Research Note

Structural optimization by spherical interpolation of objective function and constraints

H. Meshki[a] and A. Joghataie[b,*]

a. *Department of Civil Engineering, Science and Research Branch, Islamic Azad University, Tehran, P.O. Box 14515-775, Iran.*
b. *Department of Civil Engineering, Sharif University of Technology, Tehran, P.O. Box 11155-9313, Iran.*

Abstract. A new method for structural optimization is presented for successive approximation of the objective function and constraints in conjunction with Lagrange multipliers approach. The focus is on presenting the methodology with simple examples. The basis of the iterative algorithm is that after each iteration, it brings the approximate location of the estimated minimum closer to the exact location, gradually. In other words, instead of the linear or parabolic term used in Taylor expansion, which works based on a short step length, an arch is used that has a constant curvature but a longer step length. Using this approximation, the equations of optimization involve the Lagrange multipliers as the only unknown variables. The equations which depend on the design variables are decoupled linearly as these variables are directly obtained. One mathematical example is solved to explain in details how the method works. Next, the method is applied to the optimization of a simple truss structure to explain how the method can be used in structural optimization. The same problems have been solved by penalty method and compared. The results from both methods have been the same. However, because of the long step length and reduction in the number of variables, the speed of convergence has been higher in the presented method.

1. Introduction

In a real structural engineering problem with a large number of variables, the cost of numerical iterations for optimization is generally high. In this paper, Lagrange method, which is a subset of the direct optimization methods, has been used. A problem with direct optimization techniques is the number of constraints, which can be large for applied problems. For example, in Sequential Linear Programming [1], which is a widely used direct optimization method [2-5], the problem is formulated to be solved using simplex

algorithm. This requires working with not only the main constraints, but also additional constraints on the upper and lower limits of the variables, which can increase the total number of constraints, significantly, for problems with a large number of variables [6], or in sequential quadratic programming [7-9], the number of constraints is at least equal to the sum of the number of design variables and the Lagrange multipliers, which can make the procedure of optimization very time-taking. However, in the method presented in this paper, because the constraints and objective function are replaced with their equivalent spheres, the problem is discretized into several independent sub-problems which are solved independently to provide an approximate solution. Through successive iterations, the solution is obtained. This characteristic of the

*. *Corresponding author. Tel.: +98 21 66164216*
  E-mail addresses: h.meshki@sriau.ac.ir (H. Meshki);
  joghatae@sharif.edu (A. Joghataie)

method results in the increase in the speed of convergence.

In methods such as the sequential linear programming [2,10] and the sequential quadratic programming [6,11-13], Taylor expansion is used for interpolation of the functions, while in the method presented here, the functions are approximated by spheres in the n-dimensional space for n-dimensional problems. Hence, the method uses the radii of curvature, which are the radii of the spheres which are tangent to the given n-dimensional functions of constraints and objective function. At the tangent point of each of the spheres to its corresponding function, the curvature is determined based on the gradient of the function at the tangent point. Obviously, the gradient passes through the center of the sphere. This simple approximation makes the convergence become faster. After the spheres are determined, through a simple optimization algorithm, an optimum point for the system of spheres is determined. Next, the optimum point is projected back on the original functions and the procedure is repeated until convergence is achieved.

In the next sections, to provide an easy and better explanation of the method, first, the method and formulation are explained and discussed for 2-dimensional problems, based on which the general formulation and method is presented. Several examples are included to show how the method can be used to solve analytical and numerical problems. The results of numerical problems are compared with the results obtained from exterior penalty method. Since this is the first paper on the method, the main goal has been to present the formulation and its capability in solving a number of test problems. There are steps which should be taken to assess the different features of this method, including its convergence behaviour, and to determine the group of problems where application of this method is advantageous, though a better convergence speed has already been observed in the demonstration examples solved in this paper, as compared with the exterior penalty method.

2. Two-dimensional problems

Though the general mathematical model will be presented in the next section, in this section, for better clarification and simpler explanation of the spherical approximation method, the basic procedure for 2-dimentional optimization problems is presented graphically as shown in Figure 1.

In Figure 1, $X^{(0)}$ is the initial point, which is used to find the point X_g on the active constraint $g = 0$, which is also at the nearest distance from point $X^{(0)}$. To obtain point X_g, a normal is drawn from $X^{(0)}$ to the constraint $g = 0$, where the tangent point is denoted by X_g. Clearly, the vector connecting point
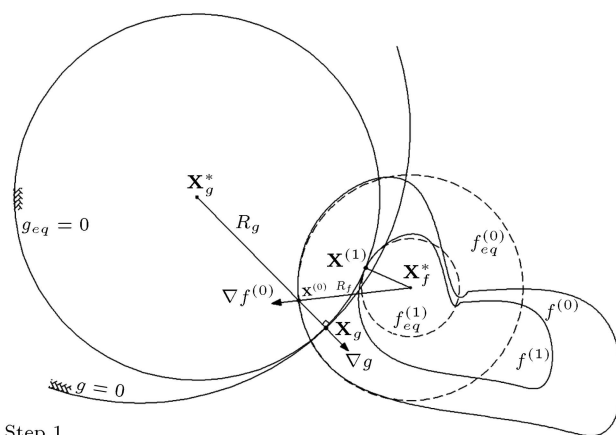
Step 1

Figure 1. Procedure of updating reference point $X^{(0)}$ to $X^{(1)}$.

$X^{(0)}$ to X_g aligns with the gradient of $g = 0$ at point X_g, denoted by Vg. The radius of curvature of the constraint at X_g is denoted by R_g. Now, using $X^{(0)}$, X_g, and R_g, it is possible to determine the center of the circle (sphere) denoted by X^*g, as shown in Figure 1. The original constraint $g = 0$ is now approximated with an equivalent constraint $g_{eq} = 0$, which is circular (spherical). Similarly, using the same point of $X^{(0)}$, it is possible to construct for the objective function, $f(x)$, an equivalent circular (spherical) objective function, which is denoted by $f_{eq}^{(0)}$.

