Chapter 6
(10 and 11 of 2nd ed.)
Numerical Methods for Constrained Optimum Design

Basic Concepts Related to Algorithms for Constrained Problems

All numerical methods discussed in this chapter are based on the following iterative prescription as also given in Eqs. (8.1) and (8.2) for unconstrained problems:

Vector form

$$x^{(k+1)} = x^{(k)} + \Delta x^{(k)}, \quad k = 0, 1, 2, \ldots \qquad (10.2)$$

Component form

$$x_i^{(k+1)} = x_i^{(k)} + \Delta x_i^{(k)}, \quad i = 1 \text{ to } n; \qquad k = 0, 1, 2, \ldots \qquad (10.3)$$

$$\Delta x^{(k)} = \alpha_k \, d^{(k)}$$

The starting design can be feasible or infeasible. If it is inside the feasible set as Point *A*, then there are two possibilities:

1. The gradient of the cost function vanishes at the point so it is an unconstrained stationary point. We need to check the sufficient condition for optimality of the point.
2. If the current point is not stationary, then we can reduce the cost function by moving along a descent direction, say, the steepest descent direction (**-*c***) as shown in Fig. We continue such iterations until either a constraint is encountered or an unconstrained minimum point is reached.

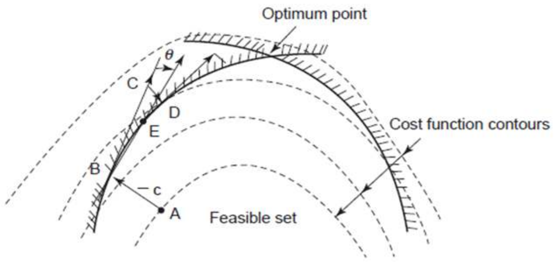


FIGURE 10-1 Conceptual steps of constrained optimization algorithms initiated from a feasible point.

When the starting point is infeasible, as Point *A,* then one strategy is to correct constraints to reach the constraint boundary at Point *B.* From there, the strategies described in the preceding paragraph can be followed to reach the optimum point. This is shown in Path 1 in Fig.

The second strategy is to iterate through the infeasible region by computing directions that take successive design points closer to the optimum point, shown as Path 2 in Fig.

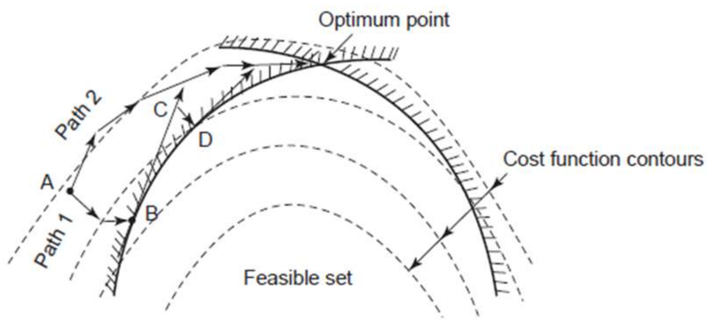


FIGURE 10-2 Conceptual steps of constrained optimization algorithms initiated from an infeasible point.

Most of the algorithms based on these strategies have the following four basic steps:

1. Linearization of cost and constraint functions about the current design point.

2. Definition of a search direction determination subproblem using the linearized functions.

3. Solution of the subproblem that gives a search direction in the design space.

4. Calculation of a step size to minimize a descent function in the search direction.

یاد آوری از فصل ۲:

قیود فعال/ غیر فعال/ نقض شده

$$g_i(x) \le 0 \begin{cases} g_i(x^*) = 0 \\ g_i(x^*) < 0 \end{cases}$$

قید در نقطه‌ی x^* فعال است.

قید در نقطه‌ی x^* غیر فعال است.

$g_i(x^*) > 0$ قید در نقطه‌ی x^* نقض شده است.

$h_j(x^*) = 0$ قیدهای مساوی یا فعالند

$h_j(x^*) \neq 0$ یا نقض شده

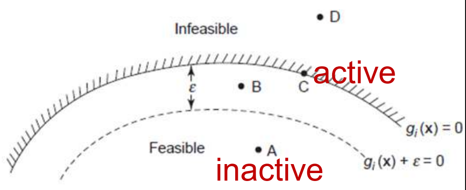
10.1.2 Constraint Status at a Design Point

An inequality constraint can be either active, ε-active, violated, or inactive at a design point. On the other hand, an equality constraint is either active or violated at a design point. The precise definitions of the status of a constraint at a design point are needed in the development and discussion of numerical methods.

