Applied Mathematical Sciences, Vol. 6, 2012, no. 54, 2649 - 2667

Utilization of Trust Region Algorithm in Solving Reactive Power Compensation Problem

Bothina El-Sobky [1], Yousria Abo-Elnaga [2] and Hanadi Zahed

Department of Mathematics, Faculty of Science
Taibah University, Al-Madinah Elmnowara, P.O. Box 344
Saudi Arabia

Abstract

In this paper, we introduce a trust region algorithm for solving a reactive power compensation (RPC) problem in a multi-objective context. The trust region algorithm has proven to be a very successful globalization technique for nonlinear programming problems with equality and inequality constraints. The proposed approach is suitable for (RPC) problems where the objective function may be ill- defined, having a nonconvex pareto-optimal front. Also, we identify the weight values which reflect the degree of satisfaction of each objective. The proposed approach is carried out on the standard IEEE 30-bus 6-generator test systems to confirm the effectiveness of the algorithm used to solve the multi-objective RPC problem.

A Matlab implementation of our algorithm was used in solving one case study and the results are reported.

Keywords: Reactive Power Compensation, Multiobjective Optimization, Trust Region, Single Optimization

1 Introduction

In the study of the expansion, planning, and operation of power systems, one of the most important problems is the issue of Reactive Power Compensation (RPC). Identifying the adequate size and the physical distribution of the compensation devices in order to ensure a satisfactory voltage profile while minimizing the cost of compensation, is the main goal of reactive power compensation. In several previous studies of this problem (see [5] and [7]), it has

[1]Permanent position: Department of Mathematics, Faculty of Science, Alexandria University, Egypt

[2]Permanent position: Department of Mathematics, Faculty of Science, Alexandria University, Egypt, e-mail: yousria_naga@yahoo.com

usually been regarded as a single objective optimization problem (SOP), where only one objective is optimized. Multiobjective optimization algorithms are preferable to solve the (RPC) problem over a single objective optimization algorithm because most problems have more than one objective to be optimized and usually the objectives are contradictory. For example, the (RPC) problem requires the optimization of power losses, voltage profile, and investment. Many studies have been undertaken considering this situation, where multiobjective optimization algorithms (MOA) were introduced to simultaneously optimize several independent objectives, (see for example [1], [2], [3], [4], [14], [17], [21], and [23]), demonstrates how traditional multiobjective optimization algorithms usually provide a unique and optimal solution.

On the other hand, multiobjective optimization evolutionary algorithms (MOEA) independently and simultaneously optimize several parameters turning most traditional constraints into new objective functions (see [1], [2], and [3]). This seems more natural for real world problems where choosing a threshold may seem arbitrary (see [17]). As a result, a wide set of optimal solutions (pareto set) may be found. Therefore, an engineer may have a whole set of optimal alternatives before deciding which solution is the best compromise from the different features. In this paper we help the decision maker (DM) in choosing the best comparative solution from all the finite sets of pareto optimal solutions (see [15]). Our work is based on one simple way to combine multiple objective functions into a scalar fitness function which is the weighted sum. Then we do some sort of parametric analysis to name the weight values which reflect the degree of satisfaction of each objective. In this paper, we introduce a new trust region algorithm for solving our problem. In this algorithm, an active set strategy is used together with multiplear method to convert the computation of the trial step to an easy trust region subproblem similar to the one for unconstrained case. The trust-region strategy for solving general nonlinear programming problem with equality and inequality constraints has proven to be very successful both theoretically and practically (see for example [8], [10], [11], and [12]). Finally, the approach has been implemented to select one solution which will satisfy the different objectives to some extent. The standard IEEE 30-bus 6-generator test system has been used to verify the validity of the proposed algorithm. The weighting approach is considered as one of the most useful algorithms in treating multiobjective optimization problems to generate a wide set of optimal solutions (Pareto set) (see [17]).

In this work, the effect of changing the weights on the compensation cost, active power losses and voltage deviation was studied to show the degree of satisfaction of each objective. In each case one weight is changed linearly and the two other weights are generated.

Here, we introduces some notation for subscripted functions denote function values at particular points; for example, $f_k = f(x_k)$, $\nabla_x f_k = \nabla_x f(x_k)$, and so on. The matrix H_k denotes the Hessian of the objective function at the point (x_k) or an approximation to it. Finally, all norms are l_2-norms.

In the following section, we present Reactive Power Compensation (RPC). In section 3, we discuss the multiobjective formulation of the (RPC) problem and in section 4, we present the outline of the trust region algorithm which is used to solve (RPC) problem. We present the implementation of our approach in section 5 and we discuss results and our approach in section 6. Finally, section 7 contains concluding remarks.

2 Reactive Power Compensation (RPC)

In order to identify the adequate size and the physical distribution of the compensation devices for obtaining a satisfactory voltage profile with minimal compensation cost, we consider the following assumptions in the formulation of the problem.

