International Journal of Enhanced Research in Science Technology & Engineering, ISSN: 2319-7463
Vol. 3 Issue 4, April-2014, pp: (44-59), Impact Factor: 1.252, Available online at: www.erpublications.com

Generalized Variable Penalty Function Method for Constraint Optimization

Dr. EmanTarik Hamed[1], Eilaf Sulaiman Khaleel[2]
[1]Department of Operations Research and Intelligent Techniques, College of Computer Sciences and Mathematics, University of Mosul, Mosul, Iraq
[2]Department of Mathematics, College of Computer Sciences and Mathematics. University of Mosul, Mosul, Iraq

Abstract: In this paper we have inveterate new algorithm cambium the best features of the interior and exterior method for inequality constraint. The new algorithm is very effect which over come ill-condition of round error. Our new modified too effective when compared with other established algorithms to solve standard constrained optimization problems.

Keywords: constrained, optimization, the interior and exterior method.

Introduction

Consider the following nonlinear programming

$$\begin{aligned} &Minimize && f(x) && && (1)\\ &subject\ to && g_i(x) \le 0 && for\ \ i = 1,\ldots\ldots\ldots\ldots, m && (2)\\ & && h_i(x) = 0 && for\ \ i = 1,\ldots\ldots\ldots\ldots, l && (3)\\ & && x \in X, && && \end{aligned}$$

Where f, $g_1,\ldots\ldots\ldots\ldots g_m$, $h_1,\ldots\ldots\ldots\ldots,h_l$ are functions defined on R^n, X is a subset of R^n, and x is vector of n components $x_1,\ldots\ldots\ldots\ldots, x_n$ the above problem must be solved for the value of the variable $x_1,\ldots\ldots\ldots\ldots, x_n$ that satisfy the restriction and mean while minimize the function f.

The function f is usually called the objective function , or the criterion function each of the constraints $g_i(x) \le 0 \quad for\ \ i = 1,\ldots\ldots\ldots\ldots, m$ is called an inequality constraint, and each of the constraints $h_i(x) = 0 \quad for\ \ i = 1,\ldots\ldots\ldots\ldots, l$ is called an equality constraint. the set X might typically include lower and upper bounds on the variable, which even if implied by the other constraints can play a useful role in some algorithms, alternatively , this set might represent some specially structured constraints that are highlighted to be exploited by the optimization routine or it might represent certain regional containment or other complicating constraints that are to be handled separately via a special mechanism . A vector $x \in$ X ,satisfying all the constraint is called a feasible solution to problem the collection of all such solutions forms the feasible region , the nonlinear programming problem , then is to find a feasible point $\bar{x}$ such $f(x) \ge f(\bar{x})$ for each feasible point x . Such appoint $\bar{x}$ is called an optimal solution , or simply a solution , to the problem .if more then one optimum exists , they are referred to collectively as alternative optimal solutions.[2]

The constrained minimization problem (1-3) may be solved by the sequence of unconstrained minimization technique (SUMT) [4],

Some important methods for constrained optimization replace the original problem by a sequence of sub problems in which the constraints are represented by terms added to the objective. In this paper we describe three approaches of this type. The quadratic penalty method adds a multiple of the square of the violation of each constraint to the objective. because of its simplicity and intuitive appeal, this approach is used often in practice, although it has some important disadvantages. In non smooth exact penalty methods, a single unconstrained problem (rather than a sequence) takes the place of the original constrained problem. Using these penalty functions, we can often find a solution by performing a single unconstrained minimization, but the non smoothness may create complications. A popular function of this type is the penalty function. A different kind of exact penalty approach is the method of multipliers or augmented Lagrangian method, in which explicit Lagrange multiplier estimates are used to avoid the ill-conditioning that is inherent in the quadratic penalty function. A related approach is used in the log-barrier method, in which logarithmic terms prevent feasible iterates from moving too close to the boundary of the feasible region [7].

2. The Penalty Function Method

Penalty function methods are developed to eliminate some or all of the constraints and add to the objective function a penalty term which prescribes a high cost to infeasible points. In theory, penalty function method uses unconstraint optimization methods to solve constraints optimization problems. Discrete iterative setup can be started with infeasible or feasible starting point and guide system to feasibility and ultimately obtained optimal solution. Penalty function methods transform the basic optimization problem into alternative formulations such that numerical solutions are sought by solving a sequence of unconstrained minimization problems. Let the basic optimization problem, with inequality constraints, be of the form:

Find x which minimizes $f(x)$

subject to

$$g_j(x) \le 0, \qquad j = 1, 2, \ldots\ldots\ldots, m \qquad \ldots.\ (4)$$

this problem is converted into an unconstrained minimization problem by constructing a function of the form

$$\phi_k = \phi(x, r_k) = f(x) + p(x) \qquad \ldots\ldots\ (5)$$

the significance of the second term on the right side of (5),called the penalty term, If the unconstrained minimization of the ϕ function is repeated for a sequence of values of the penalty parameter $r_k\ (k = 1, 2, \ldots\ldots)$ the solution may be brought to converge to that of the original problem stated in (4) . This is the reason why the penalty function methods are also known as sequential unconstrained minimization techniques (SUMT) [9].

3. The Classify of the Penalty Function

The penalty function formulations for inequality constrained problems can be divided into categories: exterior and interior methods.

3.1. Exterior Penalty Function

The exterior penalty is the easiest to incorporate into the optimization process. Here the penalty function p(x) is typically given by

$$p(x) = r_p \sum_{j=1}^{m} \{MAX[0, g_j(x)]\}^2 + r_p \sum_{k=1}^{l} [h_k(x)]^2 \qquad \ldots\ldots\ (6)$$

the subscript p is the outer loop counter which we will call the cycle number. we begin with a small value of the penalty parameter, r_p and minimize the pseudo-objective function ϕ_k . We then increase r_p and repeat the process until convergence.

Lemma (1) (penalty Lemma)

1- $\phi(x_k, r_k) \le \phi(x_{k+1}, r_{k+1})$

2- $p(x_k) \ge p(x_{k+1})$

3- $f(x_k) \le f(x_{k+1})$

4- $f(x^*) \ge \phi(x_k, r_k) \ge f(x_k)$

for more detail see [5] .

