Generation of Pareto optimal solutions for multi-objective optimization problems via a reduced interior-point algorithm

B. El-Sobky
bothinaelsobky@yahoo.com*
Y. Abo-Elnaga †
yousria-naga@yahoo.com

November 11, 2017

Abstract

In this paper, a reduced interior-point algorithm is introduced to generate a Pareto optimal front for multi-objective constrained optimization (MOCP) problem. A weighted Tchebychev metric (WTM) approach is used together with achievement secularizing function approach to convert (MOCP) problem to a single-objective constrained optimization(SOCO) problem. An active-set technique is used together with a Coleman-Li scaling matrix to find the solution of (SOCO)problem. A decrease interior-point method is used to compute Newtons step by solving a smaller dimension system.

A Matlab implementation of the proposed algorithm was used to solve three cases and application. The results showed that the algorithm out perform some existing methods in literature. The results, by using our suggested approach to benchmark problems are promising when compared with well-known algorithms. Also, our outcomes recommend that our calculation may be superior relevant for comprehending real-world application problems.

Key Words: Multi-objective optimization problem, weighted Tchebychev metric approach , active set, Coleman-Li scaling matrix, interior-point method.

MSC 2010 : 49N35, 49N10, 93D52, 49D37, 65K05.

1 Introduction

A wide variety of problems in engineering, industry, and many other fields, involve a multi-objective optimization problems (MOPs). We consider in this paper on the

[1]Department of Mathematics and Computer Science, Faculty of Science, Alexandria University, Alexandria, Egypt.

[2]Department of basic science, Higher Technological Institute, Tenth of Ramadan City , Egypt.

following multi-objective constrained optimization (MOCP) problem

$$\begin{array}{ll} minimize & [f_1(x), \cdots f_q(x)]^T \\ subject\ to & H_e(x) \;=\; 0, \quad e = 1, \cdots, m, \\ & H_i(x) \;\leq\; 0, \quad i = 1, \cdots, p, \\ & \alpha \leq x \leq \beta, \end{array} \tag{1.1}$$

where $\alpha \in \{\ \Re \bigcup \{-\infty\}\ \}^n$, $\beta \in \{\ \Re \bigcup \{\infty\}\ \}^n$, $m < n$, and $\alpha < \beta$. The functions $f_1, \cdots, f_q$, H_e, and H_i are assumed to be at least twice continuously differentiable. Let $\tilde{\boldsymbol{F}} = \{x : \alpha \leq x \leq \beta\}$ and $int(\tilde{\boldsymbol{F}}) = \{x : \alpha < x < \beta\}$.

In (MOCP) problem there is more than one objective function and there is no single optimal solution that at the same time improves all the objective functions. The idea of optimality is supplanted with that of Pareto optimality or effectiveness in (MOCP) problem. The Pareto optimal (PO) solutions are the solutions that can't be progressed in one objective function without breaking down their execution in at any rate one of the rest because of the confliction of the objectives. The decision maker (DM) is searching for the most favored solution among the PO solutions of (MOCP) problem.

There are a large variety of methods for accomplishing (MOCP) problem. None of them can be said to be generally superior to all of the others. Wang and Zhou [25] classified these methods to posteriori, no preference, and priori techniques according to the participation of the DM in the solution process.

At this paper we intrigue with the posteriori techniques, where the DM takes a section in the solution procedure. A posteriori techniques can be called strategies for producing PO solutions. Since, there are infinitely many PO solutions, then the DM chooses the most favored one from the PO set.

The weighted Tchebychev metric approach which is introduced by [19] is one of the posteriori methods. It is utilized to convert (MOCP) problem to the single-objective constrained optimization (SOCO) problem by minimizing the distance between the ideal objective vector and the feasible objective region. In any case, if the global ideal objective vector is obscure, we may bomb in delivering PO solutions. In the other word on the off chance that the reference point utilized as a part of the objective vector inside the feasible objective region, the insignificant separation to it is zero and we don't get the PO solution.

So, we will use the achievement secularizing function approach [19] which is a special type of secularizing functions. It is used to overcome the weakness of the weighted Tchebychev metric approach by replacing metrics by achievement secularizing function, and the PO solutions can be generated with any reference point. Using this approach, the (MOCP) Problem (1.1) transforms to the following (SOCO) problem

$$\begin{array}{ll} minimize & maximize_{j=1,\ldots,q}\{\omega_i(f_j(x) - \tilde{y}_j)\} \\ subject\ to & H_e(x) \;=\; 0, \quad e = 1, \cdots, m, \\ & H_i(x) \;\leq\; 0, \quad i = 1, \cdots, p, \\ & \alpha \leq x \leq \beta, \end{array} \tag{1.2}$$

where $\sum_{j=1}^{q} \omega_j = 1$, $\omega_j \geq 0$, and $\tilde{y}_j$ is the reference point for all $j = 1, ..., q$.

