

AN ACTIVE SET NEWTON'S INTERIOR-POINT ALGORITHM FOR SOLVING A CONSTRAINED OPTIMIZATION PROBLEM

B. EL-SOBKY AND Y. ABO-ELNAGA

ABSTRACT. In this paper, an active set Newton's interior-point algorithm for solving a constrained optimization problem is introduced. An active-set technique is used to convert the inequality constraint of constrained optimization problem to equality constraint. A Coleman-Li scaling matrix is used together with Newton's interior point method to solve the constrained optimization problem.

A Matlab implementation of the active set Newton's interior-point algorithm was used in solving three test problems and the results are reported. The results show that our approach is of value and merit further investigations.

1. INTRODUCTION

Operation research is concerned with the application of scientific tools and techniques to decision-making problems involving operations of an integrated system so as to provide optimum solutions to a problem. The methods of operations research are very often used in management science, industrial engineering, mathematics, economics, etc. to analyze complex real-world systems, typically with the goal of improving or optimizing performance. It provides an understanding that gives the managers new insights and capabilities to determine better solutions in their decision-making problems. In the real world, many decision-making problems can be described by using a general constrained optimization model.

In this paper, we introduce an active set Newton's interior-point algorithm for solving the constrained optimization problem. The proposed algorithm uses the active-set strategy to convert the inequality constraint of constrained optimization problem to equality constraint. The chief feature of the proposed active set is that the active set is identified and updated naturally by the trial step. Many authors, have considered active set techniques for solving general nonlinear programming problems [[4], [8], [9], [10], [11], [12], [13], [14], [16], [20], [21], [23], [24]].

2010 *Mathematics Subject Classification.* MSC: 39B05, 37N30, 37N20.

Key words and phrases. Constrained optimization , Active set, Newton's interior-point method, Coleman-Li scaling matrix .

Submitted May 7, 2015.

The proposed algorithm uses the Coleman-Li scaling matrix together with Newton's interior point method to solve the constrained optimization problem. The Coleman-Li scaling matrix was first introduced by Coleman-Li [1] for unconstrained optimization problem and Newton's interior point method was suggested by Das [2]. The Coleman-Li scaling matrix was generalized to general nonlinear programming problem by [[2], [5], [6], [7]].

The following notations are used throughout the rest of the paper. The sequence of points generated by the algorithm is denoted $\{x_k\}$. A subscripted function means the value of the function evaluated at a particular point. For example, $f_k \equiv f(x_k)$, $G_k \equiv G(x_k)$, $\nabla f_k \equiv \nabla f(x_k)$, $\nabla G_k \equiv \nabla G(x_k)$, $\phi_k \equiv \phi(x_k; \rho_k)$, $W_k \equiv W(x_k)$ and so on. We use the notation $x_k^{(j)}$ to denote the j th component of the vector x_k , $(\nabla \phi(x_k; \rho_k))_k^{(j)}$ to denote the j th component of the vector $\nabla \phi_k$, and so on.

The paper is organized as follows. In Section 2, we describe, an active set strategy and Newton's method. In Section 3, we present a reduced method that computes Newton's step by solving a smaller dimension linear system. In Section 4, we introduce an interior-point method and a formal description of the active set Newton's interior-point algorithm for solving the constrained optimization problem is presented. Section 5 contains a Matlab implementation of the active set Newton's interior-point algorithm. Three case studies are presented and the results are reported. Finally, Section 6 contains concluding remarks.

2. AN ACTIVE SET STRATEGY AND NEWTON'S METHOD

Minimization problems with upper and /or lower bounds on some of the variables form important and common class of problems. In this paper, we consider the following constrained optimization problem

$$\begin{aligned} & \text{minimize} && f(x) \\ & \text{subject to} && g_i(x) = 0 \quad i \in E, \\ & && g_i(x) \leq 0 \quad i \in I, \\ & && \alpha \leq x \leq \beta, \end{aligned} \tag{2.1}$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $g_i : \mathbb{R}^n \rightarrow \mathbb{R}^m$, $E \cup I = \{1, \dots, m\}$ and $E \cap I = \emptyset$, $\alpha \in \{\mathbb{R} \cup \{-\infty\}\}^n$, $\beta \in \{\mathbb{R} \cup \{\infty\}\}^n$, $m < n$, and $\alpha < \beta$. The functions f and g_i , $i = \{1, \dots, m\}$ are assumed to be at least twice continuously differentiable.