3.2. Interior Penalty Function

The basic concept of the interior penalty is that, as the constraints approach zero (from the negative side), the penalty grows to drive the design away from the constraint bounds [4].

a. Reciprocal

In the past, a common penalty function used for the interior method was defined by :

$$p(x)=\sum_{j=1}^{m}\frac{-1}{g_j(x)} \quad \text{.........} \ (7)$$

Using (6) and including equality constraints via the exterior penalty function of (7)

$$\varphi(x,r'_p,r_p)=f(x)+r'_p\sum_{j=1}^{m}\frac{-1}{g_j(x)}+r_p\sum_{k=1}^{l}[h_k(x)]^2 \quad \text{.........} \ (8)$$

here r'_p is initially a large number, and is decreased as the optimization progresses. The last term on (8) is the exterior penalty as before, because we wish to drive $h_k(x)$ to zero. Also, r_p has the same definition as before and $f(x)$ is the original objective function. In our remaining discussion, we will omit the equality constraints, remembering that they are normally added as an exterior penalty function as in (8) [10].

b. Log Barrier Method

An alternative form of (8) is

$$p(x)=r'_p\sum_{j=1}^{m}-\log[-g_j(x)] \quad \text{.........} \ (9)$$

and this is often recommended as being slightly better numerically conditioned.

c. Polyak's Log Barrier Method

Polyak [10] suggests a modified log-barrier function which has features of both the extended penalty function and the Augmented Lagrange multiplier method. The modified logarithmic barrier penalty function is defined as:

$$M(x,r'_p,\lambda^p)=-r'_p\sum_{j=1}^{m}\lambda_j^p\log\left[1-\frac{g_j(x)}{r'_p}\right] \quad \text{......} \ (10)$$

where the nomenclature has been changed to be consistent with the present discussion. Using this, we create the pseudo-objective function

$$\varphi(x,r'_p,\lambda^p)=f(x)-r'_p\sum_{j=1}^{m}\lambda_j^p\log\left[1-\frac{g_j(x)}{r'_p}\right] \quad \text{.........} \ (11)$$

We only consider inequality constraints and ignore side constraints. equality constraints can be treated using the exterior penalty function approach and side constraints can be considered directly in the optimization problem. Alternatively, equality constraints can be treated as two equal and opposite inequality constraints because this method acts much like an extended penalty function method, allowing for constraint violations.

d. Polyak's Log-Sigmoid Method

A more recent method by Polyak, [10] appears to have better properties than the log-barrier method by eliminating the barrier. Here, we create the penalty function as;

$$p(x)=\frac{2}{r_p}\sum_{j=1}^{m}\lambda_j^p\{\ln[1+e^{r_p g_j(x)}]-\ln 2\} \quad \text{.........} \ (12)$$

In the interior methods, the unconstrained minima of ϕ_k all lie in the feasible region and converge to the solution of (4) as r_k is varied in a particular manner. In the exterior methods, the unconstrained minima of ϕ_k all lie in the infeasible region and converge to the desired solution from the outside as r_k is changed in a specified manner.

International Journal of Enhanced Research in Science Technology & Engineering, ISSN: 2319-7463
Vol. 3 Issue 4, April-2014, pp: (44-59), Impact Factor: 1.252, Available online at: www.erpublications.com

Lemma: (2) (Barrier Lemma)

1- $\phi(x_k, r_k) \geq \phi(x_{k+1}, r_{k+1})$

2- $p(x_k) \leq p(x_{k+1})$

3- $f(x_k) \geq f(x_{k+1})$

4- $f(x^*) \leq f(x_k) \leq \phi(x_k, r_k)$

for more detail see [5] .

3.3. Extended Interior Penalty Function

This approach attempts to incorporate the best features of the reciprocal interior and exterior methods for inequality constraints. For equality constraints, the exterior penalty is used as before and so is omitted here for brevity.

a. The Linear Extended Penalty Function

The first application of extended penalty functions in engineering design is attributable to Kavlie and Moe [10]. This concept was revised and improved upon by Cassis and Schmit [10]. Here the penalty function used in (6) takes the form

$$p(x) = \sum_{j=1}^{m} \tilde{g}_j(x) \qquad \text{......... (13)}$$

where

$$\tilde{g}_j(x) = -\frac{1}{g_j(x)} \qquad if \;\; g_j(x) \leq \varepsilon \qquad \text{...... (14)}$$

$$\tilde{g}_j(x) = -\frac{2\varepsilon - g_j(x)}{\varepsilon^2} \qquad if \;\; g_j(x) < \varepsilon \qquad \text{....... (15)}$$

The parameter ε is a small negative number

b. The Quadratic Extended Penalty Function

The extended penalty given by Equations (13) - (15) is defined as the linear extended penalty function. This function is continuous and has continuous first derivatives at $g_j(x) = \varepsilon$ However, the second derivative is discontinuous, and so if a second-order method is used for unconstrained minimization, some numerical problems may result. Haftka and Starnes [6] overcome this by creating a quadratic extended penalty function as

$$\tilde{g}_j(x) = -\frac{1}{g_j(x)} \qquad if \;\; (g_j(x) \leq \varepsilon) \qquad \text{........ (16)}$$

$$\tilde{g}_j(x) = \frac{-1}{\varepsilon}\left\{\left[\frac{g_j}{\varepsilon}\right]^2 - 3\left[\frac{g_j}{\varepsilon}\right] + 3\right\} \quad if \;\; (g_j > \varepsilon) \text{ .... (17)}$$

equations (16) and (17) may be most useful if a second-order method is used for unconstrained minimization. However, the price paid for this second order continuity is that the degree of nonlinearity of the pseudo-objective function is increased[10].

c. General Form Of The Constrained When $g_j(x) > \varepsilon$

if n is odd then we have

$$\frac{1}{\varepsilon}\left\{\left[\frac{g_j(x)}{\varepsilon}\right]^n - \genfrac{}{}{0pt}{}{n+1}{1}\left[\frac{g_j(x)}{\varepsilon}\right]^{n-1} + \genfrac{}{}{0pt}{}{n+1}{2}\left[\frac{g_j}{\varepsilon}\right]^{n-2} + \ldots + \genfrac{}{}{0pt}{}{n+1}{n-1}\left[\frac{g_j}{\varepsilon}\right] - \genfrac{}{}{0pt}{}{n+1}{n}\right\}$$

…… (18)

When n even we have

$$\frac{-1}{\varepsilon}\left\{\left[\frac{g_j(x)}{\varepsilon}\right]^n - {}^{n+1}_{\;\;1}\left[\frac{g_j(x)}{\varepsilon}\right]^{n-1} + {}^{n+1}_{\;\;2}\left[\frac{g_j}{\varepsilon}\right]^{n-2} + \ldots + {}^{n+1}_{n-1}\left[\frac{g_j}{\varepsilon}\right] - {}^{n+1}_{\;\;n}\right\} \qquad \ldots\ldots(19)$$

For more detail see [1]

d. The Variable Penalty Function

Both the exterior and interior penalty function methods have been used with success. The significant modifications to these traditional methods have been related to improving the numerical conditioning of the optimization problem, as exemplified by the extended interior methods. A formulation is presented by Prasad [8] which offers a general class of penalty functions and also avoids the occurrence of extremely large numerical values for the penalty associated with large constraint violations. The variable penalty function approach creates a penalty which is dependent on three parameters: s, α , and ε as follows:

Fors≠1

$$\tilde{g}_j(x) = \frac{\left[-g_j(x)\right]^{1-s}}{s-1} \qquad if\ (g_j(x) \le \varepsilon \qquad \ldots\ldots\ldots (20)$$

$$\tilde{g}_j(x) = \left(\alpha\left[\frac{g_j(x)}{\varepsilon}-1\right]^3 + \frac{s}{2}\left[\frac{g_j(x)}{\varepsilon}-1\right]^2 - \left[\frac{g_j(x)}{\varepsilon}-1\right] + \frac{1}{s-1}\right)(-\varepsilon)^{1-s} \quad if\ (g_j(x) > \varepsilon) \quad \ldots(21)$$

and

For s=1.

$$\tilde{g}_j(x) = -\log[-g_j(x)] \qquad if\ (g_j(x) \le \varepsilon) \qquad \ldots\ldots (22)$$

$$\tilde{g}_j(x) = \alpha\left[\frac{g_J(x)}{\varepsilon}-1\right]^3 + \frac{1}{2}\left[\frac{g_j(x)}{\varepsilon}-1\right]^2 - \left[\frac{g_j(x)}{\varepsilon}-1\right] - \log(-\varepsilon) \qquad if\ (g_j(x) > \varepsilon \ldots\ldots (23)$$

α and ε are the two independent penalty parameters which control the shape of the penalty function. These parameters will be determined later. It can be checked that the expressions (20-21) and (20-21) satisfy and its first and second derivatives at the transition point ε. [8].

4. New Generalized Variable Penalty Function

The generalized variable penalty function is add the term in variable penalty function to obtained more accuracy and small error which introduce by round of error is define by :

For s ≠ 1

If n is even

$$\tilde{g}_j(x) = \frac{\left(-g_j(x)\right)^{1-s}}{s-1} \qquad if\ (g_j(x) \le \varepsilon) \qquad \ldots\ldots\ldots (24)$$

$$\tilde{g}_j(x) = \left(\begin{array}{l}\frac{s}{2}\delta\left[\frac{g_k(x)}{\varepsilon}-1\right]^n + \gamma\left[\frac{g_k(x)}{\varepsilon}-1\right]^{n-1} + \frac{s}{2}\beta\left[\frac{g_k(x)}{\varepsilon}-1\right]^{n-2} + \cdots + \alpha\left[\frac{g_k(x)}{\varepsilon}-1\right]^3 \\ + \frac{s}{2}\left[\frac{g_k(x)}{\varepsilon}-1\right]^2 - \left[\frac{g_k(x)}{\varepsilon}-1\right] + \frac{1}{s-1}\end{array}\right)(-\varepsilon)^{1-s} \quad if\ (g_k(x) > \varepsilon) \quad \ldots.(25)$$

To examine the affectivity of the new algorithm, let vi consider two example for n=4 (even) and n=5 (odd) because n=3 has been consider by [8].

when n=4

$$\tilde{g}_j(x) = \frac{\left(-g_j(x)\right)^{1-s}}{s-1} \qquad if\ (g_j(x) \le \varepsilon) \qquad \ldots.(26)$$

$$\tilde{g}_j(x) = \left(\begin{array}{l}\frac{s}{2}\beta\left[\frac{g_k(x)}{\varepsilon}-1\right]^4 + \alpha\left[\frac{g_k(x)}{\varepsilon}-1\right]^3 + \frac{s}{2}\left[\frac{g_k(x)}{\varepsilon}-1\right]^2 \\ -\left[\frac{g_k(x)}{\varepsilon}-1\right] + \frac{1}{s-1}\end{array}\right)(-\varepsilon)^{1-s} \quad if\ (g_k(x) > \varepsilon) \ \ldots\ (27)$$

It become as

$$\tilde{g}_j(x)=\begin{pmatrix} A\beta\left[\frac{g_J(x)}{\varepsilon}\right]^4-(2s\beta-\alpha)\left[\frac{g_J(x)}{\varepsilon}\right]^3+ \\ (3s\beta-3\alpha+A)\left[\frac{g_J(x)}{\varepsilon}\right]^2 \\ -(2s\beta-3\alpha+B)\left[\frac{g_J(x)}{\varepsilon}\right]+\left(A\beta-\alpha+\frac{AB}{C}\right) \end{pmatrix}(-\varepsilon)^{1-s} \quad ....(28)$$

$$\begin{aligned} &where \\ &A=\frac{s}{2} \\ &B=s+1 \\ &C=s-1 \end{aligned}$$

When n is odd

$$\tilde{g}_j(x)=\frac{(-g_j(x))^{1-s}}{s-1} \qquad if\ \ (g_j(x)\le\varepsilon) \qquad ......\ (29)$$

$$\tilde{g}_j(x)=\begin{pmatrix} \gamma\left[\frac{g_k(x)}{\varepsilon}-1\right]^n+\frac{s}{2}\beta\left[\frac{g_k(x)}{\varepsilon}-1\right]^{n-1}+\cdots+\alpha\left[\frac{g_k(x)}{\varepsilon}-1\right]^3 \\ +\frac{s}{2}\left[\frac{g_k(x)}{\varepsilon}-1\right]^2-\left[\frac{g_k(x)}{\varepsilon}-1\right]+\frac{1}{s-1} \end{pmatrix}(-\varepsilon)^{1-s}$$

$$if\ \ (g_k(x)>\varepsilon) \qquad ....\ (30)$$

for example when n=5

$$\tilde{g}_j(x)=\frac{(-g_j(x))^{1-s}}{s-1} \qquad if\ \ (g_j(x)\le\varepsilon) \qquad ....\ (31)$$

$$\tilde{g}_j(x)=\begin{pmatrix} \gamma\left[\frac{g_k(x)}{\varepsilon}-1\right]^5+\frac{s}{2}\beta\left[\frac{g_k(x)}{\varepsilon}-1\right]^4+\alpha\left[\frac{g_k(x)}{\varepsilon}-1\right]^3+ \\ \frac{s}{2}\left[\frac{g_k(x)}{\varepsilon}-1\right]^2-\left[\frac{g_k(x)}{\varepsilon}-1\right]+\frac{1}{s-1} \end{pmatrix}(-\varepsilon)^{1-s}$$

$$if\ \ (g_k(x)>\varepsilon) \qquad ....\ (32)$$

it become as

$$\tilde{g}_j(x)=\begin{pmatrix} \gamma\left[\frac{g_J(x)}{\varepsilon}\right]^5-(5\gamma-A\beta)\left[\frac{g_J(x)}{\varepsilon}\right]^4+(10\gamma-2s\beta+\alpha)\left[\frac{g_J(x)}{\varepsilon}\right]^3 \\ -(10\gamma-3s\beta+3\alpha-A)\left[\frac{g_J(x)}{\varepsilon}\right]^2+(5\gamma-2s\beta+3\alpha-B)\left[\frac{g_J(x)}{\varepsilon}\right] \\ -\left(\gamma-A\beta+\alpha-\frac{AB}{C}\right) \end{pmatrix}(-\varepsilon)^{1-s} \quad ....(33)$$

$$\begin{aligned} &where \\ &A=\frac{s}{2} \\ &B=s+1 \\ &C=s-1 \end{aligned}$$

4.1. Modified Newton's Method:

To apply Newton's method with SUMT procedure ,the point x^* that minimizes the function $\theta_V(x,r)$

$$\theta_v(x,r)=f(x)+r\sum_{k=1}^{l}\phi_v(g_j) \qquad ....\ (34)$$

for a given value of r is found by using an iterative procedure .if x^n is the initial guess for x^* ,a better approximation x^{n+1} is found from

$$x_{n+1}=x_n-\lambda H^{-1}\nabla\theta(x_n,r) \qquad ....\ (35)$$

$$H_{ij}=\frac{\partial^2\theta_v}{\partial x_i^n\partial x_j^n} \qquad ....\ (36)$$