Active Constraint

An inequality constraint $g_i(\boldsymbol{x}) \leq 0$ is said to be active (or tight) at a design point $\boldsymbol{x}^{(k)}$ if it is satisfied as an equality at that point, i.e., $g_i(\boldsymbol{x}^{(k)}) = 0$.

At C-active



Status of a constraint at design points A, B, C, and D.

Inactive Constraint

An inequality constraint $g_i(\boldsymbol{x}) \leq 0$ is said to be inactive at a design point $\boldsymbol{x}^{(k)}$ if it has negative value at that point, i.e., $g_i(\boldsymbol{x}^{(k)}) < 0$.

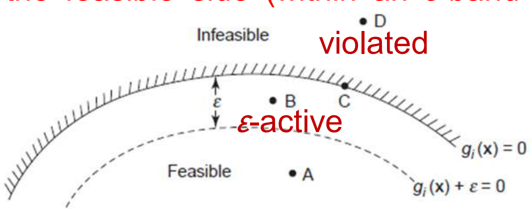
At A-inactive

Violated Constraint

An inequality constraint $g_i(\boldsymbol{x}) \leq 0$ is said to be violated at a designpoint $\boldsymbol{x}^{(k)}$ if it has positive value there, i.e., $g_i(\boldsymbol{x}^{(k)}) > 0$. An equality constraint $h_i(\boldsymbol{x}^{(k)}) = 0$ is violated at a design point $\boldsymbol{x}^{(k)}$ if it has nonzero value there, i.e., $h_i(\boldsymbol{x}^{(k)}) \neq 0$. Note that by these definitions, an equality constraint is always either active or violated for any design point. At D-violated

ε-Active Constraint

Any inequality constraint $g_i(\boldsymbol{x}) \leq 0$ is said to be ε-active at the point $\boldsymbol{x}^{(k)}$ if $g_i(\boldsymbol{x}) < 0$ but $g_i(\boldsymbol{x}^{(k)}) + \varepsilon \geq 0$, where $\varepsilon > 0$ is a small number. This means that the point is close to the constraint boundary on the feasible side (within an ε-band as shown in Fig.).



At B-ε-active

Status of a constraint at design points A, B, C, and D.

10.1.3 Constraint Normalization

In numerical calculations, it is desirable to normalize all the constraint functions. As noted earlier, active and violated constraints are used in computing a desirable direction of design change.

Usually one value for ε (say 0.10) is used for all constraints. Since different constraints involve different orders of magnitude, it is not proper to use the same ε for all the constraints unless they are normalized. For example, consider a stress constraint as

$$\sigma \le \sigma_a \quad \text{or} \quad \sigma - \sigma_a \le 0 \qquad (10.5)$$

and a displacement constraint as

$$\delta \le \delta_a \quad \text{or} \quad \delta - \delta_a \le 0 \qquad (10.6)$$

where

σ = calculated stress at a point

σ_a = an allowable stress

δ = calculated deflection at a point

δ_a = an allowable deflection

9/28

Note that the units for the two constraints are different.

Constraint of Eq. (10.5) involves stress, which has units of Pascals (Pa, N/m^2). For example, allowable stress for steel is 250 MPa.

The other constraint in Eq. (10.6) involves deflections of the structure, which may be only a few centimeters. For example, allowable deflection δ_a may be only 2 cm.

Thus, the values of the two constraints are of widely differing orders of magnitude. If the constraints are violated, it is difficult to judge the severity of their violation. We can, however, normalize the constraints by dividing them by their respective allowable values to obtain the normalized constraint as

$$R - 1.0 < 0 \qquad (10.7)$$

where $R = \sigma/\sigma_a$ for the stress constraint and $R = \delta/\delta_a$ for the deflection constraint. Here, both σ_a and δ_a are assumed to be positive; otherwise, the sense of the inequality changes. For normalized constraints, it is easy to check for ε-active constraint using the same value of ε for both of them.

10/28

There are other constraints that must be written in the form

$$1.0\text{-}R \le 0 \quad (10.8)$$

when normalized with respect to their nominal value.

Example: The fundamental vibration frequency ω of a structure or a structural element must be above a given threshold value of ω_a,

$\omega \geq \omega_a$ $\omega/\omega_a\text{-}1 \geq 0$ $R=\omega/\omega_a$ $1.0\text{-}R \le 0$

In subsequent discussions, it is assumed that all equality as well as inequality constraints have been converted to the normalized form.

