# Evolutionary Method for Nonlinear Systems of Equations

Crina Grosan[1], Ajith Abraham[2], and Alexander Gelbukh[3]

[1] Department of Computer Science
Babeş-Bolyai University, Cluj-Napoca, 3400, Romania
[2] IITA Professorship Program, School of Computer Science and Engineering, Yonsei University, 134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea
[3] Centro de Investigación en Computación (CIC)
Instituto Politécnico Nacional (IPN), Mexico
cgrosan@cs.ubbcluj.ro, ajith.abraham@ieee.org, gelbukh@gelbukh.com

**Abstract.** We propose a new perspective for solving systems of nonlinear equations by viewing them as a multiobjective optimization problem where every equation represents an objective function whose goal is to minimize the difference between the right- and left-hand side of the corresponding equation of the system. An evolutionary computation technique is suggested to solve the problem obtained by transforming the system into a multiobjective optimization problem. Results obtained are compared with some of the well-established techniques used for solving nonlinear equation systems.

## 1 Introduction

A nonlinear system of equations is defined as:

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_n(x) \end{bmatrix},$$

$x = (x_1, x_2, \ldots, x_n)$, which means there are $n$ equations and $n$ variables, where $f_1, \ldots, f_n$ are nonlinear functions in the space of all real valued continuous functions on $\Omega = \prod_{i=1}^{n} [a_i, b_i] \subset \Re^n$.

Some of the equations can be linear, but not all of them. Finding a solution for a nonlinear system of equations $f(x)$ involves finding a solution such that every equation in the nonlinear system is 0:

$$(P) \begin{cases} f_1(x_1,\, x_2, \ldots,\, x_n) = 0 \\ f_2(x_1,\, x_2, \ldots,\, x_n) = 0 \\ \vdots \\ f_n(x_1,\, x_2, \ldots,\, x_n) = 0 \end{cases}. \tag{1}$$

A. Gelbukh and C.A. Reyes-Garcia (Eds.): MICAI 2006, LNAI 4293, pp. 283–293, 2006.

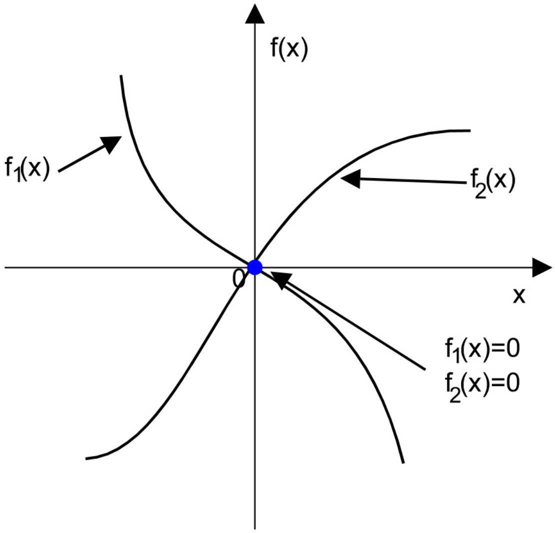

**Fig. 1.** Example of solution in the case of a two nonlinear equation system represented by $f_1$ and $f_2$

In Figure 1, the solution for a system having two nonlinear equations is depicted. There are also situations when a system of equations has multiple solutions. For instance, the system:

$$\begin{cases} f_1(x_1, x_2, x_3, x_4) = x_1^2 + 2x_2^2 + cos(x_3) - x_4^2 = 0 \\ f_1(x_1, x_2, x_3, x_4) = 3x_1^2 + x_2^2 + sin^2(x_3) - x_4^2 = 0 \\ f_1(x_1, x_2, x_3, x_4) = -2x_1^2 - x_2^2 - cos(x_3) + x_4^2 = 0 \\ f_1(x_1, x_2, x_3, x_4) = -x_1^2 - x_2^2 - cos^2(x_3) + x_4^2 = 0 \end{cases}$$

has two solutions: (1, -1, 0, 2) and (-1, 1, 0, -2). The assumption is that a zero, or root, of the system exists. The solutions of interest are those points (if any) that are common to the zero contours of $f_i, i = 1, ..., n$. There are several ways to solve nonlinear equation systems ([1], [5]-[9] and [13]). Probably the Newton type is one of the most established techniques. Other methods are depicted as follows:

- Trust-region method [3];
- Broyden method [2];
- Secant method [12];
- Halley method [4].