Various optimization techniques have been proposed by many researchers to deal with the above constrained optimization problem with varying degree of success. The proposed algorithm uses an indicator matrix [4] to convert the inequality constraint to equality constraint. The indicator matrix $W(x) \in \mathbb{R}^{m \times m}$ is a diagonal matrix whose diagonal entries are

$$w_i(x) = \begin{cases} 1, & \text{if } i \in E, \\ 1, & \text{if } i \in I \text{ and } g_i(x) \geq 0, \\ 0, & \text{if } i \in I \text{ and } g_i(x) < 0. \end{cases} \tag{2.2}$$

Using the above matrix, Problem (2.1) can be transformed to the following problem

$$\begin{aligned} & \text{minimize} && f(x), \\ & \text{subject to} && G(x)^T W(x) G(x) = 0, \\ & && \alpha \leq x \leq \beta, \end{aligned} \tag{2.3}$$

where $G(x) = (g_1(x), \dots, g_m(x))^T$ is continuously differentiable function.

The Lagrangian function associated with Problem (2.3) is given by

$$L(x, \lambda, \mu; \rho) = f(x) - \lambda^T(x - \alpha) - \mu^T(\beta - x) + \rho \|W(x)G(x)\|^2, \quad (2.4)$$

where ρ is positive parameter, λ and μ are lagrange multiplier vectors associated with the inequality constraints $(x - \alpha)$ and $(\beta - x)$ respectively.

The first-order necessary conditions for a point x_* to be a solution of Problem (2.1) are the existence of multipliers $\lambda_* \in \mathfrak{R}_+^n$, and $\mu_* \in \mathfrak{R}_+^n$, such that (x_*, λ_*, μ_*) satisfies

$$\nabla \phi(x_*; \rho_*) - \lambda_* + \mu_* = 0, \quad (2.5)$$

$$\alpha \leq x_* \leq \beta, \quad (2.6)$$

and for all j corresponding to $x^{(j)}$ with finite bound, we have

$$\lambda_*^{(j)}(x_*^{(j)} - \alpha^{(j)}) = 0, \quad (2.7)$$

$$\mu_*^{(j)}(\beta^{(j)} - x_*^{(j)}) = 0, \quad (2.8)$$

where $\nabla \phi(x_*; \rho_*) = \nabla f(x_*) + \rho_* \nabla G(x_*)W(x_*)G(x_*)$.

Following [[1], [2], [5]], we define the diagonal scaling matrix $S(x)$ whose diagonal elements are given by

$$s^{(j)}(x) = \begin{cases} \sqrt{(x^{(j)} - \alpha^{(j)})}, & \text{if } \nabla \phi(x; \rho)^{(j)} \geq 0 \text{ and } \alpha^{(j)} > -\infty, \\ \sqrt{(\beta^{(j)} - x^{(j)})}, & \text{if } \nabla \phi(x; \rho)^{(j)} < 0 \text{ and } \beta^{(j)} < +\infty, \\ 1, & \text{otherwise.} \end{cases} \quad (2.9)$$

Using the scaling matrix $S(x)$, the first order necessary conditions (2.5)-(2.8) are equivalent to the following equation

$$S^2(x)\nabla \phi(x; \rho) = 0, \quad (2.10)$$

and the point x_* satisfies the box constraint

$$\alpha \leq x \leq \beta. \quad (2.11)$$

Notice that the system (2.10) is continuous but not everywhere differentiable. The non-differentiability occurs when $s^{(j)} = 0$. These points are avoided by restricting x in the interior of the box constraint (2.11). The other non-differentiability occurs when a variable $x^{(j)}$ has a finite lower bound and an infinite upper bound and $(\nabla \phi(x; \rho))^{(j)} = 0$. But these points are not significant, so we define a vector $\eta(x)$ whose components are $\eta^{(j)}(x) = \frac{\partial((s^{(j)})^2)}{\partial x^{(j)}}$, $j = 1, \dots, n$ such that $\eta^{(j)}$ to be zero whenever $(\nabla \phi(x; \rho))^{(j)} = 0$. Hence, we can write