Problem (1.2) can be rewritten as follows

$$\begin{array}{lll} minimize & F(x) & \\ subject\ to & H_e(x) = 0, & e = 1, \cdots, m, \\ & H_i(x) \leq 0, & i = 1, \cdots, p, \\ & \alpha \leq x \leq \beta, & \end{array} \tag{1.3}$$

where $F(x) = maximize_{j=1,\ldots,q}\{\omega_(f_j(x) - \tilde{y}_j)\}$. To convert the above problem to an equality constrained optimization(EQCO) problem with bound on the variables, we use an active-set strategy which is introduced in [8]. Many authors have used active-set algorithms to solve the general (SOCO) problems. For example, see [[8], [12], [13], [14]].

Using the active-set mechanism in [8], we have a diagonal matrix $W(x_k)$, whose diagonal elements are defined as follows

$$w_i(x) = \begin{cases} 1, & \text{if } H_i(x) \geq 0, \\ 0, & \text{if } H_i(x) < 0. \end{cases} \tag{1.4}$$

The above matrix utilizes to transform Problem (1.3) to the following bounded (EQCO) problem

$$\begin{array}{ll} minimize & F(x) + \frac{r}{2}\|W(x)H_i(x)\|^2 \\ subject\ to & H_e(x) = 0, \\ & \alpha \leq x \leq \beta, \end{array} \tag{1.5}$$

where r is positive parameter. The augmented Lagrangian function associated with Problem (1.5) without the bounded constraint $\alpha \leq x \leq \beta$ is defined as follows

$$\ell(x, \lambda; r) = F(x) + \lambda^T H_e(x) + \frac{r}{2}\|W(x)H_i(x)\|^2,$$

where λ represent the Lagrange multiplier vector associated with the equality constraint $H_e(x)$.

The augmented Lagrangian function associated with Problem (1.5) is given by

$$L(x, \lambda, \mu, \nu; r) = \ell(x, \lambda; r) - \mu^T(x - \alpha) - \nu^T(\beta - x), \tag{1.6}$$

where μ and ν are Lagrange multiplier vectors associated with inequality constraints $(x - \alpha)$ and $(\beta - x)$ respectively.

The first-order necessary conditions for a point x_* to be a solution of Problem (1.5) are the existence of multiplier vectors λ_*, μ_*, and ν_* such that the point $(x_*, \lambda_*, \mu_*, \nu_*)$ satisfies

$$\nabla_x \ell(x_*, \lambda_*; r_*) - \mu_* + \nu_* = 0, \tag{1.7}$$

$$H_e(x_*) = 0, \tag{1.8}$$

$$\alpha \leq x_* \leq \beta, \tag{1.9}$$

$$\mu_*^{(b)}(x_*^{(b)} - \alpha^{(b)}) = 0, \tag{1.10}$$

$$\nu_*^{(b)}(\beta^{(b)} - x_*^{(b)}) = 0, \tag{1.11}$$

for all b corresponding to $x^{(b)}$ with finite bound and

$$\nabla_x \ell(x_*, \lambda_*; r_*) = \nabla F(x_*) + \nabla H_e(x_*)\lambda_* + r_* \nabla H_i(x_*) W(x_*) H_i(x_*).$$

Using the scaling diagonal matrix $A(x)$ whose diagonal elements are defined as follows

$$a^{(b)}(x) = \begin{cases} \sqrt{(x^{(b)} - \alpha^{(b)})}, & \text{if } \nabla_x \ell(x, \lambda; r)^{(b)} \geq 0 \text{ and } \alpha^{(b)} > -\infty, \\ \sqrt{(\beta^{(b)} - x^{(b)})}, & \text{if } \nabla_x \ell(x, \lambda; r)^{(b)} < 0 \text{ and } \beta^{(b)} < +\infty \ , \\ 1, & \text{otherwise,} \end{cases} \tag{1.12}$$

the system (1.7-1.11) converted to the following system

$$A^2(x)\nabla_x \ell(x, \lambda; r) = 0, \tag{1.13}$$

$$H_e(x) = 0, \tag{1.14}$$

where $x \in (\tilde{\boldsymbol{F}})$. For more details about the scaling matrix $A(x)$ see [[4], [9], [7]]. Notice that the above system is not everywhere differentiable but it is continuous. More details about the derivation of the above system is given in [10].