However, the circle is constructed so that $X^{(0)}$ is the tangent point, as shown in Figure 1. Denoting the radius of curvature of the objective function by R_f and its gradient by Vf at point $X^{(0)}$, the center of sphere of the objective function, X_f^*, can be obtained. Having the equivalent constraint, $g_{eq} = 0$, and the sphere center of the objective function, X_f^* can be obtained, which is the intersection point of the line connecting the two centers of the equivalent constraint and objective function. $X^{(1)}$ is now considered to be the starting point for the next iteration. It is noteworthy that $X^{(1)}$ is the tangent point of the two circles (spheres).

3. General multivariable problems

The general form of the optimization problems studied in this paper is:

$$\min f = f(X), \tag{1}$$

s.t.

$$g_i(X) \leq 0 \qquad j = 1, 2, ..., m, \tag{2}$$

$$g_j(X) = 0 \quad j = m+1, 2, ..., p, \tag{3}$$

where m denotes the number of inequality constraints and p is the number of equality constraints.

An n-dimensional spherical interpolation problem can be expressed as follows:

$$\min f = \sum_{i=1}^{n} \left(x_i - x_{if}^* \right)^2 - R_f^2 + f(X^{(k)}), \qquad (4)$$

s.t.

$$g_j(X) = \sum_{i=1}^{n} \left(x_i - x_{ij}^* \right)^2 - R_j^2 \le 0 \qquad j = 1, 2, ..., m, \qquad (5)$$

$$g_j(X) = \sum_{i=1}^{n} \left(x_i - x_{ij}^* \right)^2 - R_j^2 = 0 \quad j = m+1, ..., p, \qquad (6)$$

where k is iteration number; n, number of variables; x_i, the ith design variable; x_{if}^*, the ith component of the center point of the objective function equivalent sphere, x_{ij}^*, the ith component of the center point of the jth constraint equivalent sphere; R_f, radius of the objective function sphere; R_j, radius of the jth constraint sphere; and $f(X^{(k)})$, objective function value in the kth iteration.

3.1. Determining the approximate parameters
3.1.1. Radius of the spheres in the n-dimensional space

The radius of curvature can be determined based on the definition of curvature as the ratio of arch length to the angle of curvature. Since the angle of curvature occurs in a specific plane, first, a plane specified with two vectors in the n-dimensional space should be defined. In other words, for every point on an n-dimensional function surface, a curvature exists for every specific direction. Since the radius of curvature is perpendicular to the surface, it points toward the gradient of the surface, which is in the same direction as that of the radius of curvature. Also, since the reference point moves toward the optimal point in sequential steps, the radius of curvature should be calculated along the vector connecting the two subsequent reference points according to Figure 2, where $\mathbf{V}^{(i)}$ shows the vector. Therefore, the plane of curvature includes both $\mathbf{V}^{(i)}$ and the gradient vector, Vg. Since basically path of arch length is tangent to surface, it is assumed that the vector $\mathbf{w}^{(i)}$, which is tangent to the surface (or perpendicular to Vg) and is in the plane of $\mathbf{V}^{(i)}$ and Vg, is replaced instead of vector $\mathbf{V}^{(i)}$, based on which the radius of curvature is determined in the direction of $\mathbf{w}^{(i)}$. As it will be illustrated, use of $\mathbf{w}^{(i)}$ instead of $\mathbf{V}^{(i)}$ causes simplification of the calculations. First, vector $\mathbf{w}^{(i)}$ is obtained as described below:

In the ith computational step, according to Figure 2, $\mathbf{V}^{(i)}$ is defined as the optimal path vector, which is obtained by connecting the two subsequent reference points $X_{\text{ref}}^{(i)}$ and $X_{\text{ref}}^{(i-1)}$ as:

$$\mathbf{V}^{(i)} = \frac{\mathbf{x}_{\text{ref}}^{(i)} - \mathbf{x}_{\text{ref}}^{(i-1)}}{\left\| \mathbf{x}_{\text{ref}}^{(i)} - \mathbf{x}_{\text{ref}}^{(i-1)} \right\|}. \qquad (7)$$
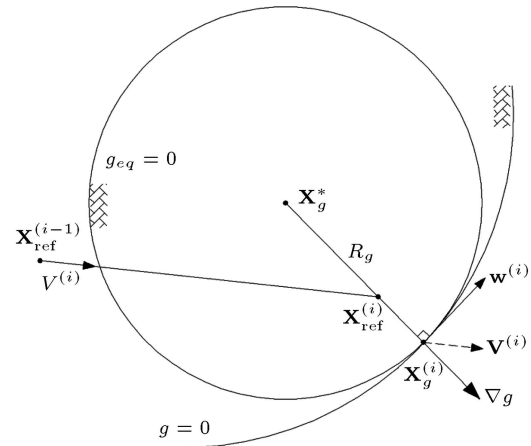

Figure 2. Constraint's radius of curvature, R_g, toward $\mathbf{w}^{(i)}$; vector $\mathbf{w}^{(i)}$ in plane of two vectors $\mathbf{V}^{(i)}$ and Vg, and also $V^{(i)}$ in direction of two sequent reference points, $\mathbf{X}_{\text{ref}}^{(i)}$ rand $\mathbf{X}_{\text{ref}}^{(i-1)}$.

Basically, the $n-1$ vectors, denoted by $\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{n-1}$, which have the property to be orthogonal to $\mathbf{V}$ in the n-dimensional space, [14] are as follows:

$$\mathbf{w}_{n-1} = \mathbf{V} - \frac{\langle \mathbf{V}, \mathbf{w}_1 \rangle}{\langle \mathbf{w}_1, \mathbf{w}_1 \rangle} \mathbf{w}_1 - ... - \frac{\langle \mathbf{V}, \mathbf{w}_{n-2} \rangle}{\langle \mathbf{w}_{n-2}, \mathbf{w}_{n-2} \rangle} \mathbf{w}_{n-2}, \qquad (8)$$

where, sign $\langle \Box \rangle$ is the inner product operator.