$$\eta^{(j)}(x) = \begin{cases} 1, & \text{if } \nabla \phi(x; \rho)^{(j)} \geq 0 \text{ and } \alpha^{(j)} > -\infty, \\ -1, & \text{if } \nabla \phi(x; \rho)^{(j)} < 0 \text{ and } \beta^{(j)} < +\infty, \\ 0, & \text{otherwise.} \end{cases} \quad (2.12)$$

Applying Newton's method on the nonlinear system (2.10) and assuming $\alpha < x < \beta$, we have the following linear system

$$[S^2(x)\nabla^2 \phi(x; \rho) + \text{diag}(\nabla \phi(x; \rho))\text{diag}(\eta(x))]\Delta x = -S^2(x)\nabla \phi(x; \rho), \quad (2.13)$$

where $\nabla^2 \phi(x; \rho) = \nabla^2 f(x) + \rho \nabla G(x)W(x)\nabla G(x)^T$.

Newton's step is computed by solving the above linear system (2.13) for Δx . One of the disadvantages of obtaining Newton's step by solving the linear system (2.13) lies in the fact that the dimension of the linear system (2.13) is large for large-scale problems. In the following section, we present a reduced method that computes Newton's step by solving a smaller dimension linear system.

3. A REDUCED METHOD

Consider an $n \times (n - m)$ matrix $Z(x)$ with columns that form an orthonormal basis for the null space of $(S^2(x)\nabla G(x)W(x)G(x))^T$. i.e.

$$Z(x)^T S^2(x)\nabla G(x)W(x)G(x) = 0, \quad (3.1)$$

where the matrix $Z(x)$ depends on ρ . Here ρ is omitted to simplify the notation.

In this paper, the matrix $Z(x)$ can be obtained from the QR factorization of $S^2(x)\nabla G(x)W(x)G(x)$ as follows:

$$S^2(x)\nabla G(x)W(x)G(x) = \begin{bmatrix} Y(x) & Z(x) \end{bmatrix} \begin{bmatrix} R(x) \\ 0 \end{bmatrix}. \quad (3.2)$$

The orthonormal columns of $Y(x) \in \mathfrak{R}^{n \times m}$ form a basis for the column space of $S^2(x)\nabla G(x)W(x)G(x)$ and $R(x)$ is an $(m \times m)$ upper triangular matrix. It is easy to see that, $Y(x)^T Y(x) = I_m$, $Z(x)^T Z(x) = I_{n-m}$, and $Y(x)^T Y(x) + Z(x)^T Z(x) = I_n$. The matrix $R(x)$ is nonsingular, if x lies in a sufficiently small neighborhood of x_* , and the matrix $S^2(x)\nabla G(x)W(x)G(x)$ has full column rank at x_* .

Using a continuous differentiable null space matrix $Z(x)$ which is constructed in Goodman [15], the first order necessary condition for a feasible point x_* with respect to the box constraint (2.11), to be solution for Problem (2.1) can be written in the form

$$Z(x)^T S^2(x)\nabla f(x) = 0. \quad (3.3)$$

Applying Newton's method on the system (3.3), then we have

$$[Z(x)^T [S^2(x)\nabla f(x)]' + [Z(x)']^T S^2(x)\nabla f(x)]\Delta x = -Z(x)^T S^2(x)\nabla f(x). \quad (3.4)$$

To compute $Z(x)'$, we differentiate the equation (3.1). This gives that

$$Z(x)^T [S^2(x)\nabla G(x)W(x)G(x)]' + [Z(x)']^T S^2(x)\nabla G(x)W(x)G(x) = 0. \quad (3.5)$$