$\nabla\theta_v$ where is the gradient of θ_v, H is the matrix of the second derivatives of $\theta_v(x,r)$ at the point x^n, given by

and λ is the step size from x^n to x^{n+1}, dropping the superscript n and using (16) &(18) can be expressed as

$$H_{ij} = \frac{\partial^2 f(x)}{\partial x_i \partial x_j} + r\sum_{k=1}^{l}\left[\frac{\partial^2 \phi_v(g_k)}{\partial x_i \partial x_j}\right] \qquad \ldots\ldots (37)$$

$$H_{ij} = \frac{\partial^2 f(x)}{\partial x_i \partial x_j} + r\sum_{k=1}^{l}\left\{\varphi_v''(g_k)\left(\frac{\partial g_k}{\partial x_i}\right)\left(\frac{\partial g_k}{\partial x_j}\right) + \varphi_v'(g_k)\left(\frac{\partial^2 g_k(x)}{\partial x_i \partial x_j}\right)\right\} \qquad \ldots\ldots(38)$$

using the definitions of variable penalty function

if n even

$$\frac{\partial^2 \phi_v}{\partial x_i \partial x_j} = \begin{cases} (-g_k)^{-s-1}\left[s\left(\frac{\partial g_k}{\partial x_i}\right)\left(\frac{\partial g_k}{\partial xj}\right) - g_k\left(\frac{\partial^2 g_k}{\partial x_i \partial x_j}\right)\right] & \frac{g_k}{\varepsilon} \le 1 \qquad \ldots\ldots\ldots(39) \\ -\varepsilon^{-s-1}\left(\begin{bmatrix} n(n-1)\frac{s}{2}\delta\left[\frac{g_k}{\varepsilon}-1\right]^{n-2} + (n-1)(n-2)\gamma\left[\frac{g_k}{\varepsilon}-1\right]^{n-3} + \\ (n-2)(n-3)\frac{s}{2}\beta\left[\frac{g_k}{\varepsilon}-1\right]^{n-4} + \cdots\cdots + 6\alpha\left[\frac{g_k}{\varepsilon}-1\right] + s \end{bmatrix}\left(\frac{\partial g_k}{\partial x_i}\right)\left(\frac{\partial g_k}{\partial xj}\right) \right. \\ \left. + \varepsilon\begin{bmatrix} n\frac{s}{2}\delta\left[\frac{g_k}{\varepsilon}-1\right]^{n-1} + (n-1)\gamma\left[\frac{g_k}{\varepsilon}-1\right]^{n-2} + (n-2)\frac{s}{2}\beta\left[\frac{g_k}{\varepsilon}-1\right]^{n-3} \\ + \cdots\cdots + 3\alpha\left[\frac{g_k}{\varepsilon}-1\right]^2 + s\left[\frac{g_k}{\varepsilon}-1\right] - 1 \end{bmatrix}\left[\frac{\partial^2 g_k}{\partial x_i \partial x_j}\right]\right) & \frac{g_k}{\varepsilon} > 1 \quad (40) \end{cases}$$

If we suppose the quantity ΔG_S and $\Delta\varepsilon_S$ is defined

$$\Delta G_S = n(n-1)\frac{s}{2}\delta\left[\frac{g_k}{\varepsilon}-1\right]^{n-2} + (n-1)(n-2)\gamma\left[\frac{g_k}{\varepsilon}-1\right]^{n-3} + \cdots\cdots + 6\alpha\left[\frac{g_k}{\varepsilon}-1\right] + s \qquad \ldots\ldots\ldots (41)$$

$$\Delta\varepsilon_S = n\frac{s}{2}\delta\left[\frac{g_k}{\varepsilon}-1\right]^{n-1} + (n-1)\gamma\left[\frac{g_k}{\varepsilon}-1\right]^{n-2} + \cdots\cdots + 3\alpha\left[\frac{g_k}{\varepsilon}-1\right]^2 + s\left[\frac{g_k}{\varepsilon}-1\right] - 1 \qquad \ldots\ldots\ldots (42)$$

$$\frac{\partial^2 \varphi_v}{\partial x_i \partial x_j} = \begin{cases} g_k^{-s-1}\left[s\left(\frac{\partial g_k}{\partial x_i}\right)\left(\frac{\partial g_k}{\partial xj}\right) - g_k\left(\frac{\partial^2 g_k}{\partial x_i \partial x_j}\right)\right] & \frac{g_k}{\varepsilon} \le 1 \qquad \ldots\ldots\ldots (43) \\ -\varepsilon^{-s-1}\left(\begin{array}{l} [\Delta G_s]\left(\frac{\partial g_k}{\partial x_i}\right)\left(\frac{\partial g_k}{\partial xj}\right) \\ +\varepsilon[\Delta\varepsilon_s]\left[\frac{\partial^2 g_k}{\partial x_i \partial x_j}\right] \end{array}\right) & \frac{g_k}{\varepsilon} > 1 \qquad \ldots\ldots\ldots (44) \end{cases}$$

when n is odd

$$\frac{\partial^2 \varphi_v}{\partial x_i \partial x_j} = \begin{cases} (-g_k)^{-s-1}\left[s\left(\frac{\partial g_k}{\partial x_i}\right)\left(\frac{\partial g_k}{\partial xj}\right) - g_k\left(\frac{\partial^2 g_k}{\partial x_i \partial x_j}\right)\right] & \frac{g_k}{\varepsilon} \le 1 \qquad \ldots\ldots\ldots(45) \\ -\varepsilon^{-s-1}\left(\begin{bmatrix} n(n-1)\gamma\left[\frac{g_k}{\varepsilon}-1\right]^{n-2} + \frac{s}{2}(n-1)(n-2)\beta\left[\frac{g_k}{\varepsilon}-1\right]^{n-3} + \\ \cdots\cdots + 6\alpha\left[\frac{g_k}{\varepsilon}-1\right] + s \end{bmatrix}\left(\frac{\partial g_k}{\partial x_i}\right)\left(\frac{\partial g_k}{\partial xj}\right) \right. \\ \left. +\varepsilon\begin{bmatrix} n\gamma\left[\frac{g_k}{\varepsilon}-1\right]^{n-1} + \frac{s}{2}(n-1)\beta\left[\frac{g_k}{\varepsilon}-1\right]^{n-2} + \\ \cdots\cdots + 3\alpha\left[\frac{g_k}{\varepsilon}-1\right]^2 + s\left[\frac{g_k}{\varepsilon}-1\right] - 1 \end{bmatrix}\left[\frac{\partial^2 g_k}{\partial x_i \partial x_j}\right]\right) \\ & \frac{g_k}{\varepsilon} > 1 \quad \ldots\ldots(46) \end{cases}$$