However, for the specific vectors $\mathbf{V}$ and Vg, the tangential vector $\mathbf{w}_0$, which is perpendicular to Vg and is also in the plane of the two vectors $\mathbf{V}$ and Vg, is:

$$\mathbf{w}_0 = \mathbf{V} - \frac{\langle \mathbf{V}, \nabla g \rangle}{\langle \nabla g, \nabla g \rangle} \nabla g \qquad (9)$$

$\mathbf{w}_0$ is normalized to provide a unit vector in the same direction as follows:

$$\mathbf{w} = \frac{\mathbf{w}_0}{\| \mathbf{w}_0 \|}. \qquad (10)$$

Based on the previous definition of curvature, the radius of curvature is equal to the ratio of the change in arch length to the change in curvature angle in the direction of $\mathbf{w}$. The unit gradient vector $\widehat{\mathbf{G}}$ normal to the surface is defined as:

$$\widehat{\mathbf{G}} = \frac{\nabla g}{\| \nabla g \|}. \qquad (11)$$

Then, by applying a simple proportioning according to Figure 3:

$$d\phi_{\vec{w}} = \frac{dS_w}{R} = \frac{\| d\widehat{\mathbf{G}}_{\mathbf{w}} \|}{\| \widehat{\mathbf{G}} \|}, \qquad (12)$$

where, $\phi_{\mathbf{w}}$ is curvature angle toward the direction $\mathbf{w}$, and $S_{\mathbf{w}}$ is arch length in the direction of $\mathbf{w}$.

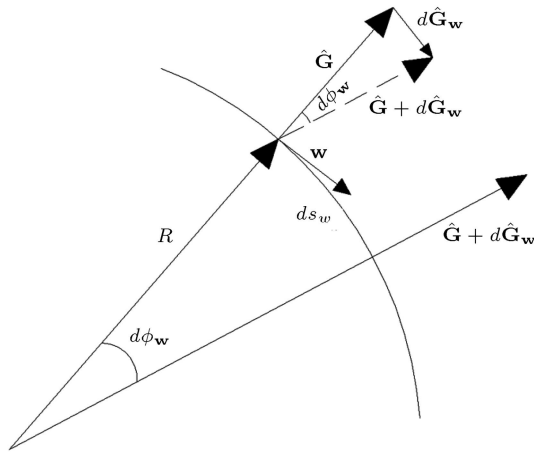Noticing the gradient, vector $\widehat{\mathbf{G}}$ is a unit vector,

Figure 3. Radius of curvature, arch length, unit gradient vector, and its difference.

Eq. (12) can be rewritten as:

$$\frac{1}{R} = \frac{\|d\widehat{\mathbf{G}}.\mathbf{w}\|}{dS_{\mathbf{w}}}.$$ (13)

Based on the definition of the directional derivative:

$$\frac{d\widehat{\mathbf{G}}_{\vec{w}}}{dS_w} = \nabla \widehat{\mathbf{G}}.\mathbf{w}.$$ (14)

Substituting Eq. (14) into Eq. (13):

$$\frac{1}{R} = \|\nabla \widehat{\mathbf{G}}.\mathbf{w}\|.$$ (15)

Hence, according to Eq. (15), the radius of curvature can be determined by the derivative of each component of the unit gradient and, subsequently, the magnitude of the directional derivatives in the direction $\mathbf{w}$ of the components of the unit gradient, as it is explained in the following. Since $\widehat{\mathbf{G}}$ is a vector:

$$\nabla \widehat{\mathbf{G}} = \nabla \left\{ \widehat{\mathbf{G}}_1, \widehat{\mathbf{G}}_2, ..., \widehat{\mathbf{G}}_n \right\},$$ (16)

where:

$$\nabla \left\{ \widehat{\mathbf{G}}_1, \widehat{\mathbf{G}}_2, ..., \widehat{\mathbf{G}}_n \right\}$$

$$= \begin{bmatrix} \partial \widehat{\mathbf{G}}_1/\partial x_1 & \cdots & \partial \widehat{\mathbf{G}}_n/\partial x_1 \\ \vdots & \ddots & \vdots \\ \partial \widehat{\mathbf{G}}_1/\partial x_n & \cdots & \partial \widehat{\mathbf{G}}_n/\partial x_n \end{bmatrix}$$ (17)

where the element ij is:

$$\frac{\partial \widehat{\mathbf{G}}}{\partial x_j} = \frac{\partial}{\partial x_j} \left(\frac{\partial g/\partial x_i}{\|\nabla g\|} \right)$$

$$= \frac{\frac{\partial^2 g}{\partial x_i \partial x_j}\|\nabla g\| - \|\nabla g\|' x_j \partial g/\partial x_i}{\|\nabla g\|^2}.$$ (18)

Since, $\|\nabla g\| = \sqrt{\nabla g^T.\nabla g}$, therefore:

$$\|\nabla g\|' x_j = \frac{\frac{\partial}{\partial x_j}\nabla g^T.\nabla g}{\|\nabla g\|}.$$ (19)

Substituting Eq. (19) in Eq. (18):

$$\frac{\partial \widehat{\mathbf{G}}_i}{\partial x_j} = \|\nabla g\|^{-2} \left(\frac{\partial^2 g}{\partial x_j \partial x_i}\|\nabla g\| - \frac{\partial}{\partial x_j}(\nabla g^T) \right.$$

$$\left. .\nabla g\|\nabla g\|^{-1}\partial g/\partial x_i \right),$$ (20)

where the matrix form of Eq. (20) is as follows:

$$\mathbf{D}_G = \|\nabla g\|^{-2}\mathbf{H}\left(\|\nabla g\|\mathbf{I} - (\nabla g.\nabla g^T)\|\nabla g\|^{-1}\right),$$ (21)

where, $\mathbf{D}_{Gij} = \frac{\partial}{\partial x_j}\left(\frac{\partial g/\partial x_i}{\|\nabla g\|}\right)$, $\mathbf{H}$ = Hessian matrix, and $\mathbf{I}$ = unit matrix. Now, from Eq. (15) the following is obtained:

$$\frac{1}{R} = \|\mathbf{D}_G.\mathbf{w}\|.$$ (22)