The power system is considered only at peak load. A shunt-capacitor bank cost per MVAr is the same for all busbars of the power system

In this present work, we have identified three objective functions to be minimized and a load flow equations. The three objective functions are f_1, f_2, and f_3, which are related to investment, transmission losses, and quality of service respectively.

The detailed descriptions of these three objective functions is as follows:

First: f_1 is investment in reactive power compensation devices

$$\begin{aligned} \text{minimize} \quad & f_1 = \textstyle\sum_{i=1}^{n} \; B_i \\ \text{subject to} \quad & 0 \leq f_1 \leq f_{1_{max}}, \\ & 0 \leq B_i \leq B_{i_{max}}, \end{aligned} \tag{1}$$

where f_{1max} is the maximum amount available for investment, B_i is the compensation at busbar i measured in MVAr and B_{imax} is the maximum compensation allowed at a particular bus of the system. The number n denoted to the number of buses in the power system. To simplify, we take the price per MVAr as unity.

Second: f_2 is the total active power losses in MW

$$\begin{aligned} \text{minimize} \quad & f_2 = P_g \; - \; P_l \\ \text{subject to} \quad & P_{g_{min}} \leq P_g \leq P_{g_{max}}, \end{aligned} \tag{2}$$

where P_g is the total active power generated, P_l is the total system load, $P_{g_{min}}$ is the minimum active power generated by the generator and $P_{g_{max}}$ is the maximum active power generated by the generator.

Third: f_3 is the average voltage deviation per unit (pu)

$$\begin{aligned} &\text{minimize} \quad f_3 = \tfrac{\|V_i - V_i^*\|_2^2}{n} \\ &\text{subject to} \quad V_{i_{min}} \leq V_i \leq V_{i_{max}}, \qquad\qquad i = 1, 2, ..., n \end{aligned} \tag{3}$$

where V_i and V_i^* are the actual voltage and the desired voltage respectively at busbar i per unity, while $V_{i_{min}}$ and $V_{i_{max}}$ are the minimum and maximum actual voltage respectively at busbar i per unity. For more details about f_1, f_2, and f_3 (see [9] and[16]).

The load flow equations illustrated in [20] are the following equations:

$$\begin{aligned} \Delta P_p &= P_{Gp} - P_{Cp} = \sum_{q=1}^{n} V_p V_q Y_{pq} \cos(\delta_p - \delta_q - \Theta_{pq}) \\ \Delta Q_p &= Q_{Gp} - Q_{Cp} = \sum_{q=1}^{n} V_p V_q Y_{pq} \sin(\delta_p - \delta_q - \Theta_{pq}) \end{aligned}$$

where P_{Gp} and P_{Cp} are the real power generations and demands at bus p respectively, while Q_{Gp} and Q_{Cp} are the reactive power generations and demands at bus p respectivly. V_p and V_q are the voltage magnitude at bus p and q, respectively. Y_{pq} and Θ_{pq} are the admittance magnitude and the admittance angle respectively. δ_p and δ_q are the voltage angle at bus p and q, respectively; where $p = 1, 2, \ldots, n$, $q = 1, 2, \ldots, n$.

The physics of the system as well as the desired voltage set points are reflected throughout the system by the load flow equations. The power flow equations require that the sums of the net injection of real and reactive power at each bus are zero, and thus enforce the physics of the power system.

The size of each reactive bank in the power system is represented by decision vector B, see [6] for example:

$$B = [B_1, B_2, \ldots, B_i, \ldots, B_n], \qquad B_i \in R, \qquad |B_i| \leq B_{imax}$$

in order to represent the amount of reactive compensation to be allocated at each busbar i.

(RPC) can therefore be seen as a complex, comprehensive optimization problem dealing with multiple nonlinear functions with multiple local minima which may be ill-defined and nonlinear constraints which lead to a nonconvex Pareto-optimal front, (see [6] and [17]). The trust region concept could cover these problems to get a finite set of Pareto optimals, where it is difficult for the (DM) to choose the best compromise solution among these Pareto set.

In the following section we rewrite the reactive power compensation problem in mathematical form and using a weighting appoach to transform it to a single objective optimization problem.

3 Multiobjective Formulation of the RPC Problem

The mathematical form of the reactive power compensation problem with n-buses and m-generator is the following multi-objective optimization problem:

$$\begin{aligned}
\text{minimize} \quad & f_1 = \textstyle\sum_{i=1}^{n} \; B_i \\
\text{minimize} \quad & f_2 = P_{g_j} \; - \; P_{l_j} \\
\text{minimize} \quad & f_3 = \tfrac{\|V_i - V_i^*\|_2^2}{n} \\
\text{subject to} \quad & 0 \leq f_1 \leq f_{1_{max}}, \\
& 0 \leq B_i \leq B_{i_{max}}, \\
& V_{i_{min}} \leq V_i \leq V_{i_{max}}, \\
& P_{g_{j_{min}}} \leq P_{g_j} \leq P_{g_{j_{max}}}, \\
& \Delta P_p = P_{Gp} - P_{Cp} = \textstyle\sum_{q=1}^{n} V_p V_q Y_{pq} cos(\delta_p - \delta_q - \Theta_{pq}), \\
& \Delta Q_p = Q_{Gp} - Q_{Cp} = \textstyle\sum_{q=1}^{n} V_p V_q Y_{pq} sin(\delta_p - \delta_q - \Theta_{pq}),
\end{aligned} \tag{1}$$

where $i = 1, 2, ..., n$ and $j = 1, ..., m$.