If we suppose the quantity ΔG_S and $\Delta\varepsilon_S$ is defined

$$\Delta G_S = n(n-1)\gamma\left[\frac{g_k}{\varepsilon}-1\right]^{n-2} + \frac{s}{2}(n-1)(n-2)\beta\left[\frac{g_k}{\varepsilon}-1\right]^{n-3} + \cdots\cdots + 6\alpha\left[\frac{g_k}{\varepsilon}-1\right] + s \quad \ldots\ldots (47)$$

$$\Delta\varepsilon_S = n\gamma\left[\frac{g_k}{\varepsilon}-1\right]^{n-1} + \frac{s}{2}(n-1)\beta\left[\frac{g_k}{\varepsilon}-1\right]^{n-2} + \cdots\cdots + 3\alpha\left[\frac{g_k}{\varepsilon}-1\right]^{2} + s\left[\frac{g_k}{\varepsilon}-1\right] - 1 \ldots\ldots (48)$$

$$\frac{\partial^2\varphi_v}{\partial x_i\partial x_j} = \begin{cases} (-g_k)^{-s-1}\left[s\left(\frac{\partial g_k}{\partial x_i}\right)\left(\frac{\partial g_k}{\partial xj}\right) - g_k\left(\frac{\partial^2 g_k}{\partial x_i\partial x_j}\right)\right] & \frac{g_k}{\varepsilon} \le 1 \quad \ldots..(49) \\ -\varepsilon^{-s-1}\begin{pmatrix} [\Delta G_s]\left(\frac{\partial g_k}{\partial x_i}\right)\left(\frac{\partial g_k}{\partial xj}\right) \\ +\varepsilon[\Delta\varepsilon_s]\left[\frac{\partial^2 g_k}{\partial x_i\partial x_j}\right] \end{pmatrix} & \frac{g_k}{\varepsilon} > 1 \quad \ldots. (50) \end{cases}$$

4.2. Determination of Constant

In order to establish a suitable value for constant, it is desirable to find the upper and lower limits that constant can assume without compromising the characteristics of a penalty function. The shape of the variable penalty function curves depends on constant. In order to ensure a higher penalty for a higher constraint violation, a curve increasing monotonically with negative g_k is needed. The slope of the variable penalty function is obtained as:

for n odd

$$\varphi'(g_k) = \begin{cases} -g_k^{-s} & \frac{g_k}{\varepsilon} \ge 1 \quad \ldots\ldots (51) \\ -\varepsilon^{-s}\begin{bmatrix} n\gamma\left[\frac{g_k}{\varepsilon}-1\right]^{n-1} + (n-1)\frac{s}{2}\beta\left[\frac{g_k}{\varepsilon}-1\right]^{n-2} + \cdots \\ \cdots + 3\alpha\left[\frac{g_k}{\varepsilon}-1\right]^{2} + s\left[\frac{g_k}{\varepsilon}-1\right] - 1 \end{bmatrix} & \frac{g_k}{\varepsilon} < 1 \quad \ldots\ldots (52) \end{cases}$$

$$\Delta\varepsilon_S = n\gamma\left[\frac{g_k}{\varepsilon}-1\right]^{n-1} + \frac{s}{2}(n-1)\beta\left[\frac{g_k}{\varepsilon}-1\right]^{n-2} + \cdots\cdots + 3\alpha\left[\frac{g_k}{\varepsilon}-1\right]^{2} + s\left[\frac{g_k}{\varepsilon}-1\right] - 1 = 0 \quad \ldots(53) \qquad \psi'(g_k) = 0$$

However, we can see from negative values of constant increase the magnitude of the associated error $\Delta\varepsilon_S$. Thus, one has to limit constant to positive values. For such positive values of constant, the penalty function does not show a strictly increasing monotonic behavior. It is thus important to select a positive value for constant which ensures an increasing penalty behavior, at least up to the most negative constraint that one may encounter. This requirement can be set as

$$\psi\left(\frac{g_k}{\varepsilon} = \tilde{d}\right) \ge \psi\left(\frac{g_k}{\varepsilon} = d^*\right) \qquad \ldots\ldots\ldots (54)$$

where d^* is the must negative constraint ratio and $\tilde{d}$ is a value of $\frac{g_k}{\varepsilon}$ for which $\psi'(\tilde{d}) = 0$ a limiting situation would occur when $\tilde{d}$ equals d^*, i.e. the penalty for the most critical constrain violation is a maximum at the value specified by must negative possible constrain . using this , $\Delta\varepsilon = 0$ the range of α can be found as follows :

$$\alpha \le \left[\frac{1 - s\left(d^* - 1\right)}{3\left(d^* - 1\right)^2}\right] \quad \ldots\ldots. (55)$$

for the possible range of g_k i.e $0 \le \alpha < -\infty$,the bounds on α can be established ($\alpha^* = 0$)

$$0 \le \alpha \le \frac{1+s}{3} \qquad \ldots\ldots (56)$$

the value α = 0 corresponds to the case when an infinitely negative value of α^* is allowed; in this particular situation (α = 0), the variable penalty function formulation for s = 2 degenerates to a quadratic extended interior penalty function . The value α = (1 + s)/3 corresponds to the case when α^* = 0. Particularly for this value of α ,

$$\Delta\varepsilon_s = (1+s)\left[\frac{g_k}{\varepsilon}\right]^2 - (2+s)\left[\frac{g_k}{\varepsilon}\right] \qquad ..... (57)$$

To generalize the new idea mentioned in this paper , the positive deftness property will be generalized for $n \geq 4$ because for $n \leq 3$, motioned in [8] .