Since $\nabla g \perp \mathbf{w}$, then $(\nabla g.\nabla g^T).\mathbf{w} = \mathbf{0}$, hence, using Eqs. (21) and (22), radius of curvature is:

$$R = \frac{\|\nabla g\|}{\|\mathbf{H}.\mathbf{w}\|}.$$ (23)

It should be mentioned here that once the magnitude of the gradient and the Hessian from Eq. (5) are substituted in Eq. (23), the radius of curvature of the approximate function is the fixed value, R_j.

3.1.2. Center of sphere in n-dimensional space
Using the radius of curvature and gradient vector at any point X on a given surface, the center of the equivalent sphere can be determined from:

$$\theta_{ij} = \frac{\frac{\partial g_j}{\partial x_i}}{\|\nabla g_j\|},$$ (24)

where θ_{ij} corresponds to the ith component of the jth constraint. On the other hand, based on the derivative of the spherical form in Eqs. (5) and (6), this parameter is obtained as:

$$\theta_{ij} = \frac{x_{ij} - x_{ij}^*}{R_j}.$$ (25)

Therefore, rewriting Eq. (25) in vector form, the coordinates of the sphere center for the jth constraint are:

$$\mathbf{X}_j^* = \mathbf{X}_j - R_j\theta_j.$$ (26)

Hence, obtaining the radius of curvature and the center of sphere for the constraints and objective function, the modified form of the optimization problem stated in Eqs. (4)-(6) can be constructed.

3.2. Updating the reference point

Following Eqs. (4)-(6), the Lagrange function can be written as follows:

$$\min_x L = f + \sum_{j=1}^{m} \mu_j g_j, \qquad (27)$$

where f is the objective function; g_j, the jth constraint; and μ_j, Lagrange multiplier.

For the inequality constraints $g_j \leq 0$, the solution technique is the active constraint method. At the first step, all the constraints are assumed active where all of them are considered to be the equality constraints. For the solution to be a minimum according to Kuhn-Tucker conditions, the multiplier(s) should be positive. Obtaining the Lagrange multiplier, if the constraint multiplier is negative, the constraint related to the most negative multiplier is eliminated and the calculation is repeated until according to Kuhn-Tucker conditions, there remains non-positive multiplier.

Using Lagrange equation $\frac{\partial L}{\partial x_i} = 0$ for Eq. (27), the following is obtained:

$$\frac{\partial f}{\partial x_i} + \sum_{j=1}^{m} \mu_j \frac{\partial g_j}{\partial x_i} = 0. \qquad (28)$$

Here, substituting the approximate spheres for f and g_j as discussed in Eqs. (4)-(6) and also Eq. (25):

$$\frac{\partial L}{\partial x_i} = 2\left(x_i - x_{if}^*\right) + 2\sum_{j=1}^{m} \mu_j \left(x_i - x_{ij}^*\right) = 0. \qquad (29)$$

The variable x_i is obtained from Eq. (29) as:

$$x_i = \frac{x_{if}^* + \sum_{j=1}^{m}\left(\mu_j x_{ij}^*\right)}{1 + \sum_{j=1}^{m}\left(\mu_j\right)}. \qquad (30)$$

Since the denominator of Eq. (30) is a scalar quantity, the vector of design variable X is obtained as follows:

$$\mathbf{X} = \left(\mathbf{X}_f^* + \mathbf{X}_{n\times m}^*\mu\right) / \left(1 + \mathbf{1}^T\mu\right), \qquad (31)$$

where the components of matrix $\mathbf{X}^*$ and vector $\mathbf{X}_f^*$, respectively, are x_{ij}^* and x_{if}^*. In the next section, the multiplier vector μ in Eq. (31) is obtained.

3.3. Determining Lagrange multipliers for constraints

Using the last Lagrange equation $\frac{\partial L}{\partial \mu_j} = 0$:

$$\sum_{i=1}^{n}\left(x_i - x_{ij}^*\right)^2 = R_j^2, \qquad j = 1, 2, ..., m, \qquad (32)$$

where, m is the number of active constraints: The vector form of Eq. (32) is:

$$\left(\mathbf{X} - \mathbf{X}_j^*\right)^T \left(\mathbf{X} - \mathbf{X}_j^*\right) = R_j^2. \qquad (33)$$

Rewriting Eq. (33) gives:

$$\mathbf{X}^T\mathbf{X} - 2\mathbf{X}^T\mathbf{X}_j^* + \mathbf{X}_j^{*T}\mathbf{X}_j^* = R_j^2. \qquad (34)$$

Substituting Eq. (31) in Eq. (34) gives:

$$\mathbf{X}_f^{*T}\mathbf{X}_f^* + 2\mathbf{X}_f^{*T}\mathbf{X}^*\mu + \mu\mathbf{X}^{*T}\mathbf{X}^*\mu$$
$$- 2\left(\mathbf{X}_f^{*T}\mathbf{X}_j^* + \mathbf{X}_j^{*T}\mathbf{X}^*\mu\right)\left(1 + \mathbf{1}_{m\times 1}^\mathbf{T}\mu\right)$$
$$+ \left(\mathbf{X}_j^{*T}\mathbf{X}_j^* - R_j^2\right)\left(1 + \mathbf{1}_{m\times 1}^\mathbf{T}\mu\right)^2 = 0. \qquad (35)$$