EXAMPLE 10.1 Constraint Normalization and Status at a Point

Consider the two constraints:

$$\bar{h} = x_1^2 + \frac{1}{2}x_2 = 18 \quad \text{(a)}$$

$$\bar{g} = 500x_1 - 30000x_2 \le 0 \quad \text{(b)}$$

At the design points (1,1) and (-4.5, -4.5), investigate whether the constraints are active, violated, ε-active, or inactive. Use ε=0.1 to check ε-active constraints.

Solution. Let us normalize the constraint and express it in the standard form as

$$h = \frac{1}{18}x_1^2 + \frac{1}{36}x_2 - 1.0 = 0 \quad \text{(c)}$$

h(1,1)=-0.9166, violated at (1,1)

h(-4.5,-4.5)=0., active at (-4.5,-4.5).

The inequality constraint $\bar{g}$ cannot be normalized by dividing it by $500x_1$ or $30{,}000x_2$ because x_1 and x_2 can have negative values which will change the sense of the inequality. We must normalize the constraint functions using only positive constants or positive variables.

To treat this situation, we may divide the constraint by $30{,}000|x_2|$ and obtain a normalized constraint as $\frac{x_1}{60|x_2|} - \frac{x_2}{|x_2|} \le 0$. This type of normalization is, however, not desirable since it changes the nature of the constraint from linear to nonlinear. Linear constraints are more efficient to treat than the nonlinear constraints in numerical calculations.

Therefore, care and judgment needs to be exercised while normalizing constraints. If a normalization procedure does not work, another procedure should be tried. In some cases, it may be better to use the constraints in their original form, especially the inequality constraints.

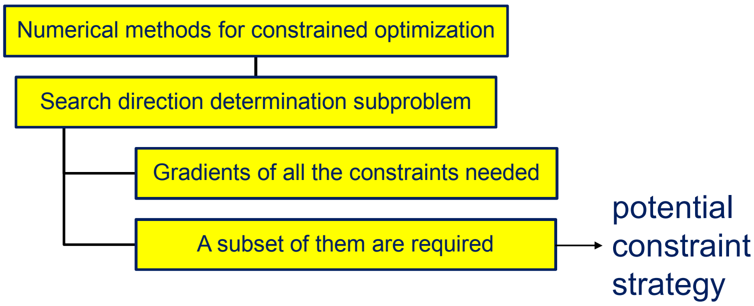
Correct the book

Thus, in numerical calculations, some experimentation with normalization of constraints may be needed for some forms of the constraints. For the present constraint, we normalize it with respect to the constant 500 and then divide by 100 to obtain it in the percent form as

$$g = \frac{1}{100}(x_1 - 60x_2) \le 0 \qquad \text{(d)}$$

At (1,1), g=-0.59<0 (so, inactive)
At (-4.5,-4.5), g=2.655>0 (so, violated).

11.1 Potential Constraint Strategy



To implement this strategy, a potential constraint index set needs to be defined, which is composed of active, *ε*-active, and violated constraints at the current iteration. At the *k*th iteration, we define a potential constraint index set I_k as follows:

$$I_k = [\{j \mid \text{ j=1 to } p \text{ for equalities}\} \text{ and } \{i \mid g_i(x^{(k)}) + \varepsilon \geq 0, \text{i=1 to } m\}] \quad (11.1)$$

(all active, *ε*-active and violated inequalities and all equality constraints) Only inactive constraint are excluded.

Convergence of an Algorithm

An algorithm is said to be convergent if it reaches a minimum point starting from an arbitrary point. An algorithm that has been proven to converge starting from an arbitrary point is called a robust (powerful) method.

In practical applications of optimization, such reliable algorithms are highly desirable. Many engineering design problems require considerable numerical effort to evaluate functions and their gradients. Failure of the algorithm can have disastrous effects with respect to wastage of valuable resources as well as morale (confidence) of designers.

Thus, it is extremely important to develop convergent algorithms for practical applications. It is equally important to enforce convergence in numerical implementation of algorithms in general purpose design optimization software.

A convergent algorithm satisfies the following two requirements:

1. There is a descent function for the algorithm. The idea is that the descent function must decrease at each iteration. This way, progress towards the minimum point can be monitored.
2. The direction of design change $d^{(k)}$ is a continuous function of the design variables.

This is also an important requirement. It implies that a proper direction can be found such that descent toward the minimum point can be maintained. This requirement also avoids "oscillations," or "zigzagging" in the descent function.