In the rest of the paper, we use the symbol $F_k \equiv F(x_k)$, $H_{e_k} \equiv H_e(x_k)$, $H_{i_k} \equiv H_i(x_k)$, $W_k \equiv W(x_k)$, $A_k \equiv A(x_k)$ $\ell_k \equiv \ell(x_k, \lambda_k; r_k)$, $\nabla_x \ell_k \equiv \nabla_x \ell(x_k, \lambda_k; r_k)$, $\tilde{\ell}_k \equiv \tilde{\ell}(x_k; r_k)$, $Z_k \equiv Z(x_k)$ and so on to denote the function value at a particular point.

Applying Newton's method on the above nonlinear system, we have the following $(n+m) \times (n+m)$ system

$$\begin{bmatrix} A_k^2 \nabla_x^2 \ell_k + diag(\nabla_x \ell_k) diag(\eta_k) & A_k^2 \nabla H_{e_k} \\ \nabla H_{e_k}^T & 0 \end{bmatrix} \begin{bmatrix} \Delta x_k \\ \Delta \lambda_k \end{bmatrix} = - \begin{bmatrix} A_k^2 \nabla_x \ell_k \\ H_{e_k} \end{bmatrix} \tag{1.15}$$

where $diag(\eta_k)$ is a diagonal matrix whose diagonal elements are defined as follows

$$\eta^{(b)}(x_k) = \begin{cases} 1, & \text{if } \nabla_x \ell(x_k, \lambda_k; r_k)^{(b)} \geq 0 \text{ and } \alpha^{(b)} > -\infty, \\ -1, & \text{if } \nabla_x \ell(x_k, \lambda_k; r_k)^{(b)} < 0 \text{ and } \beta^{(b)} < +\infty \ , \\ 0, & \text{otherwise.} \end{cases} \tag{1.16}$$

For more details about $diag(\eta(x_k))$, see [[14], [15]]. This system is called an extended system.

One of the disadvantages of obtaining Newtons step $(\Delta x_k, \Delta \lambda_k)$ by solving the extended system lies in the fact that the dimension of the extended system is large for large-scale problems. In this paper, we present a method that computes Newtons step $(\Delta x_k, \Delta \lambda_k)$ by solving a smaller dimension linear system.

The paper is arranged as follows. In Section 2, a detailed description of a reduced interior point (RIPT) algorithm to compute Newtons step $(\Delta x, \Delta \lambda)$ by solving a smaller dimension system. Section 3 contains a Matlab implementation of the main algorithm. The main algorithm is applied to three cases study and problem was chosen from the engineering application (Two-Bar Truss), and the results are reported. Finally, Section 4 contains concluding remarks.

In the following section, we offered a detailed description of the main steps to (RIPT) algorithm to solve the system (1.15).

2 A reduced interior-point algorithm

A detailed description of (RIPT) algorithm to compute Newtons step $(\Delta x_k, \Delta\lambda_k)$ by solving a smaller dimension system is introduced in this section.

Let $Z(x_k) \in \Re^{n\times(n-m)}$ be a matrix belongs to the null space of $(A_k^2\nabla H_{e_k})^T$. That is

$$Z_k^T A_k^2 \nabla H_{e_k} = 0. \tag{2.1}$$

Applying QR factorization on $A_k^2\nabla H_{e_k}$, we have

$$A_k^2\nabla H_{e_k} = \begin{bmatrix} Y(x_k) & Z(x_k) \end{bmatrix} \begin{bmatrix} R(x_k) \\ 0 \end{bmatrix}. \tag{2.2}$$

where $R(x_k)$ is an ($m \times m$) upper triangular matrix and $Y(x_k)$ is an $n \times m$ matrix whose orthonormal columns form a basis for the column space of $A^2(x_k)\ \nabla H_e(x_k)$. Notice that, $Y(x_k)^T\, Y(x_k) = I_m$, $Z(x_k)^T\, Z(x_k) = I_{n-m}$, and $Y(x_k)^T\, Y(x_k) + Z(x_k)^T\, Z(x_k) = I_n$. The matrix $R(x_k)$ is nonsingular, if x_k lies in a neighborhood of x_*, and the matrix $A^2(x_k)\ \nabla H_e(x_k)$ is a nonsingular at x_*.