From (3.2) and (3.5), we have

$$[Z(x)']^T Y(x) = -Z(x)^T [S^2(x)\nabla G(x)W(x)G(x)]' R(x)^{-1}.$$

Since $Y(x)^T Y(x) = I_m$, then we have

$$[Z(x)']^T = -Z(x)^T [S^2(x)\nabla G(x)W(x)G(x)]' R(x)^{-1} Y(x)^T.$$

Multiplying both side of the above system from the right in $S^2(x)\nabla f(x)$ and take

$$\rho = -R(x)^{-1} Y(x)^T S^2(x)\nabla f(x), \quad (3.6)$$

then, we have

$$\begin{aligned} [Z(x)']^T S^2(x)\nabla f(x) &= \rho Z(x)^T [S^2(x)\nabla G(x)W(x)G(x)]' \\ &= \rho Z(x)^T [S^2(x)\nabla G(x)W(x)\nabla G(x)^T + \text{diag}(\nabla G(x)W(x)G(x))\text{diag}(\eta(x))]. \end{aligned} \quad (3.7)$$

Since

$$Z(x)^T [S^2(x)\nabla f(x)]' = Z(x)^T [S^2(x)\nabla^2 f(x) + \text{diag}(\nabla f(x))\text{diag}(\eta(x))], \quad (3.8)$$

then from (3.4), (3.7), and (3.8), we have

$$Z(x)^T [S^2(x)\nabla^2 \phi(x; \rho) + \text{diag}(\nabla \phi(x; \rho))\text{diag}(\eta(x))] \Delta x = -Z(x)^T S^2(x)\nabla f(x). \quad (3.9)$$

Hence the Newton step is computed by solving the system (3.9). A detailed description of the main steps of the active set Newton's interior point algorithm for solving Problem (2.1) is presented in the following section.

4. AN INTERIOR-POINT METHOD

Once the step Δx_k is computed by solving the system (3.9) at any iteration k at which $\alpha \leq x_k \leq \beta$, a damping parameter ξ_k is needed to ensure that x_{k+1} is feasible with respect to the box constraint (2.11). The damping parameter ξ_k is defined to be

$$\xi_k = \min\{1, \min_j \{u_k^{(j)}, v_k^{(j)}\}\}, \quad (4.1)$$

where

$$u_k^{(j)} = \begin{cases} \frac{\alpha^{(j)} - x_k^{(j)}}{\Delta x_k^{(j)}}, & \text{if } \alpha^{(j)} > -\infty \text{ and } \Delta x_k^{(j)} < 0, \\ 1, & \text{otherwise,} \end{cases}$$

and

$$v_k^{(j)} = \begin{cases} \frac{\beta^{(j)} - x_k^{(j)}}{\Delta x_k^{(j)}}, & \text{if } \beta^{(j)} < \infty \text{ and } \Delta x_k^{(j)} > 0, \\ 1, & \text{otherwise.} \end{cases}$$

Since, we always require $\{x_k\}$ satisfy, $\alpha < x_k < \beta$ for all k , then another damping in the step may be needed to satisfy this inequality. Therefore, we set $x_{k+1} = x_k + \delta_k \xi_k \Delta x_k$, where δ_k is defined to be 1 if $\alpha < x_k + \xi_k \Delta x_k < \beta$. Otherwise, we choose $\delta_k \in [1 - \gamma \|\Delta x_k\|, 1]$, where $\gamma > 0$ is pre-specified fixed constant.

We outline the active set Newton's interior-point algorithm for solving Problem(2.1)

Algorithm 4.1. (An active set Newton's interior-point algorithm)

A formal description of the active set Newton's interior-point algorithm for solving Problem (2.1) is presented in the following

Step 0. (Initialization)

Given $x_0 \in \mathbb{R}^n$ such that $\alpha < x_0 < \beta$.
Evaluate ρ_0, W_0, S_0 , and η_0 . Set $k = 0$.
Choose ε such that $\varepsilon > 0$.

Step1. (Test for convergence)

If $\|Z_k^T S_k^2 \nabla f_k\| \leq \varepsilon$, then terminate the algorithm.

Step2. (Compute the Newton step Δx_k)

Compute Newton's step Δx_k by solving the system(3.9).