10.2 Linearization of Constrained Problem

Let $x^{(k)}$ be the design estimate at the kth iteration and $\Delta x^{(k)}$ be the change in design. Writing Taylor's expansion of the cost and constraint functions about the point $x^{(k)}$, we obtain the linearized subproblem as

$$\min f(x^{(k)} + \Delta x^{(k)}) \cong f(x^{(k)}) + \nabla f^T(x^{(k)})\Delta x^{(k)} \qquad (10\text{-}9)$$

subject to the linearized equality constraints

$$h_j(x^{(k)} + \Delta x^{(k)}) \cong h_j(x^{(k)}) + \nabla h_j^T(x^{(k)})\Delta x^{(k)} = 0; \quad j = 1 \text{ to } p \qquad (10\text{-}10)$$

and the linearized inequality constraints

$$g_j(x^{(k)} + \Delta x^{(k)}) \cong g_j(x^{(k)}) + \nabla g_j^T(x^{(k)})\Delta x^{(k)} \le 0; \quad j = 1 \text{ to } m \qquad (10\text{-}11)$$

Simplified notations for the current design $x^{(k)}$

$$f_k = f(x^{(k)});\ e_j = -h_j(x^{(k)});\ b_j = -g_j(x^{(k)});\ c_i = \partial f(x^{(k)})/\partial x_i;$$

$$n_{ij} = \partial h_j(x^{(k)})/\partial x_i;\ a_{ij} = \partial g_j(x^{(k)})/\partial x_i,\ d_i = \Delta x_i^{(k)} \qquad (10\text{-}12 \text{ to } 18)$$

Note also that the linearization of the problem is done at any design iteration, so the argument $x^{(k)}$ as well as the superscript k indicating the iteration number shall be omitted for some quantities.
Using these notations, the approximate subproblem given in Eqs. (10.9) to (10.11) gets defined as follows:

$$\min \bar{f} = \sum_{i=1}^{n} c_i d_i \qquad (\bar{f} = c^T d) \qquad (10\text{-}19)$$

subject to the linearized equality constraints:

$$\sum_{i=1}^{n} n_{ij} d_i = e_j; \qquad j = 1 \text{ to } p \qquad (N^T d = e) \qquad (10\text{-}20)$$

and the linearized inequality constraints:

$$\sum_{i=1}^{n} a_{ij} d_i \le b_j; \qquad j = 1 \text{ to } m \qquad (A^T d \le b) \qquad (10\text{-}21)$$

Note that since f_k is a constant that does not affect solution of the linearized subproblem, it is dropped from Eq. (10.19). Therefore, $\bar{f}$ represents the linearized change in the original cost function.

EXAMPLE 10.2 Definition of Linearized Subproblem

Consider the optimization problem of Example 4.31,

$$\min f(x) = x_1^2 + x_2^2 - 3x_1x_2 \qquad (a)$$

subject to the constraints

$$g_1 = \frac{1}{6}x_1^2 + \frac{1}{6}x_2^2 - 1 \le 0,\ g_2 = -x_1 \le 0,\ g_3 = -x_2 \le 0 \qquad (b)$$

Linearize the cost and constraint functions about the point $x^{(0)}$=(1,1) and write the approximate problem given by Eqs. (10.19) to (10.21).

Solution.

The graphical solution for the problem is shown in Fig. 10-4.
$(x_1,x_2)=(\sqrt{3},\sqrt{3})$
f= - 3.
The given point (1,1) is inside the feasible region.

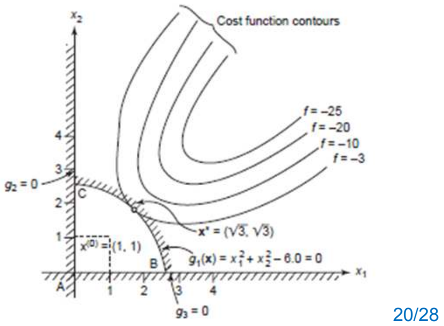


FIGURE 10-4 Graphical representation of the cost and constraints for Example 10.2.