Using a continuous differentiable matrix $Z(x_k)$, the system of equations (1.13)-(1.14) are equivalent to the following system

$$\begin{aligned} Z_k^T A_k^2 \nabla\tilde{\ell}(x_k; r_k) &= 0, \\ H_{e_k} &= 0, \end{aligned}$$

where $\nabla\tilde{\ell}(x_k; r_k) = \nabla F(x_k) + r_k \nabla H_i(x_k) W(x_k) H_i(x_k)$. This system consists of n nonlinear equations. Applying Newton's method on the above system, we have

$$\begin{bmatrix} [Z_k^T[A_k^2\nabla\tilde{\ell}_k]' + [Z_k']^T A_k^2 \nabla\tilde{\ell}_k] \\ \nabla H_{e_k}^T \end{bmatrix} \Delta x = - \begin{bmatrix} Z_k^T A_k^2 \nabla\tilde{\ell}_k \\ H_{e_k} \end{bmatrix}. \tag{2.3}$$

To compute the Jacobian of the above system, we have

$$Z_k^T[A_k^2\nabla\tilde{\ell}_k]' = Z_k^T[A_k^2\nabla^2\tilde{\ell}_k + diag(\nabla\tilde{\ell}_k) diag(\eta_k)], \tag{2.4}$$

where $\nabla^2\tilde{\ell}_k = \nabla^2 F(x_k) + r\nabla H_i(x_k) W(x_k) \nabla H_i(x_k)^T$ and we need to compute $[Z(x_k)']^T$.

To compute $Z(x_k)'$, we differentiate the equation (2.1) to obtain

$$Z_k^T[A_k^2\nabla H_{e_k}]' + [Z_k']^T A_k^2 \nabla H_{e_k} = 0. \tag{2.5}$$

From (2.2) and (2.5), we have

$$[Z_k']^T Y_k = -Z_k^T[A_k^2\nabla H_{e_k}]' R(x_k)^{-1}.$$

Since $[Z_k']^T = [Z_k']^T(Y_k Y_k^T + Z_k Z_k^T)$ and $[Z_k']^T Z_k = 0$, then

$$\begin{aligned} [Z_k']^T &= [Z_k']^T(Y_k Y_k^T + Z_k Z_k^T) \\ &= [Z_k']^T(Y_k Y_k^T) \\ &= -Z_k^T[A_k^2\nabla H_{e_k}]' R(x_k)^{-1} Y_k^T. \end{aligned}$$

Hence

$$[Z_k']^T A_k^2 \nabla \tilde{\ell}_k = -Z_k^T [A_k^2 \nabla H_{e_k}]' R(x_k)^{-1} Y_k^T A_k^2 \nabla \tilde{\ell}_k.$$

If we take

$$\lambda_k = -R(x_k)^{-1} Y_k^T A_k^2 \nabla \tilde{\ell}_k,$$

then

$$[Z_k']^T A_k^2 \nabla \tilde{\ell}_k \quad = \quad Z_k^T [A_k^2 \nabla H_{e_k}]' \lambda. \tag{2.6}$$

From (2.3), (2.4), and (2.6), Newton's step is computed by solving the following system

$$\begin{bmatrix} Z_k^T [A_k^2 \nabla_x^2 \tilde{\ell}_k + diag(\nabla_x \tilde{\ell}_k) diag(\eta_k)] \\ \nabla H_{e_k}^T \end{bmatrix} \Delta x = - \begin{bmatrix} Z_k^T A_k^2 \nabla_x \tilde{\ell}_k \\ H_{e_k} \end{bmatrix}. \tag{2.7}$$

But we need to estimate the positive parameter r_{k+1}. To estimate it, the following subproblem is used

$$minimize \;\; \|\nabla F_{k+1} + r_k \nabla H_{i_{k+1}} W_{k+1} H_{i_{k+1}}\|^2, \tag{2.8}$$

see **??**. The Lagrange multiplier λ should be given by

$$R(x_k)\lambda = -Y(x_k)^T A_k \nabla \tilde{\ell}_k. \tag{2.9}$$

Once the step Δx is evaluated, the damping parameter τ_k is computed to ensure that $x_{k+1} \in int(\tilde{\boldsymbol{F}})$. To ensure that $x_{k+1} \in int(\boldsymbol{F})$, we need to compute a damping parameters τ_k and θ_k. The steps for computing τ_k and θ_k are presented in the following algorithm

Algorithm 2.1 *: (Computing the damping parameters τ_k and θ_k)*
Compute the damping parameter τ_k and θ_k as follows
For $i = 1 : n$, do