The last parenthesis in Eq. (35) can be written as:

$$\left(1 + \mathbf{1}_{m\times 1}^\mathbf{T}\mu\right)^2 = 1 + \boldsymbol{\mu}^T\mathbf{1}_{m\times m}\boldsymbol{\mu} + 2\mathbf{1}_{m\times 1}^\mathbf{T}\boldsymbol{\mu}. \qquad (36)$$

Therefore, rewriting Eq. (35) and using Eq. (36) the following is obtained:

$$g_j = \boldsymbol{\mu}^T\mathbf{A}_j\boldsymbol{\mu} + \mathbf{B}_j^T\boldsymbol{\mu} + C_j = 0, \qquad j = 1, 2, ..., m, \qquad (37)$$

where:

$$\mathbf{A}_j = \mathbf{X}^{*T}\mathbf{X}^* - 2\left(\mathbf{X}_j^*\mathbf{1}_{m\times 1}^\mathbf{T}\right)^T\mathbf{X}^*$$
$$+ \left(\mathbf{X}_j^{*T}\mathbf{X}_j^* - R_j^2\right)\mathbf{1}_{m\times m}, \qquad (38)$$

$$\mathbf{B}_j^T = 2\left(\mathbf{X}_f^* - \mathbf{X}_j^*\right)^T\left(\mathbf{X}^* - \mathbf{X}_j^*\mathbf{1}_{m\times 1}^\mathbf{T}\right) - 2R_j^2\mathbf{1}_{m\times 1}^\mathbf{T}, \qquad (39)$$

$$C_j = \left(\mathbf{X}_f^* - \mathbf{X}_j^*\right)^T\left(\mathbf{X}_f^* - \mathbf{X}_j^*\right) - R_j^2. \qquad (40)$$

Using the m Eqs. in (37),

$$\mathbf{g} = \mathbf{A}_\mu + \mathbf{B^T}\mu + \mathbf{C} = \mathbf{0}, \qquad (41)$$

where:

$$A_{\mu,j} = \boldsymbol{\mu}^T\mathbf{A}_j\boldsymbol{\mu}. \qquad (42)$$

The Lagrange multipliers can be obtained by one of the numerical methods such as the Jacobean technique for the matrix equation (41).

3.4. Returning the reference point to constraint surface

Using the previous relations, the radius of curvature and center of sphere for the objective function and the constraints can be determined. For the objective function, these calculations are performed at the reference point, but for the constraints, because they are fixed in the n-dimensional space and on the other hand the constraints themselves are not approximated but their curvatures are approximated, first, the reference point is returned to all the active constraints; then, at these points, the approximate parameters are calculated. The criterion for returning to the active constraint is based on the shortest distance of the reference point from each of the active constraints, because the

reference point is the basis for optimization calcula-
tions obtained by Lagrange equations. Hence, every
computational step includes two parts: determination
of the reference point and its returning to the surface
of active constraints. The shortest distance between
the reference point and each of the active constraints is
obtained based on the following optimization problem:

$$\min_x \mathbf{l}_j(X)^2 = \left(\mathbf{X}_j - \mathbf{X}_{\text{ref}}^{(\mathbf{k})}\right)^T \left(\mathbf{X}_j - \mathbf{X}_{\text{ref}}^{(k)}\right). \qquad (43)$$

s.t.

$$g_j(x) = 0, \qquad (44)$$

where, $\mathbf{l}_j(X)$ is direction vector between $\mathbf{X}_j$ and $\mathbf{X}_{\text{ref}}^{(k)}$;
$\mathbf{X}_{\text{ref}}^{(k)}$, coordinate of reference point in the kth iteration;
and $\mathbf{X}_j$, point coordinate on the jth active constraint
and with the shortest distance from the reference point.

To solve the above problem, the following method
is performed. Substituting the Lagrange function as:

$$\phi = \mathbf{l}_j^2 + \mu g_j, \qquad (45)$$

in the Lagrange equation, $\nabla\phi = 0$ can be written as:

$$2\mathbf{l}_j = -\mu \nabla g_j. \qquad (46)$$

Multiplying the left and right sides of Eq. (46) by their
own transposes,

$$4\mathbf{l}_j^2 = \mu^2 \nabla g_j^T \nabla g_j, \text{ and } \mu = \left(4\mathbf{l}_j^2 / \nabla g_j^T \nabla g_j\right) 6^{0.5}, \quad (47)$$

μ is plugged in Eq. (46) to obtain the vector $\mathbf{X}_j$ in
iteration $i + 1$:

$$\mathbf{X}_j^{(i+1)} = -\frac{\|\mathbf{l}_j\|}{\|\nabla g_j\|} \nabla g_j \bigg|_{\mathbf{X}_j^{(i)}} + \mathbf{X}_{\text{ref}}^{(k)}. \qquad (48)$$

In Eqs. (44) and (48), For $k = 1$, where the reference
point is $X_{\text{ref}}^{(0)}$, the initial point $X_j^{(0)}$ is selected as:

$$\mathbf{X}_j^{(0)} = \mathbf{X}_{\text{ref}}^{(0)} + \Delta \mathbf{X}, \qquad (49)$$

where $\Delta \mathbf{X}$ is arbitrary vector with small values.

For the next iterations, when $k > 1$, the point
obtained on the jth constraint at iteration $k - 1$ can
be chosen as the initial point $\mathbf{X}_j^{(0)}$. Now, according to
Eq. (43), the direction vector $\mathbf{l}_j^{(1)}(X)$, which according
to Eq. (46) or (48) has the same line of action of Δg_j,
is determined from $\mathbf{l}_j^{(1)}(X) = \mathbf{X}_j^{(0)} - \mathbf{X}_{\text{ref}}^{(k)}$. Also $\mathbf{X}_j^{(1)}$ is
on the same line; so, $\mathbf{X}_j^{(1)} = \alpha \mathbf{l}_j^{(1)}(X) + \mathbf{X}_j^{(0)}$, which is
inserted in Eq. (51) for $g_j\left(\mathbf{X}_j^{(1)}\right) = 0$ to obtain α and,
consequently, $\mathbf{X}_j^{(1)}$. At the next step, $\mathbf{X}_j^{(1)}$ is used in
Eq. (55) to determine if $\mathbf{X}_j^{(2)}$ is obtained.

