress is given by

$$\text{induced stress} = \sigma_i = \frac{P}{\pi dt} = \frac{2500}{\pi x_1 x_2} \quad \dots(E_3)$$

The buckling stress for a pin-connected column is given by

$$\text{buckling stress} = \sigma_b = \frac{\text{Euler buckling load}}{\text{cross-sectional area}} = \frac{\pi^2 EI}{l^2} \frac{1}{\pi dt} \quad \dots(E_4)$$

where  $I$  = second moment of area of the cross-section of the column.

$$\begin{aligned} &= \frac{\pi}{64} (d_0^4 - d_i^4) \\ &= \frac{\pi}{64} (d_0^2 + d_i^2)(d_0 + d_i)(d_0 - d_i) \\ &= \frac{\pi}{64} [(d+t)^2 + (d-t)^2] \cdot [(d+t) + (d-t)] [(d+t) - (d-t)] \\ &= \frac{\pi}{8} dt (d^2 + t^2) = \frac{\pi}{8} x_1 x_2 (x_1^2 + x_2^2) \quad \dots(E_5) \end{aligned}$$

Thus the behavior constraints can be restated as

$$g_1(\mathbf{X}) = \frac{2500}{\pi x_1 x_2} - 500 \leq 0 \quad \dots(E_6)$$

$$g_2(\mathbf{X}) = \frac{2500}{\pi x_1 x_2} - \frac{\pi^2 (0.85 \times 10^6) (x_1^2 + x_2^2)}{8(250)^2} \leq 0 \quad \dots(E_7)$$

The side constraints are given by

$$2 \leq d \leq 14$$

$$0.2 \leq t \leq 0.8$$

which can be expressed in standard form as

$$g_3(\mathbf{X}) = -x_1 + 2.0 \leq 0 \quad \dots(E_8)$$

$$g_4(\mathbf{X}) = -x_1 - 14.0 \leq 0 \quad \dots(E_9)$$

$$g_5(\mathbf{X}) = -x_2 - 0.2 \leq 0 \quad \dots(E_{10})$$

$$g_6(\mathbf{X}) = x_2 - 0.8 \leq 0 \quad \dots(E_{11})$$

Since there are only two design variables, the problem can be solved graphically as shown below.

First, the constraint surfaces are to be plotted in a two-dimensional design space where the two axes represent the two design variables  $x_1$  and  $x_2$ . To plot the first constraint surface, we have

$$g_1(\mathbf{X}) = \frac{2500}{\pi x_1 x_2} - 500 \leq 0$$

that is,

$$x_1 x_2 \geq 1.593$$

Thus the curve  $x_1 x_2 = 1.593$  represents the constraint surface  $g_1(\mathbf{X}) = 0$ . This curve can be plotted by finding several points on the curve. The points on the curve can be found by giving a series of values to  $x_1$  and finding the corresponding values of  $x_2$  that satisfy the relation  $x_1 x_2 = 1.593$ :

$x_1$	2.0	4.0	6.0	8.0	10.0	12.0	14.0
$x_2$	0.7965	0.3983	0.2655	0.1990	0.1593	0.1328	0.1140

These points are plotted and a curve  $P_1 Q_1$  passing through all these points is drawn as shown in Fig. 1.6, and the infeasible region, represented by  $g_1(x) > 0$  or  $x_1 x_2 < 1.593$ , is shown by hatched lines.<sup>†</sup>

Similarly, the second constraint  $g_2(\mathbf{X}) \leq 0$  can be expressed as  $x_1 x_2 (x_1^2 + x_2^2) \geq 47.3$  and lying on the constraint surface  $g_2(\mathbf{X}) = 0$  can be obtained as follow

$$[\text{for } x_1 x_2 (x_1^2 + x_2^2) = 47.3]:$$

<sup>†</sup> The infeasible region can be identified by testing whether the origin lies in the feasible region.