where n=5

$$\Delta\varepsilon_S = 5\gamma\left[\frac{g_k}{\varepsilon}-1\right]^4 + \frac{s}{2}4\beta\left[\frac{g_k}{\varepsilon}-1\right]^3 + 3\alpha\left[\frac{g_k}{\varepsilon}-1\right]^2 + s\left[\frac{g_k}{\varepsilon}-1\right] - 1 = 0 \qquad ...... (58)$$

$$\gamma = \frac{-\beta}{5\left[\frac{g_k}{\varepsilon}-1\right]^4} \qquad ...... (59)$$

$$0 \leq \gamma \leq = \frac{-(1+s)}{30s} \qquad ...... (60)$$

$$\Delta\varepsilon_s = \frac{-(1+s)}{6s}\left(\frac{g_k}{\varepsilon}\right)^4 + \frac{(s^2+3s+2)}{3s}\left(\frac{g_k}{\varepsilon}\right)^3 - \frac{(1+s)}{s}\left(\frac{g_k}{\varepsilon}\right)^2 + \frac{(2+s)}{3s}\left(\frac{g_k}{\varepsilon}\right) - \frac{1+s}{3} \qquad ......(61)$$

if n is odd in general the new constant is

$$0 \leq \gamma_{new} \leq \frac{-\beta_{old}}{n\left[\frac{g_k}{\varepsilon}-1\right]^{n-1}} \qquad ...... (62)$$

when n even

$$\varphi'(g_k) = \begin{pmatrix} -g_k^{-s} & \frac{g_k}{\varepsilon} \geq 1 \quad ... (63) \\ -\varepsilon^{-s}\begin{bmatrix} n\frac{s}{2}\delta\left[\frac{g_k}{\varepsilon}-1\right]^{n-1} + (n-1)\gamma\left[\frac{g_k}{\varepsilon}-1\right]^{n-2} + \cdots \\ \cdots + 3\alpha\left[\frac{g_k}{\varepsilon}-1\right]^2 + s\left[\frac{g_k}{\varepsilon}-1\right] - 1 \end{bmatrix} & \frac{g_k}{\varepsilon} < 1 \quad ....(64) \end{pmatrix}$$

$$\Delta\varepsilon_S = n\frac{s}{2}\delta\left[\frac{g_k}{\varepsilon}-1\right]^{n-1} + (n-1)\gamma\left[\frac{g_k}{\varepsilon}-1\right]^{n-2} + \cdots + 3\alpha\left[\frac{g_k}{\varepsilon}-1\right]^2 + s\left[\frac{g_k}{\varepsilon}-1\right] - 1 = 0 \qquad ....(65)$$

If n is even in general the new constant is

$$0 \leq \beta_{new} \leq \frac{\alpha_{old}}{n\frac{s}{2}\left[\frac{g_k}{\varepsilon}-1\right]^{n-1}} \qquad ...... (66)$$

for example n=4

$$\Delta\varepsilon_S = 4\frac{s}{2}\beta\left[\frac{g_k}{\varepsilon}-1\right]^3 + 3\alpha\left[\frac{g_k}{\varepsilon}-1\right]^2 + s\left[\frac{g_k}{\varepsilon}-1\right] - 1 = 0 \qquad ..... (67)$$

$$\beta = \frac{-\alpha}{2s\left[\frac{g_k}{\varepsilon} - 1\right]^3} \quad \text{....... (68)}$$

$$0 \le \beta \le \; = \frac{1+s}{6s} \quad \text{...... (69)}$$

$$\Delta\varepsilon_s = \frac{1+s}{3}\left(\frac{g_k}{\varepsilon}\right)^3 - \left(\frac{g_k}{\varepsilon}\right) - \frac{1+s}{3} \quad \text{....... (70)}$$

4.3. Selecting constants and Updating:

During an iteration, with any arbitrary starting point, the following conditions can exist:

(a) all the constraints $g_k(x)$ are satisfied;

(b) all the constraints $g_k(x)$ are violated; and

(c) some of the constraints are satisfied and some are not, i.e., mixed.

there are two case

First case:

The first iteration in our loop of new algorithm s.t.:

1. If condition (a) arises and if n is even

the value of new constant is selected using

$$\beta_{new} = \frac{\alpha_{old}}{n\,\frac{s}{2}} \quad \text{....... (71)}$$

this comes from (56) and ensures the minimum error in the approximation of the hessian matrix.

If n is odd

the value of new constant is selected using

$$\gamma_{new} = \frac{-\beta_{old}}{n} \quad \text{...... (72)}$$

2. If the condition (b) or (c) arises and if n odd

$$\gamma_{new} \le \frac{-\beta_{old}}{n\left[\frac{g_k}{\varepsilon} - 1\right]^{n-1}} \quad \text{.......... (73)}$$

If n is even

$$\beta_{new} \le \frac{\alpha_{old}}{n\,\frac{s}{2}\left[\frac{g_k}{\varepsilon} - 1\right]^{n-1}} \quad \text{........ (74)}$$

in which g* represents the most violated constraint encountered during an iteration.

Succeeding case:

The succeeding iterations in our loop of new algorithm s.t. in the subsequent iterations, the VPM algorithm determines the degree of severity on the constraints. If, at any instant, condition (a) occurs, the value for α is determined using α = (1 + s)/3.

If condition (b) appears, it is based on value of n:

International Journal of Enhanced Research in Science Technology & Engineering, ISSN: 2319-7463
Vol. 3 Issue 4, April-2014, pp: (44-59), Impact Factor: 1.252, Available online at: www.erpublications.com

if n is even :

the new constant is selected based on Eq , i.e.,

$$\beta \geq \frac{-s}{n(n-1)\frac{s}{2}\left(1-\frac{g_k}{\varepsilon}\right)^{n-2}} \quad \text{……... (75)}$$

if n is odd

the new constant is selected based on Eq. , i.e.,

$$\gamma \leq \frac{s}{n(n-1)\left(1-\frac{g_k}{\varepsilon}\right)^{n-2}} \quad \text{…….. (76)}$$

in which g* represents the most violated constraint encountered during an iteration.

4. 4. The New Initial Value of Algorithm:

initial Value of the Penalty Parameter (r_k). Since the unconstrained minimization of $\theta(x,r_k)$ is to be carried out for a decreasing sequence of r_k, it might appear that by choosing a very small value of r_0, we can avoid an excessive number of minimizations of the function θ . But from a computational point of view, it will be easier to minimize the unconstrained function $\theta(x,r_k)$,the numerical values of r_k has to be chosen carefully in order to achieve a faster convergence. we have to find r_k such that depend on $\phi(x)$ [9].
the initial value r0 which is derived as

$$\theta(x,r_k) = f(x) + r_k\phi(x) \quad \text{...….. (77)}$$

such that $\theta(x,r_k) = 0$

we have

$$f(x) + r_k\phi(x) = 0 \quad \text{….…. (78)}$$

now $r_k > 0$, then we obtain

$$r_{\min} = \frac{-f(x)}{\phi(x)} \quad \text{…….. (7)}$$

In the above suggestion corresponding to the assumption for deriving a new parameter to make balance between the previous algorithms, we have suggested the following a new algorithm.