Step3. (Test for interior-point)

- a) Compute the damping parameter ξ_k using (4.1).
 b) Set $x_{k+1} = x_k + \xi_k \Delta x_k$.
 c) If $\alpha < x_{k+1} < \beta$, then go to step 4.
 Else, set $x_{k+1} = x_k + \delta_k \xi_k \Delta x_k$, where $\delta_k \in [1 - \gamma \|\Delta x_k\|, 1]$,

Step 4. (Compute the parameter ρ_{k+1})

Compute ρ_{k+1} by solving (3.6).

Step 5. (Update the active set)

Compute W_{k+1} .

Step 6. (Update the scaling matrix)

Compute $S(x_{k+1})$ and ξ_{k+1} .

Step 7. Set $k = k + 1$ and go to Step 1.

5. IMPLEMENTATIONS

To demonstrate the effectiveness of the active set Newton's interior-point algorithm, three test examples with linear constraints, nonlinear constraints, and non-convex constraints, particularly on test problem of team is large size. These test problems was already solved by researchers [[18], [22], [19]] who have used other approaches to evaluate the performance of our suggested algorithm. We will present a comparison between their results and our results in Table (6.1). The proposed algorithm is coded in MATLAB environment and run under MATLAB Version 7 with machine epsilon about 10^{-16} .

Successful termination with respect to the proposed algorithm means that the termination condition of the algorithm is met with $\varepsilon = 10^{-10}$. On the other hand, unsuccessful termination means that the number of iterations is greater than 500, the number of function evaluations is greater than 1000.

5.1. Test Problem 1. This problem has 13 variables and nine inequality constraints [18]

$$\begin{aligned}
 & \text{minimize} && f_1(x) = 5 \sum_{i=1}^4 x_i - 5 \sum_{i=1}^4 x_i^2 - \sum_{i=5}^{13} x_i \\
 & \text{subject to} && g_1(x) = 2x_1 + 2x_2 + x_{10} + x_{11} \leq 10, \\
 & && g_2(x) = 2x_1 + 2x_3 + x_{10} + x_{12} \leq 10, \\
 & && g_3(x) = 2x_2 + 2x_3 + x_{11} + x_{12} \leq 10, \\
 & && g_4(x) = -8x_1 + x_{10} \leq 0, \\
 & && g_5(x) = -8x_2 + x_{11} \leq 0, \\
 & && g_6(x) = -8x_3 + x_{12} \leq 0, \\
 & && g_7(x) = -2x_4 - x_5 + x_{10} \leq 0, \\
 & && g_8(x) = -2x_6 - x_7 + x_{11} \leq 0, \\
 & && g_9(x) = -2x_8 - x_9 + x_{12} \leq 0, \\
 & && 0 \leq x_i \leq 1, \quad i = 1, \dots, 9, \\
 & && 0 \leq x_i \leq 100, \quad i = 10, 11, 12, \\
 & && 0 \leq x_{13} \leq 1.
 \end{aligned}$$

Notes that the convexity of the feasible domain is satisfies due the linearity of the constraint which usually require smaller effort than the non convex problems because in non convex problem the feasible domain (search space) may be disjoint or irregular, but the

proposed algorithm is capable to handle such problem, as we will introduce in the next example.

By using algorithm (4.1) the global solution of the Test Problem 1 is the same solution which is obtained by algorithm [18] such that $x^* = (1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 1)$ with a function value equal to $f_1^* = -15$.

5.2. Test Problem 2. We first choose two-dimensional general constrained optimization problem [22]

$$\begin{aligned} \text{minimize} \quad & f_2(x) = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2 \\ \text{subject to} \quad & g_1(x) = x_1^2 + (x_2 - 2.5)^2 - 4.84 \leq 0, \\ & g_2(x) = 4.84 - (x_1 - 0.05)^2 - (x_2 - 2.5)^2 \leq 0, \\ & 0 \leq x_i \leq 6, \quad i = 1, 2. \end{aligned}$$

By using algorithm (4.1) the optimum solution of the Test Problem 2 is $x^* = (2.2483, 2.3803)$ with a function value equal to $f_2^* = 13.545$ while algorithm [22] introduced an approach handling the same problem with an optimum solution $x^* = (2.246826, 2.381865)$ with a function value equal to $f_2^* = 13.59085$.