The gradients of cost and constraint functions are

$$\nabla f = (2x_1 - 3x_2, 2x_2 - 3x_1), \quad \nabla g_1 = (\frac{2}{6}x_1, \frac{2}{6}x_2), \quad \nabla g_2 = (-1, 0), \quad \nabla g_3 = (0, -1) \qquad \text{(c)}$$

Evaluating the cost and constraint functions and their gradients at the point (1, 1), we get

$$f(x^{(0)}) = -1.0, \quad b_1 = -g_1(x^{(0)}) = \frac{2}{3}, \quad b_2 = -g_2(x^{(0)}) = 1, \quad b_3 = -g_3(x^{(0)}) = 1, \quad \nabla f(x^{(0)}) = c^{(0)} = (-1, -1), \quad \nabla g_1 = (\frac{1}{3}, \frac{1}{3}) \qquad \text{(d)}$$

Note that the given design point (1,1) is in the feasible set since all the constraints are satisfied. The matrix A and vector b of Eq. (10.21) are defined as

$$A = \begin{bmatrix} 1/3 & -1 & 0 \\ 1/3 & 0 & -1 \end{bmatrix}, \quad b = \begin{bmatrix} 2/3 \\ 1 \\ 1 \end{bmatrix} \qquad \text{(e)}$$

Now the linearized subproblem of Eqs. (10.19) to (10.21) can be written as, minimize

$$\bar{f} = \begin{bmatrix} -1 & -1 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \qquad \text{(f)}$$

subject to

$$\begin{bmatrix} 1/3 & 1/3 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \le \begin{bmatrix} 2/3 \\ 1 \\ 1 \end{bmatrix} \qquad \text{(g)}$$

Or, in the expanded notation, we get $\min \bar{f} = -d_1 - d_2$ subject to

$$\frac{1}{3}d_1 + \frac{1}{3}d_2 \le \frac{2}{3}, \quad -d_1 \le 1, \quad -d_2 \le 1 \qquad \text{(h)}$$

The last two constraints in the subproblem ensure nonnegativity of the design variables required in the problem definition. Note that unless we enforce limits on the design changes d_i, the subproblem may be unbounded.

Note also that the linearized subproblem is in terms of the design changes d_1 and d_2. We may also write the subproblem in terms of the original variables x_1 and x_2. To do this we replace d with $x\text{-}x^{(0)}$ in all the foregoing expressions or in the linear Taylor's expansion and obtain:

$$\bar{f}(x_1,x_2)=f(x^{(0)})+\nabla f(x-x^{(0)})=-1+\begin{bmatrix}-1 & -1\end{bmatrix}\begin{bmatrix}(x_1-1)\\(x_2-1)\end{bmatrix}=-x_1-x_2+1 \quad (i)$$

$$\bar{g}_1(x_1,x_2)=g_1(x^{(0)})+\nabla g_1(x-x^{(0)})=-\frac{2}{3}+\begin{bmatrix}\frac{1}{3} & \frac{1}{3}\end{bmatrix}\begin{bmatrix}(x_1-1)\\(x_2-1)\end{bmatrix}=\frac{1}{3}(x_1+x_2-4)\le 0 \quad (j)$$

$$\bar{g}_2=-x_1\le 0; \qquad \bar{g}_3=-x_2\le 0 \qquad (k)$$

In the foregoing expressions, "overbar" for a function indicates linearized approximation. The feasible regions for the linearized problem at the point (1,1) and the original problem are shown in Fig. 10-5.

Since the linearized cost function is parallel to the linearized first constraint $\bar{g}_1$, the optimum solution for the linearized problem is any point on the line D–E in Fig. 10-5.

$$\bar{f}=-5=-1-x_1-x_2$$

$$4-x_1-x_2=0$$

$$x_1+x_2-4=0\equiv\bar{g}_1$$

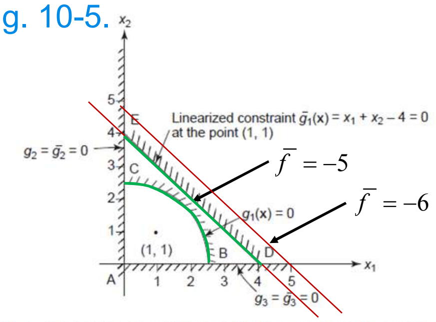


FIGURE 10-5 Graphical representation of the linearized feasible region for Example 10.2.

It is important to note that the linear approximations for the functions of the problem change from point to point. Therefore, the feasible region for the linearized subproblem will change with the point at which the linearization is performed.

EXAMPLE 10.3 Linearization of Rectangular Beam Design Problem

Linearize the rectangular beam design problem formulated in Section 3.8 at the point (50,200)mm.