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

and

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

Set $\varphi(i) \;=\; \min_i\{u_k^{(i)}, v_k^{(i)}\}\}$.
End.
Set $\tau_k \;=\; \min\{1, \min \varphi\}$.
Set $x_{k+1} = x_k + \tau_k \Delta x_k$.
For $i \;=\; 1 : n$, *do*
If $x_{k+1}(i) \;\in\; int(\boldsymbol{F}(i))$
Set $x_{k+1}(i) \;=\; x_k(i) + \tau_k \Delta x_k(i)$.
Else, compute $\theta_k \in [1 - \sigma\|\Delta x_k\|, 1]$ *and* $\sigma > 0$ *is a pre-specified fixed constant.*
Set $x_{k+1}(i) \;=\; x_k(i) + \theta_k \tau_k \Delta x_k(i)$.
End if.
End for.

Master steps of the reduced interior point (RIPT) algorithm to solve (SOCO) problem are offered in the following algorithm.

Algorithm 2.2 *: (* ***RIPT algorithm*** *)*
A formal description of RIPT algorithm is offered in the following algorithm
Step 0. *Given* $x_0 \in (\tilde{\boldsymbol{F}})$. *Evaluate* r_0, λ_0, W_0, A_0, *and* η_0. *Choose* $\epsilon = 10^{(-10)}$. *Set* $k = 0$.
Step1. *If* $\|Z_k^T A_k^2 \nabla \tilde{\ell}_k\| + \|H_{e_k}\| \leq \epsilon$, *then stopping the algorithm.*
Step2. *Compute Newton's step* Δx_k *by solving the system(2.7).*
Step3. *Using Algorithm (2.1) to compute the damping parameters* τ_k *and* θ_k.
Step 4. *Compute* r_{k+1} *by solving (2.8) and* λ_{k+1} *by solving (2.9).*
Step5. *Compute* W_{k+1} *by using (1.4).*
Step6. *Compute* A_{k+1} *and* η_{k+1} *by using (1.12) and (1.16) respectively.*
Step7. *Set* $k = k + 1$ *and return to Step 1.*

The major steps of our strategy to generate a Pareto optimal front points for (MOCP) problem are expounded in detail as follows:

Algorithm 2.3 ***(The major algorithm)***
Step1. *: Ask the Decisin maker to specify the reference point* $\tilde{y}_j)$ *for all* $j = 1, ..., q$.
Step2. *: Using the achievement secularizing function approach to convert Problem (1.1) to (SOCO) Problem (1.3).*
Step3. *: Algorithm (2.2) is used to solve Problem (1.2) for a certain values of* ω_j *where* $\omega_j \geq 0$, $\sum_{j=1}^{q} \omega_j = 1$.
Step4. *: Formulate Problem (1.3) where* $F(x_k) = maximize_{j=1,...,q}\{\omega_(f_j(x_k) - \tilde{y}_j)\}$.
Step5. *: Compute the PO solutions for the objective functions by using Algorithm (2.2) to solve Problem (1.2).*
Step6. *: Repeat again Steps* $4-5$ *for different values of* ω_j *where* $\omega_j \geq 0$, $\sum_{j=1}^{q} \omega_j = 1$ *to generate the* Pareto-optimal front *for the objective functions.*

3 Implementations

To clarify the effectiveness of Algorithm (2.2), we introduce an extensive diversity of possible numerical constrained test functions. Based upon a general principle of test function selection indicated in [2], various constrained MOPs and distinct *Pareto optimal front* are suggested. Next is Osyczka's second MOP [21], which is a heavily constrained, six decision variable problems. Tanaka suggested by Tanaka [24] is third MOP is also selected such that its constraints makes the Pareto-optimal set discontinuous. We, solve these three problems using our proposed approach and illustrate that results obtained using our approach performed equally well as the that introduced by different authors using well-known algorithms [20]. The numerical results of Algorithm (2.2) have been performed on *a laptop with Intel Core*$(T\ M)i\ 7 - 2670\ QM$ *CPU 2.2 GHz and 8 GB RAM.* Algorithm (2.2) was implemented as a MATLAB code and run under MATLAB version 7.10.0.499 (R 2013 a).