***Newton's method.*** In Newton's method, $f$ is approximated by the first order Taylor expansion in a neighborhood of a point x$^k \in \Re^n$. The Jacobian matrix $J(x^k) \subset \Re^{nxn}$ to $f(x)$ evaluated at $x^k$ is given by:

$$J = \begin{bmatrix} \frac{\delta f_1}{\delta x_1} \cdots \frac{\delta f_1}{\delta x_n} \\ \vdots \qquad \vdots \\ \frac{\delta f_n}{\delta x_1} \cdots \frac{\delta f_n}{\delta x_n} \end{bmatrix}.$$

Then:

$$f(x^k + t) = f(x^k) + J(x^k)t + O(||p||^2).$$

By setting the right side of the equation to zero and discarding terms higher than first order ($O(||p||^2)$) the relationship $J(x^k)t = -f(x^k)$ is obtained. Then, the Newton algorithm is described as follows:

**Algorithm 1.** Newton algorithm

```
Set k=0.
Guess an approximate solution x$^0$.
Repeat
    Compute J(x$^k$) and f(x$^k$).
    Solve the linear system $J(x^k)t = -f(x^k)$.
    Set $x^{k+1} = x^k + t$.
    Set t = t + 1.
Until converge to the solution
```

The index $k$ is an iteration index and $x^k$ is the vector $x$ after $k$ iterations. The idea of the method is to start with a value which is reasonably close to the true zero and then replaces the function by its tangent and computes the zero of this tangent. This zero of the tangent will typically be a better approximation to the function's zero, and the method can be iterated. This algorithm is also known as *Newton-Raphson* method. There are also several other Newton methods. It is very important to have a good starting value (the success of the algorithm depends on this). The Jacobian matrix is needed but in many problems analytic derivatives are unavailable. If function evaluation is expensive, then the cost of finite-difference determination of the Jacobian can be prohibitive.

***Broyden's method.*** The approximate Jacobian is denoted by: $\delta x = -J^{-1}f$. Then the $i$-th quasi-Newton step $\delta x_i$ is the solution of $B_i \delta xi = -f_i$, where $\delta x_i = x_{i+1}x_i$. The quasi-Newton or secant condition is that $B_{i+1}$ satisfy $B_{i+1}\delta x_i = \delta f_i$, where $\delta f_i = f_{i+1} - f_i$. This is the generalization of the one-dimensional secant approximation to the derivative $\frac{\delta f}{\delta x}$.

Many different auxiliary conditions to pin down $B_{i+1}$ have been explored, but the best-performing algorithm in practice results from Broyden's formula. This formula is based on the idea of getting $B_{i+1}$ by making the least change to $B_i$ consistent with the secant equation. Broyden showed that the resulting equation as:

$$B_{i+1} = B_i + \frac{(\delta f_i - B_i \delta x_i) \otimes \delta x_i}{(\delta x_i)^2}$$

***Secant method.*** The secant method is a root-finding algorithm that uses a succession of roots of secant lines to better approximate a root of a function. The secant method is defined by the recurrence relation

$$x_{n+1} = x_n - \frac{x_n - x_{n-1}}{f(x_n) - f_{(}x_{n-1})} f(x_n)$$

As evident from the recurrence relation, the secant method requires two initial values, $x_0$ and $x_1$, which should ideally be chosen to lie close to the root. As illustrated in Figure 2, two points $a$ and $b$ are initially considered. Then the secant of chord of the the graph of function $f$ through the points $(a, f(a))$, $(b, f(b))$ is defined as:

$$y - f(b) = \frac{f(b) - f(a)}{b - a}(x - b).$$

The point $c$ is chosen to be the root of this line such that:

$$f(b) + \frac{f(b) - f(a)}{b - a}(c - b) = 0.$$

Solving this equation gives the recurrence relation for the secant method. The new value $c$ is equal to $x_{n+1}$, and $b$ and $a$ are $x_n$ and $x_{n-1}$, respectively.