Solution. The problem, after normalization, is defined as follows: Find width *b* and depth d to minimize *f(b,d)=bd*
subject to

$$g_1 = \frac{(2.40E+07)}{bd^2} - 1 \le 0 \quad \text{(a)}$$

$$g_2 = \frac{(1.125E+05)}{bd} - 1 \le 0 \quad \text{(b)}$$

$$g_3 = \frac{d}{2b} - 1.0 \le 0 \quad \text{(c)}$$

$$g_4 = -b \le 0;\ g_5 = -d \le 0 \quad \text{(d)}$$

At the given point the problem functions are evaluated as

$$\begin{aligned} & f(50,200) = 10{,}000 \\ & g_1(50,200) = 11.00 > 0 \quad \text{(violation)} \\ & g_2(50,200) = 10.25 > 0 \quad \text{(violation)} \\ & g_3(50,200) = 1.00 > 0 \quad \text{(violation)} \\ & g_4(50,200) = -50 < 0 \quad \text{(inactive)} \\ & g_5(50,200) = -200 < 0 \quad \text{(inactive)} \end{aligned} \quad \text{(e)}$$

In the following calculations, we shall ignore constraints g_4 and g_5 assuming that they will remain satisfied, that is, the design will remain in the first quadrant. The gradients of the functions are evaluated as

$$\begin{aligned} & \nabla f(50,200) = (d, b) = (200, 50) \\ & \nabla g_1(50,200) = (2.40E+07)\left(\frac{-1}{b^2 d^2}, \frac{-2}{bd^3}\right) = (-0.24, -0.12) \\ & \nabla g_2(50,200) = (1.125E+05)\left(\frac{-1}{b^2 d}, \frac{-1}{bd^2}\right) = (-0.225, -0.05625) \\ & \nabla g_3(50,200) = \frac{1}{2}\left(\frac{-1}{b^2} d, \frac{1}{b}\right) = (-0.04, 0.01) \end{aligned} \quad \text{(f)}$$

Using the function values and their gradients, the linear Taylor's expansions give the linearized subproblem at the point (50,200) in terms of the original variables as

$$\bar{f}(b,d) = f(50,200) + [200 \quad 50]\begin{bmatrix} b-50 \\ d-200 \end{bmatrix} = 200b + 50d - 10{,}000$$

$$\bar{g}_1(b,d) = -0.24b - 0.12d + 47.00 \le 0 \qquad \text{(g)}$$

$$\bar{g}_2(b,d) = -0.225b - 0.05625d + 32.75 \le 0$$

$$\bar{g}_3(b,d) = -0.04b + 0.01d + 1.00 \le 0$$

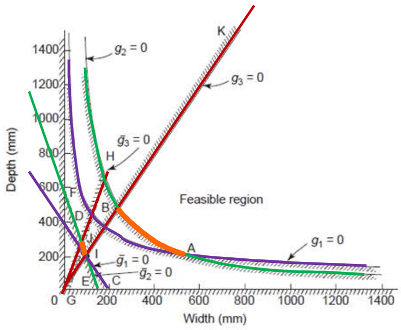


FIGURE 10-6 Feasible region for the original and the linearized constraints of the rectangular beam design problem of Example 10.3.

27/28

The linearized constraint functions are plotted in Fig. 10-6 and their feasible region is identified.
The feasible region for the original constraints is also identified.
It can be observed that the two regions are quite different. Since the linearized cost function is parallel to constraint $\bar{g}_2$, the optimum solution lies on the line *I–J*. If point *I* is selected as the solution for the linearized subproblem, then the new point is given as

b=95.28 mm, *d*=201.10 mm, $\bar{f} = 19{,}111\,\text{mm}^2$ (h)

For any point on line *I–J* all the original constraints are still violated. Apparently, for nonlinear constraints, iterations may be needed to correct constraint violations and reach the feasible set.
One interesting observation concerns the third constraint; the original constraint *d-2b*≤0 is normalized as *d/2b-1*≤0. The normalization does not change the constraint boundary; thus the graphical representation for the problem remains the same, as may be verified in Fig. 10-6.

However, the normalization changes the form of the constraint function that affects its linearization. If the constraint is not normalized, its linearization will give the same functional form as the original constraint for all design points, i.e., $d\text{-}2b\leq 0$. This is shown as line *0–K* in Fig. 10-6. The linearized form of the normalized constraint changes; it gives the line *G–H* for the point (50,200).

This is quite different from the original constraint. The iterative process with and without the normalized constraint can lead to different paths to the optimum point.

In conclusion, we must be careful while normalizing the constraints so as not to change the functional form for the constraints as far as possible.

مسائل زیر را حل کرده و تا دو هفته دیگر تحویل فرمایید:

6) 3, 4

درس تا اول ۶.۴ الگوریتم برنامه ریزی خطی دنباله‌ای