3.1 Test problems and application

Test problem 1:(MOP-C1 Binh2)

$$\begin{array}{ll} \textit{minimize} & f_1(x,y) \;=\; 4\,x^2 \;+\; 4\,y^2 \\ \textit{minimize} & f_2(x,y) \;=\; (5-x)^2 \;+\; (5-y)^2, \\ \textit{subject to} & (5-x)^2 \;+\; y^2 - 25 \;=\; 0, \\ & 7.7 - (8-x)^2 \;-\; (3+y)^2 \;\leq\; 0, \\ & 0 \;\leq\; x \;\leq\; 5, \\ & 0 \;\leq\; y \;\leq\; 3. \end{array}$$

Test problem 2:(MOP-C2 Ocyczka 2)

$$\begin{array}{ll} \textit{minimize} & f_1 \;=\; -\,[25\;(2-x_1)^2 + (2-x_2)^2 + (1-x_3)^2 + (4-x_4)^2 + (1-x_5)^2] \\ \textit{minimize} & f_2 \;=\; x_1^2 \;+\; x_2^2 \;+\; x_3^2 \;+\; x_4^2 \;+\; x_5^2 \;+\; x_6^2, \\ \textit{subject to} & x_1 \;+\; x_2 \;-2 \;=\; 0, \\ & x_1 \;+\; x_2 \;-6 \;\leq\; 0, \\ & -\,x_1 \;+\; x_2 \;-2 \;\leq\; 0, \\ & x_1 \;-\; 3\,x_2 \;-2 \;\leq\; 0, \\ & (3-x_3)^2 \;+\; x_4 \;-4 \;\leq\; 0, \\ & -\,((3-x_5)^2 + x_6) + \;4 \;\leq\; 0, \\ & 0 \;\leq\; x_1,\;\; x_2,\;\; x_6 \;\leq\; 10, \\ & 1 \;\leq\; x_3,\; x_5 \;\leq\; 5, \quad 0 \;\leq\; x_4 \;\leq\; 6. \end{array}$$

Test problem 3:(Tanaka)

$$\begin{array}{ll} \textit{minimize} & f_1(x,y) \;=\; x \\ \textit{minimize} & f_2(x,y) \;=\; y, \\ \textit{subject to} & -\,(x^2+y^2) + \;1 \;+\; 0.1\cos(16\tan^{-1}(\frac{x}{y})) \;\leq\; 0, \\ & (0.5-x)^2 \;+\; (0.5-y)^2 \;-0.5 \leq\; 0, \\ & 0 \;\leq\; x \;\leq\; \pi, \\ & 0 \;\leq\; y \;\leq\; \pi. \end{array}$$

The **Binh2** problem is genuinely basic in that way, the constraints may not present extra trouble in finding the PO solutions as appeared in **Figure (4.2)**. It was watched that all multi-objective optimization algorithms performed similarly well, and gave a dense sampling of solutions along the true Pareto-ideal bend. The Osyczka **Figure (4.4)** and the Tanaka **Figure (4.3)** are generally troublesome. The PO set is discontinuous in the Tnaka problem due to the constraints. The constraints in the TNK problem partition the Pareto-ideal set into five. We compare our strategy and a dependable and efficient multi-objective algorithm. The outcomes demonstrate that our strategy can be utilized productively for constrained MOP problem. For the Osyczka problem, it can be seen that our strategy gave a good sampling of points at the midsection of the curve and a good sampling of points at the extreme ends of Pareto curve. Also, our strategy is applicable to the problem was looked over the engineering application (Two-Bar Truss) [17] and [22].

Application:(*Two-Bar Truss*)

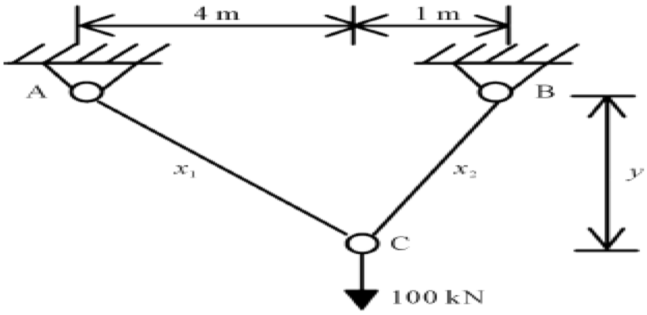

Figure 3.1: Two bar truss

Figure(3.1) illustrates the Two-Bar Truss that will be optimized [22] and was adjusted from [17].