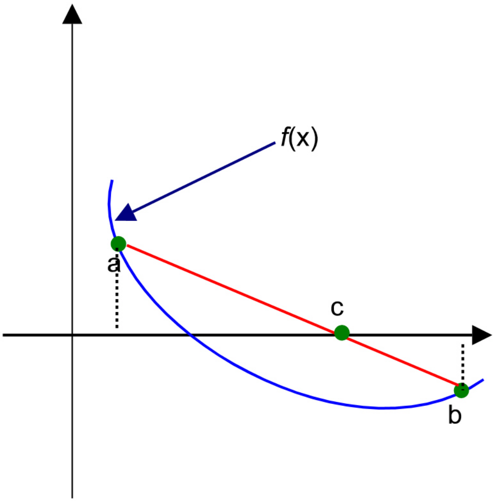

**Fig. 2.** An example of secant method

***Effati's method.*** Effati and Nazemi [10] proposed a new method for solving systems of nonlinear equations. Their proposed method [10] is summarized below. The following notation is used:

$$x_i(k+1) = f_i(x_1(k), x_2(k), ..., x_n(k));$$

$$f(x_k) = (f_1(x_k), f_2(x_k), ..., f_n(x_k));$$

$i = 1, 2..., n$ and $x_i : N \rightarrow \Re$.

If there exist a $t$ such that $x(t) = 0$ then $f_i(x(t-1)) = 0, i = 1, ..., n$. This involves that $x(t-1)$ is an exact solution for the given system of equations.

Define $u(k) = (u_1(k), u_2(k), ..., u_n(k))$, $x(k+1) = u(k)$, and $f^0 : \Omega \times U \rightarrow \Re$ ($\Omega$ and $U$ are compact subsets of $\Re^n$):

$$f^0(x(k), u(k)) = \|u(k) - f(x(k))\|_2^2.$$

The error function $E$ is defined as follows:

$$E[x^t, u^t] = \sum_{k=0}^{t-1} f^0(x(k), u(k)),$$

$$\begin{array}{c} x^t = (x(1), x(2), ..., x(t-1), 0) \\ u^t = (u(1), u(2), ..., u(t-1), 0). \end{array}$$

Consider the following problem:

$$(P_1) \begin{cases} \text{minimize} E[x^t, u^t] = \sum\limits_{k=0}^{t-1} f^0(x(k), u(k)) \\ \text{subject to} \\ x(k+1) = u(k) \\ x(0) = 0, x(t) = 0, (x^0 \text{ is known}) \end{cases} . \qquad (2)$$

As per theorem illustrated in [10], if there is an optimal solution for the problem $P_1$ such that the value of $E$ will be zero, then this is also a solution for the system of equations we want to solve. The problem is transformed to a measure theory problem. By solving the transformed problem $u^t$ is firstly constructed and from there, $x^t$ is obtained. Reader is advised to consult [10] for details. The measure theory method is improved in [10]. The interval $[1, t]$ is divided into the subintervals $S_1 = [1, t-1]$ and $S_2 = [t-1, t]$. The problem $P_1$ is solved in both subintervals and two errors $E_1$ and $E_2$ respectively are obtained. This way, an upper bound for the total error if found. If this upper bound is estimated to be zero then an approximate solution for the problem is found.

## 2 Transforming the Problem into a Multiobjective Optimization Problem

The basic definitions of a multiobjective optimization problem and what it denotes an optimal solution is formulated as follows [15]:

Let $\Omega$ be the search space. Consider $n$ objective functions $f_1$, $f_2 \ldots f_n$,

$$f_i : \Omega \rightarrow \Re, \quad i = 1, 2, \ldots, n,$$

where $\Omega \subset \Re^m$. The multiobjective optimization problem is defined as:

$$\begin{cases} \textit{optimize}\, f(x) = (f_1(x), ..., f_n(x)) \\ \textit{subject to} \\ x = (x_1,\, x_2, \ldots x_m) \in \Omega. \end{cases}$$

For deciding wether a solution is better than another solution or not, the following relationship between solutions might be used:

*Definition* 1 (*Pareto dominance*). Consider a maximization problem. Let $x$, $y$ be two decision vectors (solutions) from $\Omega$. Solution $x$ *dominates* $y$ (also written as $x \succ y$) if and only if the following conditions are fulfilled:

(*i*) $f_i(x) \geq f_i(y), \forall i = 1, 2, \ldots, n,$
(*ii*) $\exists j \in \{1,\, 2,\, \ldots,\, n\}$: $f_j(x) > f_j(y)$.