4.5. New Theorem

Consider problem to minimize $f(x)$ subject $g_i(x) \geq 0$ for $i = 1,2,\ldots,m$. Let KKT condition is satisfying the second order sufficiency condition for a local minimum. Defined $I = \{i : g_i(\bar{x}) = 0\}$, $N = \{i : g_i(\bar{x}) < 0\}$ and the cone $C = \{d \neq 0, \nabla g_i(\bar{x})d = 0$ for $i \in I$ and $\nabla g_i(\bar{x})d > 0$ for all $i \in N\}$. Then, if there exists r_k such that $r_k \succ r_{k+1}$ therefore $\nabla^2\theta(x,r_k)$ is a positive definite $\forall\, d \in C$ and $\bar{x}$ is strict local minimum for (1) for all $r \geq 0$.

Proof :

Since $(\bar{x},\bar{\vartheta},\bar{\omega})$ is KKT a solution satisfy the second –order sufficiency condition for a local minimum in cone c and $\nabla^2 L(x,\vartheta,\omega)$ the Hessian of the Lagrange function of (1). Suppose that there exists d_k with $\|d_k\|=1$, such that

$$\theta(x_k,r_k)=f(x_k)+r_k\sum_{i=1}^{m}\varphi[g_i(x)] \qquad \text{.........(80)}$$

Thus, the gradient of $\theta(x_k,r_k)$ should be defined by

$$\nabla\theta(x_k,r_k)=\nabla f(x_k)+r\sum_{i=1}^{m}\varphi'(g_i(x))\nabla g_i(x) \qquad \text{.......... (81)}$$

The second derivatives of $\theta(x_k,r_k)$ defined by

$$\nabla^2\theta(x_k,r_k)=\nabla^2 f(x_k)+r\sum_{i=1}^{m}\varphi'[g_i(x)]\nabla^2 g_i(x)$$

$$+r\sum_{i=1}^{m}\varphi''[g_i(x)]\nabla g_i(x)\nabla g_i(x)^T \qquad \text{...(82)}$$

$$\nabla^2\theta(x_k,r_k)=\nabla^2 L(x,\vartheta,\omega)+r\sum_{i=1}^{m}\varphi''[g_i(x)]\nabla g_i(x)\nabla g_i(x)^T \qquad \text{..........(83)}$$

Where $\nabla^2 L(x,\vartheta,\omega)$ is the Hessian Lagrangian function for eq.(1)with multiplier ϑ and ω

$$d_k^T\nabla^2\theta(x_k,r_k)d_k = d_k^T\nabla^2 L(x,\vartheta,\omega)d_k$$

$$+d_k^T r\sum_{i=1}^{m}\varphi''[g_i(x)]\nabla g_i(x)\nabla g_i(x)^T d_k \qquad \text{........(84)}$$

Clearly the first term $\nabla^2 L(\bar{x},\bar{\vartheta},\bar{\omega})$ is positive definite on the cone C, then we shall prove the second term of (83)

If n is even then the second derivative is define by :

$$\frac{\partial^2\varphi}{\partial x_i\partial x_j}=\begin{cases}(-g_k)^{-s-1}\left[s\left(\dfrac{\partial g_k}{\partial x_i}\right)\left(\dfrac{\partial g_k}{\partial xj}\right)\right] & \dfrac{g_k}{\varepsilon}\le 1\\[2ex] -\varepsilon^{-s-1}\left(\begin{bmatrix} n(n-1)\dfrac{s}{2}\delta\left[\dfrac{g_k}{\varepsilon}-1\right]^{n-2}+(n-1)(n-2)\gamma\left[\dfrac{g_k}{\varepsilon}-1\right]^{n-3}+\\ (n-2)(n-3)\dfrac{s}{2}\beta\left[\dfrac{g_k}{\varepsilon}-1\right]^{n-4}+\cdots\cdots+6\alpha\left[\dfrac{g_k}{\varepsilon}-1\right]+s\end{bmatrix}\left(\dfrac{\partial g_k}{\partial x_i}\right)\left(\dfrac{\partial g_k}{\partial xj}\right)\right) & \dfrac{g_k}{\varepsilon}>1\end{cases} \qquad \text{.......(85)}$$

we require prove that $\phi''(g_i)\ge 0$ which depend on the constants

$$\beta\ge\frac{-s}{n(n-1)\dfrac{s}{2}\left(1-\dfrac{g_k}{\varepsilon}\right)^{n-2}} \qquad \text{..........(86)}$$

So we have $d_k^T \nabla^2 \theta(x_k, r_k) d_k \geq 0$ for all $i \in I$ or $i \in N$ and so we have $\bar{x}$ is a strict local minimum.

Now if n is odd then the second derivative is define by :

$$\frac{\partial^2 \varphi}{\partial x_i \partial x_j} = \begin{cases} (-g_k)^{-s-1}\left[s\left(\frac{\partial g_k}{\partial x_i}\right)\left(\frac{\partial g_k}{\partial xj}\right)\right] & \frac{g_k}{\varepsilon} \leq 1 \\ -\varepsilon^{-s-1}\left(\left[\begin{matrix} n(n-1)\gamma\left[\frac{g_k}{\varepsilon}-1\right]^{n-2} + \frac{s}{2}(n-1)(n-2)\beta\left[\frac{g_k}{\varepsilon}-1\right]^{n-3} + \\ \cdots\cdots + 6\alpha\left[\frac{g_k}{\varepsilon}-1\right] + s \end{matrix}\right]\left(\frac{\partial g_k}{\partial x_i}\right)\left(\frac{\partial g_k}{\partial xj}\right)\right) & \frac{g_k}{\varepsilon} > 1 \end{cases} \quad \text{….(87)}$$

we require prove that $\phi''(g_i) \geq 0$ which depend on the constants

$$\alpha \leq \frac{s}{6\left(1 - \frac{g^*}{\varepsilon}\right)} \quad \text{…….. (88)}$$

$$\gamma \leq \frac{s}{n(n-1)\left(1 - \frac{g_k}{\varepsilon}\right)^{n-2}} \quad \text{…….. (89)}$$

So we have $d_k^T \nabla^2 \theta(x_k, r_k) d_k \geq 0$ for all $i \in I$ or $i \in N$ and so we have $\bar{x}$ is a strict local minimum.

4.6. Outline New Extended Interior Methods

Step1: Find an initial approximation x0 in the interior of the feasible region for the inequality constraints i.e. $g_i(x_0) > 0$.

Step2: Set $i = 1$ and $r_0 = 1$ is the initial value of r_0.

Step3: Set $d_i = -H_i g_i$

Step5: Set $x_{i+1} = x_i + \lambda_i d_i$, where λ is a scalar.

Step6: Update H by correction matrix defined in (83) .

Step7: Check for convergence i.e. if $\|x_i - x_{i-1}\| < \delta$ satisfied then stop,

otherwise, continue .

Step8: Set $r_{i+1} = \frac{r_i}{10}$ and take x=x* and set k=k+1 and go to step 5.

Results and Conclusion

Several standard non-linear constrained test functions were minimized to compare the new algorithms with standard algorithm see (Appendix,A). with $1 \leq m \leq 4$ and $1 \leq g_i(x) \leq 4$ .This paper includes five parts.. Is considered as the comparative performance of the following algorithm.