The mathematical formulation of the Two-Bar Truss application is shown below.

$$
\begin{array}{ll}
minimize & f_{volume} = x_1 \sqrt{16 + y^2} + x_2 \sqrt{1 + y^2} \\
minimize & f_{stressAC} = \frac{20\sqrt{16 + y^2}}{x_1 y}, \\
subject\ to & f_{volume} \leq 0.1, \\
 & f_{stressAC} \leq 100000, \\
 & f_{stressBC} = \frac{80\sqrt{1 + y^2}}{x_2 y} \leq 100000, \\
 & 0 \leq x_1, \ \ x_2 \\
 & 1 \leq y \leq 3.
\end{array}
$$

The PO solution of the Two-Bar Truss is presented in **Figure (4.5)**. From the results, we can say that RIPT algorithm is able to get a uniform set of *non-dominated* solution points along the true PO front. Obviously RIPT algorithm beat the algorithm in [17] and [22].

We compare RIPT algorithm with a reliable and efficient multi-objective genetic algorithm NSGA II [1].

4 Conclusions

On treating the multi-objective programming problems. We hope To find a good apportionment of solutions near the PO front in small computational time. We introduce a reduced interior-point method to solve MOP problem, where the active-set technique and Coleman-Li scaling matrix are used together with Newton's method to solve the MOP problem after transform it to SOCO problem and generate approximate true PO front. Algorithm RIPT retains path of all the feasible solutions found during the optimization. It is clear from the results that RIPT algorithm for different test problems and engineering application, is one of the promising approaches for generating a good apportionment of solutions near the PO front in tiny computational times. Also, our results propose that RIPT algorithm could be applied to solve application problems of the real-world. We hope to apply RIPT algorithm for more complex real world application in the future.

References

[1] M. Al-Thobaiti, A. Kotb, and A. Mousa, Integrating optimization technique coupling an evolutionaryalgorithm and local search scheme, Journal of Global Research in Mathematical Archives 2(1)(2014), 11-21.

[2] T. Binh and U. Korn, A parallel multiobjective evolutionary algorithm, Technical report, Institute for Automation and Communication, Barleben, Germany, (1999).

[3] M. Bazaraa, H. Sherali, and C. Shetty, Nonlinear programming theory and algorithms, John Wiley and Sons, (2006).

[4] T. Coleman and Y. Li, An interior trust region approach for nonlinear minimization subject to bounds. SIAM J. Optimization 6(1996), 418-445.

[5] V. Chankongand Y. Haimes, Multiobjective decision making: theory and methodology, North-Holland, New York, 1983.

[6] J. Cohon, Multiobjective programming and planning, Academic Press, New York, 1978.

[7] I. Das, An interior point algorithm for the general nonlinear programming problem, Technical Report 96-61, Institute for Computer Applications in Science and Engineering, NASA Langley Research Center Hampton, VA, USA,(1996).

[8] 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.

[9] J. Dennis, M. Heinkenschloss, and L. Vicente, Trust-region interior-point SQP algorithms for a class of nonlinear programming problems, SIAM Journal on Control and Optimization, 36(1998), 1750-1794.

[10] 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.

[11] 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.

[12] B. El-Sobky, A multiplier active-set trust-region algorithm for solving constrained optimization problem, Applied Mathematics and Computation, 219(2012), 127-157.

[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] B. El-Sobky, An interior-point penalty active-set trust-region algorithm, Journal of the Egyptian Mathematical Society, 24(2016), 672-680.

[15] B. El-Sobky and Y. Abouel-Naga, Multi-objective optimal load flow problem with interior-point trust-region strategy, Electric Power Systems Research 148(2017) 127 - 135.

[16] J. Goodman, Newtons method for constrained optimization, Math. Programming 33(1985), 162-171.

[17] U. Kirsch, Optimal structural design, McGraw-Hill Co., New York, (1981).

[18] K. Miettinen, Nonlinear multiojective optimization, Kluwer Publishers (2002).

[19] K. Miettinen and Makela, M. Interactive bundle-based method for non-differentiable multi-objective optimization. NIMBUS, Optimization, 34(3) (1995), 231-246.

[20] A. Mousa , M. El-Shorbagy, and W. Abd El-Wahed, Local search based hybrid particle swarm optimization for multiobjective optimization, International Journal of Swarm and Evolutionary Computation, 3(2012),1-14.

[21] A. Osyczka and S. Kundu, A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm, Structural Optimization, 10(1995), 94-99.

[22] R. Sarker, H. Abbass, and S. Karim, An evolutionary algorithm for constrained multiobjective optimization problems, 5th Australasia-Japan Joint Workshop, University of Otago, Dunedin, (2001), 1-10.

[23] R. Steuer, Multiple criteria optimization: Theory, computation and applications, New York ,Viley,(1986).

[24] M. Tanaka, GA- baseddecision support system for multi-criteria optimization, Proceeding of the International Conference on Systems, Man and Cybrnetics, 2(1995), 1556-1561.