This procedure is repeated until the exact solution
is obtained, so that $\mathbf{X}_j^{(i+1)}$ in iteration $i+1$ is placed on
the jth active constraint and at the shortest distance
from the reference point $\mathbf{X}_{\text{ref}}^{(k)}$.

4. Spherical optimization algorithm

Based on the equations explained in the previous
sections, an algorithm is developed, which is explained
in this section. Also, two examples are solved by
this algorithm, which is reported in the next section.
Briefly, the algorithm is:

(a) Select an initial feasible reference point, denoted
by $X^{(0)}$;

(b) For each of the constraints, return the reference
point to the surface of the constraint, following
Eqs. (44) and (48) and according to Sections 3
and 4;

(c) Obtain the approximate objective function at the
reference point and also the approximate con-
straints at the points obtained at step (b) using
Eqs. (23) and (26) according to Section 3.1 of this
paper, explained before;

(d) Construct and solve the system of Lagrange mul-
tiplier equations according to Eqs. (38)-(41) and
Section 3.3 and also using the parameters of the
spherical functions obtained at step (c);

(e) Eliminate the most negative Lagrange multiplier
and its constraint from the system of Lagrange
multiplier equations from step (d) by eliminating
the rows and columns corresponding to coefficients
A, B, and C related to the most negative Lagrange
multipliers represented in Eqs. (38)-(40). Repeat
the previous steps until all the remaining Lagrange
multipliers are non-negative;

(f) Determine the reference point using Eq. (31);

(g) If either the reference point is at the intersection
of all the active original constraints or convergence
has been achieved according to the following cri-
teria for monitoring the change in the location of
two successive reference points as:

$$e = \|\mathbf{X}_{\text{ref}}^{(i)} - \mathbf{X}_{\text{ref}}^{(i-1)}\| \leq \in, \qquad (50)$$

then terminate; where, $\in$ is small value, $\mathbf{X}_{\text{ref}}^{(i)}$ is
reference point at the ith step;

(h) Otherwise select the latter reference point as the
new initial point and go to step (b) and repeat the
previous steps.

5. Example problems

Two examples, which cover simple to complicated
cases, are solved by the algorithm and the results are
compared with widely used optimization methods, in-
cluding the penalty method and the active-set method,
according to Matlab software.

5.1. Example-1: Objective function with 3 variables subject to 2 constraints

This example has 3 variables subject to 2 constraints. The problem is:

$$\min_x F = 2x_1 + 5x_2^2 - 7x_3^2, \tag{51}$$

s.t.

$$g_1 = x_3 - 5\frac{x_1}{x_2} - 1 \le 0, \tag{52}$$

$$g_2 = 2x_1^2 - 3x_2 + x_3^3 - 10 \le 0. \tag{53}$$

First, using a commercial optimization software and employing another optimization method and starting from the initial point $X^{(0)} = (2, 2, 2)$, the problem was solved and the optimal point was obtained as $X_{\mathrm{opt}} = (0.1348, 0.5343, 2.2615)$, where $F_{\mathrm{opt}} = -34.1045$. Subsequently, solving the problem by the spherical optimization method, the result of the problem was obtained as in Table 1.

Using Eqs. (44) and (48), the points on the 2 constraints, with the shortest distance from reference point have been obtained. Next, using Eqs. (9) and (10), the tangential vector, $\mathbf{w}$, related to the 2 successive reference points for the objective function and the 2 constraints have been obtained. Then, using Eqs. (23) and (26), the radius of curvature and center of sphere for the objective function and the 2 constraints have been calculated.

As can be seen in Table 1, the distances of both constraints from the reference point have decreased step by step, monotonically. Both of the Lagrange multipliers are positive at all the steps. This shows that both constraints have been active in all the solution steps. The fast variation of error values in Table 1 illustrates a good convergence in this method.
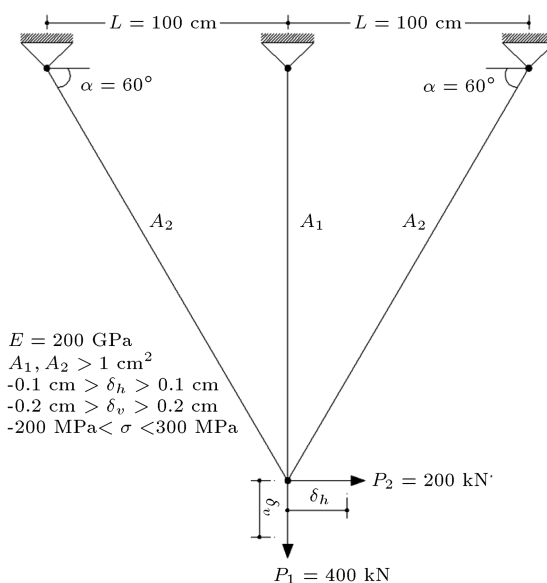

Figure 4. Truss optimization problem where the cross-sectional areas should be determined for minimum weight, subject to minimum area of sections, displacement, and stress constraints.

5.2. Example 2: Structural example

This is a numerical structural optimization problem. The structure is the steel truss shown in Figure 4, where the elasticity modulus is $E = 200$ GPa.

The problem is formulated as:

$$\min_{A_1, A_2} f = hA_1 + 2dA_2, \tag{54}$$

where $h = L \tan \alpha$ and $d = L/\cos \alpha$. The optimization is subjected to:

$$\sigma_i \le \begin{cases} \sigma_t & \sigma_i \ge 0 \\ \sigma_c & \sigma_i < 0 \end{cases} \quad i = 1, 2, \text{ and } 3 \tag{55}$$

$$|\delta_h| \le \delta_{h0}, \tag{56}$$

Table 1. Numerical results until the 4th computational steps.