5.3. Test Problem 3. This problem has five variables and six inequality constraints [19].

$$\begin{aligned} \text{minimize} \quad & f_3(x) = 37.293239x_1 + 0.8356891x_1x_5 + 5.3578547x_3^2 - 40792.141 \\ \text{subject to} \quad & g_1(x) = 0.0022053x_3x_5 - 0.0006262x_1x_4 - 0.0056858x_2x_5 - 85.334407 \leq 0, \\ & g_2(x) = -0.0022053x_3x_5 + 0.0006262x_1x_4 + 0.0056858x_2x_5 + 85.334407 \leq 92, \\ & g_3(x) = -80.51249 - 0.0071317x_2x_5 - 0.0029955x_1x_2 - 0.0021813x_3^2 \leq -90, \\ & g_4(x) = 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 + 0.0021813x_3^2 \leq 110, \\ & g_5(x) = -9.300961 - 0.0047026x_3x_5 - 0.0012547x_1x_3 - 0.0019085x_3x_4 \leq -20, \\ & g_6(x) = 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_4 \leq 25, \\ & 78 \leq x_1 \leq 102, \\ & 33 \leq x_2 \leq 45, \\ & 27 \leq x_i \leq 45, \quad i = 3, 4, 5. \end{aligned}$$

By using algorithm (4.1) the optimum solution of the Test Problem 3 is $x^* = (78, 33, 29.9, 45, 36.823)$ with a function value equal to $f_3^* = -30693$ while [19] introduced an approach handling the same problem with an optimum solution $x^* = (78, 33, 29.995, 45, 36.776)$ with a function value equal to $f_3^* = -30665.5$.

Test problem	algorithm	solution
Test problem 1	Michalewicz and Attia [18] new algorithm	$f_1^* = -15$ $f_1^* = -15$
Test problem 2	Powell and Skolnick [22] new algorithm	$f_2^* = 13.59085$ $f_2^* = 13.545$
Test problem 3	Michalewicz and Schoenauer, [19] new algorithm	$f_3^* = -30665.5$ $f_3^* = -30693$

TABLE 6.1.

6. CONCLUDING REMARK

We introduced the active set Newton's interior point algorithm for solving the constrained optimization problem. The proposed algorithm uses active-set strategy to convert the inequality constrain to equality constrain and the Coleman-Li scaling matrix together with a formulation for Newton's interior point method are used to solve the equality constraint problem. To study the performance of this algorithm, we have considered three test problems. The results were reported. We believe that our approach is of value and merit further numerical investigations.

The following points are the signification contributions of this paper

- The proposed approach combines Coleman-Li scaling matrix together with Newton's interior point method to solve the equality constraint problem.
- One of the disadvantages of obtaining Newton's step by solving the linear system lies in the fact that the dimension of the linear system is large for large-scale problems. In this algorithm, we present a reduced method that computes Newton's step by solving a smaller dimension linear system.
- For future work, there are many question should be answered
 - Improving the proposed algorithm to be capable for treating large scale non-linear programming problems and non differentiable case.
 - We have to impalement the proposed algorithm on real life problem.