[25] S. Wang and X. Zhou, Mulltiobjective Optimization in Spillway Profile, Approximation Optimization and Computation: Theory and Applications, (1990) 309-312.

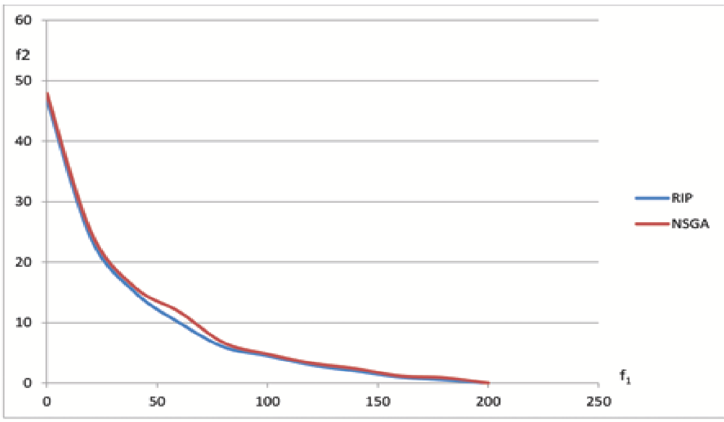

Figure 4.2: Results for Binh problem

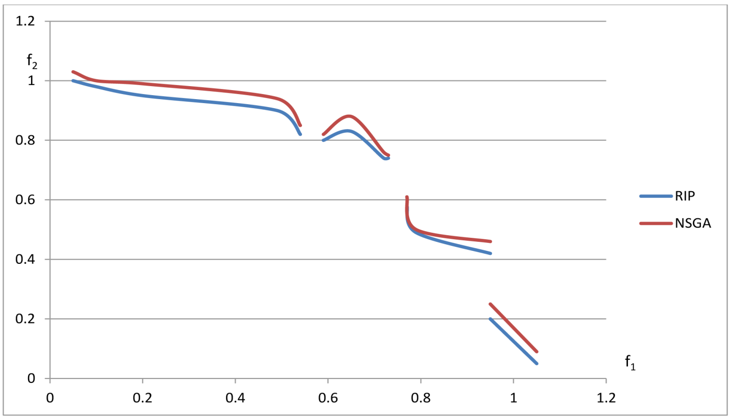

Figure 4.3: Results for Tannaka problem

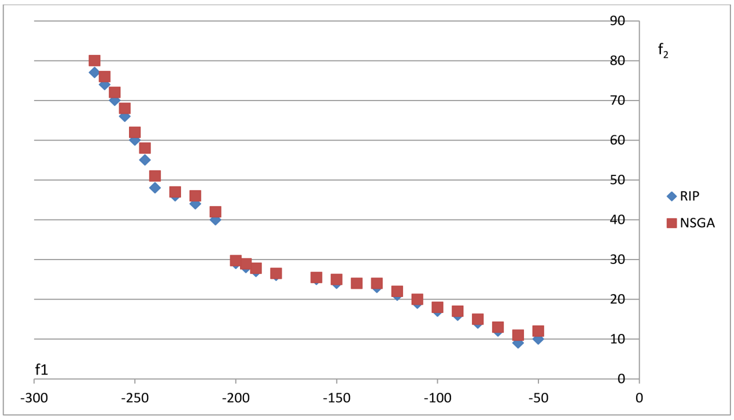

Figure 4.4: Results for OSY problem

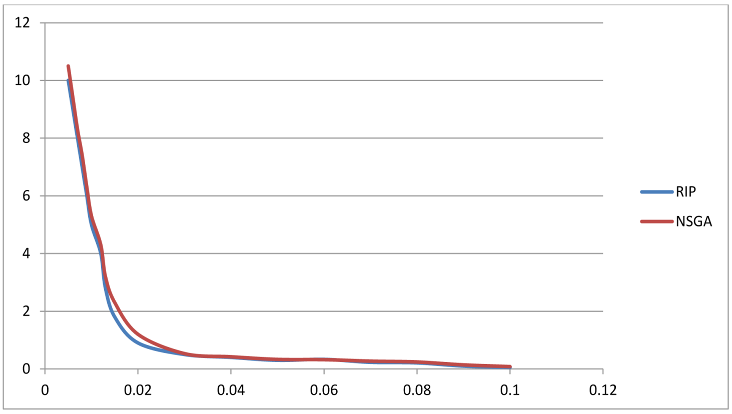

Figure 4.5: The Pareto optimal solution of the Two Bar Truss