That is, a feasible vector $x$ is Pareto optimal if no feasible vector $y$ can increase some criterion without causing a simultaneous decrease in at least one other criterion. In the literature other terms have also been used instead of Pareto optimal or minimal solutions, including words such as non-dominated, non-inferior, efficient, functional-efficient solutions etc. The solution $x^0$ is *ideal* if all objectives have their optimum in a common point $x^0$.

*Definition* 2 (*Pareto front*). The images of the Pareto optimum points in the criterion space are called *Pareto front.* The system of equations ($P$) can be transformed into a multiobjective optimization problem. Each equation can be considered as an objective function. The goal of this optimization function is to minimize the difference (in absolute value) between left side and right side of the equation. Since the right term is zero, the objective function will be given by the absolute value of the left term.

The system ($P$) is then equivalent to:

$$(P') \begin{cases} \text{minimize } abs(f_1(x_1,\, x_2, ...,\, x_n)) \\ \text{minimize } abs(f_2(x_1,\, x_2, ...,\, x_n)) \\ \vdots \\ \text{minimize } abs(f_n(x_1,\, x_2, ...,\, x_n)) \end{cases}$$

## 3 Evolutionary Nonlinear Equation System

An evolutionary technique is proposed for solving the multiobjective problem obtained by transforming the system of equations. Some starting points (initial solutions) are generated based on the problem domain defined and these solutions are evolved in an iterative manner. In order to compare the two solutions, Pareto dominance relationship is used. Genetic operators such as Convex crossover and Gaussian mutation are used [11]. An external set is used for storing all the non-dominated solutions found during the iteration process. Tournament selection is applied. $n$ individuals are randomly selected from the unified set of current population and external population. Out of these $n$ solutions the one which dominated a greater number of solutions is chosen. If there are two or more 'equal' solutions then one of them is picked at random. At each iteration, this archive is updated by introducing all the non-dominated solutions obtained at the respective step and by removing from the external set of all solutions that might become dominated.

The proposed algorithm is described as follows:

**Algorithm 2.** Evolutionary Multiobjective Optimization (EMO) algorithm

```
Step 1. Set $t = 0$.
Randomly generate population $P(t)$.
Set $EP(t) = \emptyset$. ($EP$ denoted the external population).
Step 2. Repeat
            Step 2.1. Evaluate $P(t)$
            Step 2.2. Selection $(P(t) \cup EP(t))$
            Step 2.3. Crossover
            Step 2.4. Mutation
            Step 2.3. Select all nondominated individuals obtained
            Step 2.3. Update $EP(t)$
            Step 2.3. Update $(P(t)$ (keep best between parents and offspring)
            Step 2.3. Set $t := t + 1$
      Until $t = number_o f_g enerations$
Step 3. Print $EP(t)$
```

## 4 Experiments, Results, and Discussions

This section reports several experiments and comparisons. We consider the same problems (Examples 1 and 2 below) as Effati [10]. Parameter values used by the evolutionary approach are given in Table 1.

**Table 1.** Parameter values used in the experiments by the evolutionary approach

| Parameter | Value | |
|---|---|---|
| | Example 1 | Example 2 |
| Population size | 250 | 300 |
| Number of generations | 150 | 200 |
| Sigma (for mutation) | 0.1 | 0.1 |
| Tournament size | 4 | 5 |

**Example 1.** Consider the following nonlinear system:

$$\begin{cases} f_1(x_1, x_2) = cos(2x_1) - cos(2x_2) - 0.4 = 0 \\ f_2(x_1, x_2) = 2(x_2 - x_1) + sin(2x_2) - sin(2x_1) - 1.2 = 0 \end{cases} .$$

Results obtained by applying Newton's method, Effati's technique, and the proposed EMO method are presented in Table 2. A sample solution obtained by the EMO approach is presented in Table 2. More Pareto solutions and the corresponding absolute values of the functions $f_1$ and $f_2$ are presented in Table 3.