3- variable extended (n=3) .

4 – new generalize variable extended

(n is even n=4 , 6, 8,……..).

5 - new generalize variable extended

(n is odd n=5 , 7, ….).

We denoted linear extended(LE), quadratic extended(QE), variable extended (VE) , new generalize variable extended (n is even) (NEW 1) , new generalize variable extended (n is odd) (NEW 2) .

1- linear extended (n=1) .

2- quadratic extended (n=2) .

All the results are obtained using (Laptop) . All programs are written in visual FORTRAN language and for all cases the stopping criterion taken to be $\|x_i - x_{i-1}\| < \delta$, $\delta = 10^{-5}$

All the algorithms in this paper use the same ELS strategy which is the quadratic interpolation technique directly adapted from [3] .

The comparative performance for all of these algorithms are evaluated by considering NOF, NOI, NOG and NOC, where NOF is the number of function evaluation and NOI is the number of iteration and NOG is the number of gradient evaluation and NOC number of constrained evaluation.

In table (1) we have compared our new algorithm NEW1 with quadratic extended

In table (2) we have compared our new algorithm NEW2 with linear – variable extended

Table (1)

Comparison of New1 algorithm quadratic algorithm

NO	NO.	N=2 QE NOF(NOG)NOI(NOC)	N=4 NEW1 NOF(NOG)NOI(NOC)	N=6 NEW1 NOF(NOG)NOI(NOC)	N=8 NEW1 NOF(NOG)NOI(NOC)
1	ε =.5 R=.01 S=2	13471(500)1(1)	250(4)3(3)	215(7)2(1)	220(8)2(1)
2	ε =.05 R=.01 S=2	18906(500)1(1)	10481(500)2(1)	9483(500)2(1)	9446(500)2(1)
3	ε =.05 R=1 S=2	18958(500)1(1)	308(11)2(1)	297(12)2(1)	296(12)2(1)
4	ε =.2 R=.01 S=2	18401(500)1(1)	901(116)4(5)	413(32)4(5)	285(6)3(4)
5	ε =.02 R=.01 S=2	107(2)1(1)	19425(500)4(5)	14936(500)4(5)	4493(500)2(1)
6	ε =.5 R=.0001 S=2	209(4)2(1)	209(4)2(1)	209(4)2(1)	209(4)2(1)
To.		70052(2006)7(6)	31574(1135)17(16)	25553(1055)16(14)	14949(1030)13(9)

Table (2)

Comparison of New2 algorithm linear – variable algorithm

NO	NO.	N=1 LE NOF(NOG)NOI(NOC)	N=3 VE NOF(NOG)NOI(NOC)	N=5 NEW2 NOF(NOG)NOI(NOC)	N=7 NEW2 NOF(NOG)NOI(NOC)
1	ε=.5 R=.01 S=2	654(9)1(1)	30(2)1(1)	20(2)1(1)	18(2)1(1)
2	ε=.5 R=.01 S=2	10380(167)2(1)	6794(500)1(1)	4979(500)1(1)	2922(362)1(1)
3	ε=.05 R=.01 S=2	22496(364)1(1)	8500(500)1(1)	29(2)1(1)	27(2)1(1)
4	ε=.2 R=.01 S=2	239(4)3(3)	6970(500)1(1)	5878(500)1(1)	14(3)1(1)
5	ε=.05 R=.1 S=2	11443(154)3(3)	25900(500)1(1)	17284(500)1(1)	16(4)1(1)
6	ε=.5 R=.0001 S=2	545(58)1(1)	209(4)2(1)	209(4)2(1)	209(4)2(1)
To.		45757(756)11(10)	48403(2006)7(6)	28399(1508)7(6)	3206(377)7(6)

Reference

[1]. [1] Al. Bayati , A.Y. and Qasim, A.M, "anew hybrid algorithm to the modified barrier function form " ,J. of Edit & Sci, mosul ,Iraq , vol.21 , NO.4, pp.131-148, 2008.
[2]. [2] Bazaraa, M.S. and Sherali, H.D. and Shetty, C. M, " Nonlinear Programming: Theory and Algorithms",3rd ed., John Wiley & Sons, 2006 .
[3]. [3] Bunday, B.D , "Basic Optimization Methods", Edward Arnold, London, 1984 .
[4]. [4] Fiacco, A.V. and McCormick, G. P. , " Nonlinear Programming: Sequential Unconstrained Minimization Techniques", the Society for Industrial and Applied Mathematics, New York , 1990 .
[5]. [5] Freund , Robert M . , "penalty and barrier methods for constrained optimization " ,Massachusetts Institute Of Technology , 2004 .
[6]. [6] Haftka, R. T., and Starnes jr, J. H.,"Applications of a Quadratic Extended Interior Penalty Function for Structural Optimization", AIAA J., vol. 14, no. 6, pp. 718 – 724, 1976 .
[7]. [7] Nocedal, J. and Wright J. , " Numerical Optimization" , Springer Series in Operations Research, Springer Verlag, New York, USA , 2006 .
[8]. [8] Prasad, B. , "A Class of Generalized Variable Penalty Methods for Nonlinear Programming" , J. Optim. Theory Appl., vol. 35, no. 2, pp. 159 – 182, October 1981 .
[9]. [9] Rao, Singireas S.,"engineering optimization theory and practice ", john wiley & sons Inc. All rights reserved , 2009 .
[10]. [10] Vanderplaats, C.,"Very Large Scale Optimization", NASA/CR , 2002 .

Appendix , A

1. min $f(x) = (x_1 - 2)^2 + .25\,x_2^2$

s.t

$$2x_1 + 3x_2 - 4$$
$$-x_1 + 3.5x_2 + 1$$
$$x = [3,5]$$

2. min $f(x) = x_1 x_2$

s.t

$$25 - x_1^2 - x_2^2$$
$$x_1 + x_2$$
$$x = [3,2]$$

3. min $f(x) = (x_1 - 2)^2 + (x_2 - 1)^2$

s.t

$$-x_1^2 + x_2$$
$$x_1 + x_2 - 2$$
$$x = [2,2]$$

4. min $f(x) = x_1^2 + x_2^2$

s.t

$$x_1 + 2x_2 - 4$$
$$-x_1^2 - x_2^2 + 5$$
$$x_1$$
$$x_2$$
$$x = [.9, 1.3]$$

5. min $f(x) = (x_1 - 2)^2 + (x_2 - 1)^2$

s.t

$$x_1 - 2x_2 + 1$$
$$-\frac{x_1^2}{4} - x_2^2 + 1$$
$$x = [.7, .7]$$

6. min $f(x) = (x_1 - 2)^2 + (x_2 - 1)^2$

s.t

$$x_1 - 2x_2 - 1$$
$$-x_1^2 + x_2$$