Steps		Step 1	Step 2	Step 3	Step 4
Distance of reference	$\|l_1\|$	1.4275	0.0918	0.0308	0.0239
Point from 2 constraints	$\|l_2\|$	0.0000	0.2875	0.0265	0.0029
Lagrange multipliers	μ_1	15.0047	1.5893	2.2459	1.8649
	μ_2	15.0047	6.6928	8.6159	9.4487
Reference point	x_1	0.2802	0.1089	0.1453	0.1326
	x_2	0.8979	0.3197	0.4814	0.5182
	x_3	2.0304	2.1928	2.2478	2.2583
Objective function		24.2649	32.9307	33.9192	34.0916
Error from Eq. (50)		2.0428	0.6245	0.1746	0.0403

$$|\delta_v| \leq \delta_{v0}, \tag{57}$$

$$A_1 \geq A_0, \tag{58}$$

$$A_2 \geq A_0. \tag{59}$$

The boundary values in this optimization problem are: $\sigma_t = 300$ MPa, $\sigma_c = -200$ MPa, $\delta_{h0} = 1$ cm, $\delta_{v0} = 2$ cm and $A_0 = 1$ cm^2.

5.2.1. Solution steps
Determination of approximate spherical interpolation parameters.
The initial cross-section, which is the reference point, is selected arbitrarily as: $A^{(0)} = (70, 16)$. Similar to Example 1, the reference point has been projected on the 7 constraints and for the obtained points, the gradient vector, tangential vector, radius of curvature, and centers of spheres of the constraints have been determined. For the linear constraints and the objective functions, the radius of curvature is selected equal to 10^7, which is large enough to be considered infinite.

Elimination of parallel constraints.
To simplify calculations, some of the constraints are eliminated here. Since $x_2 > 1$, based on the seventh constraint, the first, fourth, and sixth linear constraints are satisfied by the fifth constraint according to Figure 5. Therefore, these constraints are never active in this problem and can be eliminated. Hence, only the second, third, fifth, and seventh constraints need to be considered in the optimization.

Determination of coefficients (A, B, and C).
Now that the radii and the center of the spherical objective function and also the second, third, fifth, and seventh constraints are determined, the coefficients A, B, and C related to the Lagrange equations corresponding to these constraints have been calculated.
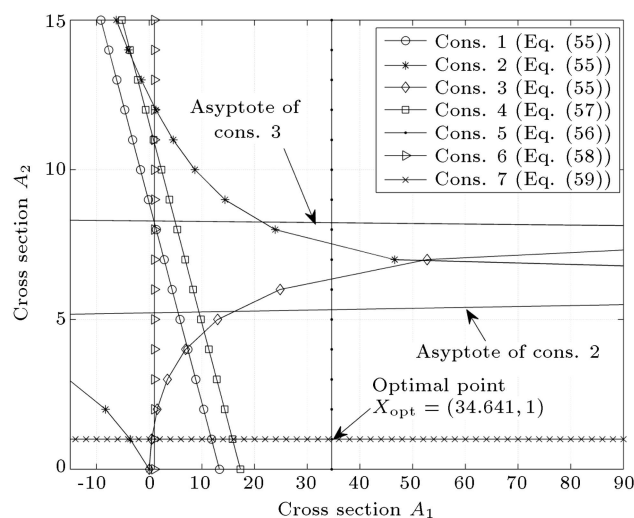

Figure 5. Diagram of the 7 constraints with respect to A_1 and A_2 (cm^2).

Solution of Lagrange equations.
Using the above determined coefficients in conjunction with Eq. (41) and solving by Newton nonlinear method, the Lagrange multipliers are determined as $\boldsymbol{\mu} = (-0.9689, -1.3375, 0.3620, 0.9470)$. The second Lagrange multiplier is the most negative; so, according to Kuhn-Tucker condition, its Lagrange equation should be eliminated. Therefore, matrix $\mathbf{A}_3$ and also the second row and column of matrices $\mathbf{A}_2$, $\mathbf{A}_5$, $\mathbf{A}_7$, $\mathbf{B}$, and $\mathbf{C}$ are eliminated. Inserting these modified coefficients in Eq. (41) and solving them, the Lagrange multipliers are $\boldsymbol{\mu} = (-1.3011, 0.3529, 0.9394)$. Since the first multiplier is negative, again matrix $\mathbf{A}_2$ and also the first row and column of matrices $\mathbf{A}_5$, $\mathbf{A}_7$, $\mathbf{B}$, and $\mathbf{C}$ are eliminated. Now, using Eq. (41) for the two constraints and its solution, the Lagrange multipliers are obtained as $\boldsymbol{\mu} = (0.3443, 0.9343)$. The multipliers are positive; so, according to the Kuhn-Tucker condition, this solution is a minimum.

Determination of reference point.
Since all the remaining constraints as well as the objective function are linear, the radii of curvature of the functions are selected as $R = 10^4$, which are large enough to replace infinity. From Eq. (26), the new centers of spheres have been calculated.

Although the coefficients A, B, and C and the Lagrange multipliers in Eqs. (38)-(41) depend on the location of centers of spheres, selecting a large enough value for all the radii of curvature provides the required precision. Thus, using Eq. (31), the positive Lagrange multipliers obtained at the previous step, and the centers of spheres, the vector of design variables (reference point) is obtained as $\mathbf{A}^{(1)} = (34.6906, 1.0242)$.

Determination of error and termination.
This reference point differs from the initial reference point, significantly, noticing:

$$\mathbf{e} = \|\mathbf{A}^{(1)} - \mathbf{A}^{(0)}\| = \left\| \begin{Bmatrix} 34.6906 \\ 1.0242 \end{Bmatrix} - \begin{Bmatrix} 70 \\ 16 \end{Bmatrix} \right\| = 38.354$$

$$> \varepsilon = 10^{-3}. \tag{60}$$

So, the above steps should be repeated until convergence is achieved. Thus, the calculations are repeated.