**Table 2.** Empirical results for Example 1

| Method | Solution | Function absolute values |
|---|---|---|
| Newton's method | (0.15, 0.49) | (-0.00168, 0.01497) |
| Secant method | (0.15, 0.49) | (-0.00168, 0.01497) |
| Broyden's method | (0.15, 0.49) | (-0.00168, 0.01497) |
| Effati's method | (0.1575, 0.4970) | (0.005455, 0.00739) |
| EMO approach | (0.15772, 0.49458) | (0.001264, 0.000969) |

**Table 3.** Nondominated solutions and the corresponding objectives values obtained by EMO approach for Example 1

| $x_1$ | $x_2$ | $f_1$ | $f_2$ |
|---|---|---|---|
| 0.15780 | 0.4943 | 0.00212 | 0.00075 |
| 0.1577 | 0.4945 | 0.001139 | 0.00119 |
| 0.1578 | 0.4942 | 0.000583 | 0.002439 |
| 0.1577 | 0.4943 | 0.000812 | 0.00173 |
| 0.1578 | 0.04941 | 0.000416 | 0.00274 |
| 0.15775 | 0.4945 | 0.00111 | 0.00131 |
| 0.1577 | 0.49455 | 0.00123 | 0.000964 |
| 0.1569 | 0.4942 | 0.001142 | 0.00107 |
| 0.1568 | 0.4941 | 0.001035 | 0.00115 |
| 0.1570 | 0.4942 | 0.001078 | 0.000681 |

In Figure 3, the Pareto front obtained by the solutions presented in the Table 3 are presented. As evident from Figure 3, all the solutions plotted are nondominated. The user can select the desired solution taking into account of the different preferences (for instance, the one for which one objective is having a value closed to the desired value, or the one for which the sum of both objectives is minimal, etc). As illustrated in Table 2, results obtained by the evolutionary approach are better than the ones obtained by the other techniques. Also, by applying an evolutionary technique we don't need any additional information about the problem (such as the functions to be differentiable, a good starting point, etc).

**Example 2.** We consider the following problem:

$$\begin{cases} f_1(x_1, x_2) = e^{x_1} + x_1 x_2 - 1 = 0 \\ f_2(x_1, x_2) = sin(x_1 x_2) + x_1 + x_2 - 1 = 0 \end{cases} .$$

Results obtained by Effati's method and one solution obtained by the EMO approach are given in Table 4.

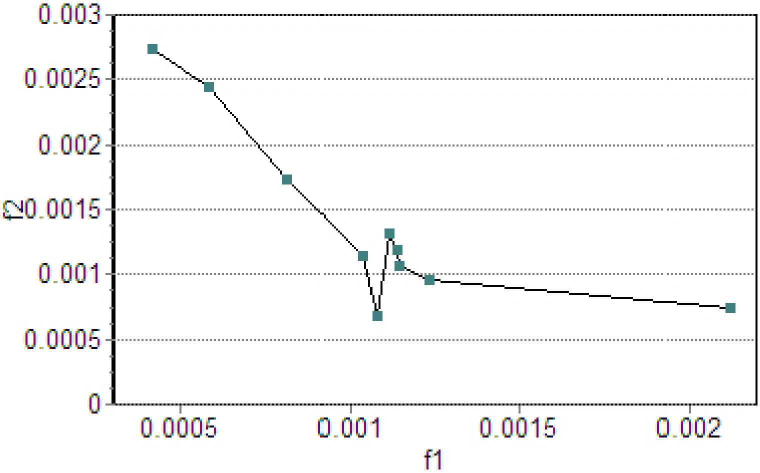

**Fig. 3.** Pareto front obtained by the EMO approach for the Example 1

**Table 4.** Empirical results for Example 2

| Method | Solution | Function absolute values |
|---|---|---|
| Effati | (0.0096, 0.9976) | (0.019223, 0.016776) |
| EMO approach | (0.00138, 1.0027) | (0.00276, 6.37E-5) |

**Table 5.** Nondominated solutions and the corresponding objectives values obtained by EMO approach for Example 2

| $x_1$ | $x_2$ | $f_1$ | $f_2$ |
|---|---|---|---|
| 0.00130 | 1.0025 | 0.00260 | 0.00510 |
| 0.0011 | 0.0030 | 0.00220 | 0.00520 |
| 0.0012 | 1.0020 | 0.002403 | 0.00440 |
| 0.0004 | 1.0023 | 0.000801 | 0.00310 |
| 0.0003 | 1.0028 | 0.000600 | 0.00340 |
| 0.00028 | 1.0029 | 0.000560 | 0.00346 |
| 0.00025 | 1.004 | 0.000501 | 0.00450 |
| 0.0015 | 1.0043 | 0.0003006 | 0.00460 |
| 0.0017 | 1.0041 | 0.000340 | 0.00444 |
| 0.0001 | 1.005 | 0.0002005 | 0.00520 |