REFERENCES

- [1] T. R. COLEMAN and Y. LI An interior trust region approach for nonlinear minimization subject to bounds, *SIAM J. Optimization*, 6(1996), 418-445.
- [2] I. Das, An interior point algorithm for the general nonlinear programming problem, Department of computational and applied mathematics, Rice University, Houston, TX 77251-1892,(1996).
- [3] J. Dennis, M. El-Alem, and M. Maciel, A global convergence theory for general trust-region-based algorithms for equality constrained optimization, *SIAM J Optimization*, 7(1)(1997), 177-207.
- [4] J. Dennis, M. El-Alem, and K. Williamson, A trust-region approach to nonlinear systems of equalities and inequalities, *SIAM J Optimization*, 9(1999), 291-315.
- [5] J. Dennis, M. Heinkenschloss, and L. Vicente. Trust-region interior-point SQP algorithms for a class of nonlinear programming problems. Tech. Rep. TR94-45, Department of Computational and Applied Mathematics, Rice University, Houston, TX 77251 (1994).
- [6] M. El-Alem, M. El- Sayed, and B. El-Sobky, Local Convergence of Interior - point Newton Method for Constrained Optimization, *Journal of Optimization theory and applications*, 120(3)(2004), 487 - 502.
- [7] M.El-Alem, B. El- sobky, and M. Abdel -Aziz, Extending Goodman's Method to General Nonlinear Programming, *Bulletin pure and applied mathematics*, 3(2)(2009), 135-141.
- [8] B. El-Sobky, Arobust trust-region algorithm for general nonlinear constrained optimization problems, PhD thesis, Department of Mathematics, Alexandria University, Alexandria, Egypt(1998).
- [9] B. El-Sobky, A global convergence theory for an active trust region algorithm for solving the general nonlinear programming problem, *Applied Mathematics and computation archive*, 144(1)(2003), 127-157.
- [10] B. El-Sobky, A Multiplier active-set trust-region algorithm for solving constrained optimization problem, *Applied Mathematics and computation archive*, 219(2012), 127-157.
- [11] B. El-Sobky, A New Convergence theory for Trust-Region Algorithm for Solving Constrained Optimization Problems , *Applied Mathematics Science*, 7(110)(2013), 5469-5489.
- [12] B. El-Sobky, A penalty active-set trust-region algorithm for solving general nonlinear programming problem , *Sylwan*, 9(158)(2014), 273-290.
- [13] B.El-sobky and Y.Abouel-Naga, Multi-objective Economic Emission Load Dispatch Problem with Trust-region Strategy, *Electric Power Systems Research*, 108(2014), 254-269.
- [14] P. Gill, W. Murray, M. Saunders, and M. Wright, Some theoretical properties of an augmented Lagrangian merit function, Stanford University, report SOL (1986), 86-6.
- [15] J. GOODMAN, Newton's method for constrained optimization, *Math Programming*, 33(1985), 162-171.
- [16] S. Han. A globally convergent method for nonlinear programming. *JOTA*, 22:297-309, (1977).
- [17] D. Himmelblau, *Applied nonlinear programming*, New York: McGraw-Hill, (1972).
- [18] Z. Michalewicz and N. Attia, Evolutionary optimization of constrained problems, *Proc, Third Annual Conference on Evolutionary Programming*, (1994), 98?108.
- [19] Z. Michalewicz and M. Schoenauer, Evolutionary algorithms for constrained parameter optimization problems, *Evolutionary Computation*, 4(1)(1996), 1?32.
- [20] E. Omojokun, Trust-region strategies for optimization with nonlinear equality and inequality constraints, PhD thesis, Department of Computer Science, University of Colorado, Boulder, Colorado(1989).
- [21] M. Powell. Algorithms for nonlinear constraints that used Lagrangian functions. *Mathematical Programming*, North-Holland Publishing Company, 14(1978), 224-248.
- [22] D. Powell and M. Skolnick, Using genetic algorithms in engineering design optimization with nonlinear constraints, *Proc, Fifth International Conference on Genetic Algorithms*, Urbana, IL, (1993), 424?430.
- [23] K. Schittkowski, On the convergence of a sequential quadratic programming method with an augmented Lagrangian line search function, *Math. Operations Forchung U.Statistic Ser., Optimization*, 14(1983), 197-216.
- [24] Yuan. Y, On the convergence of a new trust region algorithm. *Numer. Math.* 70(1995), 515-539.

BOTHINA EL-SOBKY

FACULTY OF SCIENCE, ALEXANDRIA UNIVERSITY, ALEXANDRIA, EGYPT

E-mail address: bothinaelsobky@yahoo.com

YUSRIA ABO-ELNAGA

DEPARTMENT OF BASIC SCIENCE, HIGHER TECHNOLOGICAL INSTITUTE, TENTH OF RAMADAN CITY , EGYPT

E-mail address: yousria_naga@yahoo.com