The previous reference point was obtained as $\mathbf{A}^{(1)} = (34.6906, 1.0242)$ and also the points on the fifth and seventh constraints have been again obtained, which are equal to $\mathbf{A}_5^{(1)} = (34.6410, 1.0242)$ and $\mathbf{A}_7^{(1)} = (34.6906, 1.0000)$, respectively. Therefore, according to step (g) of the algorithm, when the points on the active constraints have approached the reference point, the solution is reached. Thus, the optimal point is $\mathbf{A}_{opt} = (34.6906, 1.0242)$, while according to Figure 5, the exact solution is $\mathbf{A}_{exact} = (34.6410, 1.0000)$, very close to the solution.

6. Summary and conclusions

A spherical interpolation is presented for the objective function and the constraints using Lagrange multipliers approach and also an algorithm are presented for successive iterations to improve the solution towards the exact answer. The following remarks and conclusions are pertinent with regard to the formulation and the results presented in the paper:

1. The mathematical and structural examples solved by the method, presented here, have also been solved by the exterior penalty method, where both methods have provided exactly the same optimum solutions;

2. Though the time of computation of converge to the final solution is of great importance and should be discussed in detail, the space limitation does not let a proper comparison of convergence behaviour between the presented method and the exterior penalty method. Hence, this issue has been postponed to a follow-up paper, but, just qualitatively, the presented method has shown to converge faster;

3. The proposed method does not depend to the convexity or the concavity of the constraints or the objective function, because the radius of sphere, which indicates the curvature, is directly utilized at each computational step;

4. It should be noticed that there is a possibility that in an iteration, a number of the equivalent spheres of constraints would not intersect with the other constraints, because their radii may be too large to cross the surface of the small spheres. In such situations, obviously, such constraints can be eliminated from the iteration, because they are already satisfied and cannot become active in that iteration. On the other hand, basically, by increasing the radius of curvature of the approximate spherical functions, common space between spheres can be created. Even, increase in radius of curvature up to infinity cannot obstruct convergence of the solution, because the sphere with infinite radius is a plane surface which has convergence similar to the first order of Taylor expansion;

5. Similar to the other optimization methods, the convergence behaviour and success of the proposed method depend on the starting point.

References

1. Etman, L.F.P., Groenwold, A.A. and Rooda, J.E. "First-order sequential convex programming using approximate diagonal QP subproblems", *Structural and Multidisciplinary Optimization*, **45**, pp. 479-488 (2012).

2. Cheney, E.W. and Goldstein, A.A. "Newton's method of convex programming and Tchebycheff approximation", *Numerische Mathematik*, **1**, pp. 253-268 (1959).

3. Kolda, T.G., Lewis, R.M. and Torczon, V. "Optimization by direct search: New perspectives on some classical and modern methods", *Journal of SIAM*, **45**(3), pp. 385-482 (2003).

4. Izmailov, A.F. and Kurennoy, A.S. "Abstract Newtonian frameworks and their applications", *SIAM J. Optim.*, **23**(4), pp. 2369-2396 (2013).

5. Li, C. and Ng, K.F. "Approximate solutions for abstract inequality systems", *SIAM J. Optim.*, **23**(2), pp. 1237-1256 (2013).

6. Wolf, P. "The simplex method for quadratic programming", *Econometrica*, **27**(3), pp. 382-398 (1959).

7. Curtis, F.E., Johnson, T.C., Robinson, D.P. and Wächter, A. "An inexact sequential quadratic optimization algorithm for nonlinear optimization", *SIAM J. Optim.*, **24**(3), pp. 1041-1074 (2014).

8. Park, S., Jeong, S.H., Yoon, G.H., Groenwold, A.A. and Choi, D. "A globally convergent sequential convex programming using an enhanced two-point diagonal quadratic approximation for structural optimization", *Structural and Multidisciplinary Optimization*, **50**, pp. 739-753 (2014).

9. Groenwold1, A.A. and Etman, L.F.P. "A quadratic approximation for structural topology optimization", *International Journal for Numerical Methods in Engineering*, **82**(4), pp. 505-524 (2010).

10. Mahini, M.R., Moharrami, H. and Cocchetti, G. "Elastoplastic analysis of frames composed of softening materials by restricted basis linear programming", *Computers & Structures*, **131**, pp. 98-108 (2014).

11. Groenwold, A.A., Wood, D.W., Etman, L.F.P. and Tosserams, S. "Globally convergent optimization algorithm using conservative convex separable diagonal quadratic approximations", *AIAA Journal*, **47**(11), pp. 2649-2657 (2009).

12. Groenwold, A.A. "Positive definite separable quadratic programs for non-convex problems", *Structural and Multidisciplinary Optimization*, **46**(6), pp. 795-802 (2012).

13. Burke, J.V., Curtis, F.E. and Wang, H. "A sequential quadratic optimization algorithm with rapid infeasibility detection", *SIAM J. Optim.*, **24**(3), pp. 1041-1074 (2014).

14. Edwards Jr., C.H., *Advanced Calculus of Several Variables*, pp. 15-16, Academic Press, New York and London (1973).

Biographies

Hossein Meshki is presently a PhD candidate in the Department of Civil Engineering, Science and Research branch, Islamic Azad University, Tehran, Iran. He received his BS in Civil Engineering from

Islamic Azad University of Kashan, Iran, in 1999 and MS in Earthquake Engineering from the International Institute of Earthquake Engineering and Seismology of Iran in 2002. His is also a lecturer at Islamic Azad University, Tehran, Iran and his research interests are in numerical methods, structural optimization, seismic damage identification, and retrofitting of historical buildings, as well as electrical equipment and seismic performance of light weight steel structures.

Abdolreza Joghataie is a faculty member in Civil Engineering Department at Sharif University of Technology. His research interests include structural health monitoring and optimization, numerical methods, and artificial neural networks.