The nondominated solutions and the corresponding functions values obtained by EMO approach are presented in Table 5. Pareto front obtained by the EMO method for Example 2 are depicted in Figure 4. For this example, the evolutionary approach obtained better results than the results reported by Effati's method. These experiments show the efficiency and advantage of applying

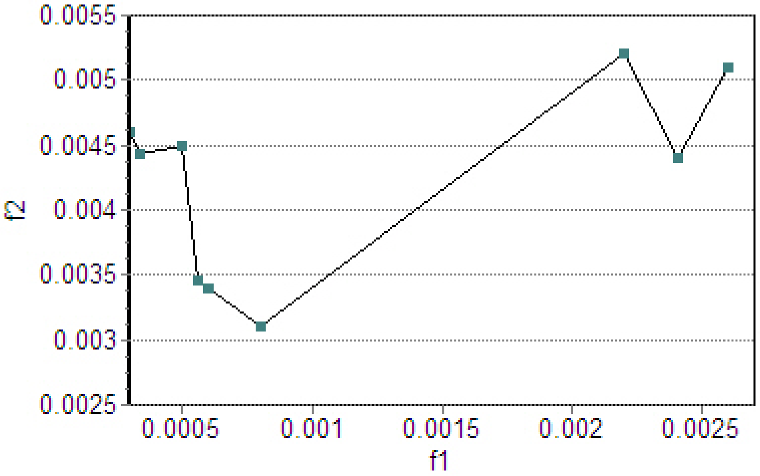

**Fig. 4.** Pareto front obtained by the EMO approach for the Example 2

evolutionary techniques for solving systems of nonlinear equations against standard mathematical approaches.

## 5 Conclusions

The proposed approach seems to be very efficient for solving equation systems. In this paper, we analyzed a case of nonlinear equation systems. The proposed approach could be extended and applied for higher dimensional systems. In a similar manner, inequations systems could be also solved.

## Acknowledgments

The second author acknowledges the support received from the International Joint Research Grant of the IITA (Institute of Information Technology Assessment) foreign professor invitation program of the Ministry of Information and Communication, South Korea.

## References

1. C. Brezinski, Projection methods for systems of equations, Elsevier, 1997.
2. C.G. Broyden, A class of methods for solving nonlinear simultaneous equations. Mathematics of Computation, 19, 577-593, 1965.
3. A.R. Conn, N.I.M. Gould, P.L. Toint, Trust-Region methods, SIAM, 2000.
4. A. Cuyt, P.van der Cruyssen, Abstract Pade approximants for the solution of a system of nonlinear equations, Comp. Math. and Appl., 9, 139-149, 1983.

5. J.E. Denis, On Newtons Method and Nonlinear Simultaneous Replacements, SIAM Journal of Numerical Analisys, 4, 103108, 1967.
6. J.E. Denis, On Newtonlike Methods, Numerical Mathematics, 11 , 324330, 1968.
7. J.E. Denis, On the Convergence of Broydens Method for Nonlinear Systems of Equations, Mathematics of Computation, 25, 559567, 1971.
8. J.E. Denis, H. Wolkowicz, LeastChange Secant Methods, Sizing, and Shifting, SIAM Journal of Numerical Analisys, 30, 12911314, 1993.
9. J.E. Denis, M. ElAlem, K. Williamson, A Trust-Region Algorithm for Least-Squares Solutions of Nonlinear Systems of Equalities and Inequalities, SIAM Journal on Optimization 9(2), 291-315, 1999.
10. S. Effati, A.R. Nazemi, A new methid for solving a system of the nonlinear equations, Applied Mathematics and Computation, 168, 877-894, 2005
11. Goldberg, D.E. Genetic algorithms in search, optimization and machine learning. Addison Wesley, Reading, MA, 1989.
12. W. Gragg, G. Stewart, A stable variant of the secant method for solving nonlinear equations, SIAM Journal of Numerical Analisys, 13, 889-903, 1976.
13. J. M. Ortega and W. C. Rheinboldt, Iterative solution of nonlinear equations in several variables. New York: Academic Press, 1970
14. W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes in C: The Art of Scientific Computing, Cambridge University Press, 2002.
15. Steuer, R. E. Multiple Criteria Optimization. Theory, Computation, and Application. Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics. New York: John Wiley & Sons, Inc, 1986.