Using a weighting approach to transform problem (1) to a single-objective optimization problem which has the following form:

$$\begin{aligned}
\text{minimize} \quad & f(x) = w_1 f_1 + w_2 f_2 + w_3 f_3 \\
\text{subject to} \quad & 0 \leq f_1 \leq f_{1_{max}}, \\
& 0 \leq B_i \leq B_{i_{max}}, \\
& V_{i_{min}} \leq V_i \leq V_{i_{max}}, \\
& P_{g_{j_{min}}} \leq P_{g_j} \leq P_{g_{j_{max}}}, \\
& \Delta P_p = P_{Gp} - P_{Cp} = \textstyle\sum_{q=1}^{n} V_p V_q Y_{pq} cos(\delta_p - \delta_q - \Theta_{pq}), \\
& \Delta Q_p = Q_{Gp} - Q_{Cp} = \textstyle\sum_{q=1}^{n} V_p V_q Y_{pq} sin(\delta_p - \delta_q - \Theta_{pq}),
\end{aligned} \tag{2}$$

where $x^T = [B_1, ..., B_n, P_{g_1}, ..., P_{g_m}, V_1, ..., V_n]$, $\sum_{k=1}^{3} w_k = 1$, and $w_k \geq 0$, $k = 1, 2, 3$.

The following section is devoted to presenting the detailed description of our trust-region algorithm for solving problem (2).

4 Turst Region Algorithm Outline

In this section, we consider the following single objective optimization problem

$$\begin{array}{lrcl} \text{minimize} & f(x) & & \\ \text{subject to} & h(x) & = & 0, \\ & g(x) & \leq & 0, \end{array} \tag{1}$$

where the functions $f(x) : \Re^{2n+m} \rightarrow \Re$, $h(x) : \Re^{2n+m} \rightarrow \Re^{2n}$, and $g(x) : \Re^{2n+m} \rightarrow \Re^{4n+2m+2}$ are twice continuously differentiable. The Lagrangian function associated with problem (1) is the function

$$l(x, \lambda, \mu) = f(x) + \lambda^T h(x) + \mu^T g(x), \tag{2}$$

where $\lambda \in \Re^{2n}$ and $\mu \in \Re^{4n+2m+2}$ are the Lagrange multiplier vectors associated with equality and inequality constraints respectivly.

Following [8], we define a 0-1 diagonal indicator matrix $Z(x) \in \Re^{\bar{p} \times \bar{p}}$, whose diagonal entries are

$$z_i(x) = \begin{cases} 1 & \text{if } g_i(x) \geq 0, \\ 0 & \text{if } g_i(x) < 0, \end{cases} \tag{3}$$

where $\bar{p} = 2n + 4m + 2$.

Using the above matrix, we transform problem (1) to the following equality constrained optimization problem

$$\begin{array}{lrcl} \text{minimize} & f(x) & & \\ \text{subject to} & h(x) & = & 0, \\ & Z(x)g(x) & = & 0. \end{array} \tag{4}$$

Using a multiplier method, we transform the equality constrained optimization problem (4) to the following unconstrained optimization problem

$$\begin{array}{ll} \text{minimize} & \Phi(x, \lambda, \mu; \rho; r) = l(x, \lambda, \mu) + \frac{\rho}{2}\|Z(x)g(x)\|_2^2 + \frac{r}{2}\|h(x)\|_2^2, \\ \text{subject to} & x \in \Re^{2n+m}, \end{array} \tag{5}$$

where ρ is the positive parameter and $r > 0$ is a parameter usually called the penalty parameter.

4.1 Computing a Trial Step

We compute the trial step s_k by solving the following trust-region subproblem

$$\begin{array}{ll} \text{minimize} & l_k + \nabla l_k^T s + \frac{1}{2} s^T H_k s + \frac{\rho_k}{2} \| Z_k (g_k + \nabla g_k^T s)\|^2 + \frac{r_k}{2} \|(h_k + \nabla h_k^T s)\|^2 \\ \text{subject to} & \|s\| \leq \delta_k, \end{array} \tag{6}$$

where H_k is the Hessian matrix of the Lagrangian function $l(x_k, \lambda_k, \mu_k)$ or an approximation to it. Since our convergence theory is based on the fraction of Cauchy decrease condition, any method that computes the trial step in such a way that the fractions of the Cauchy decrease can be used. Therefore, a dogleg algorithm can be used to compute the trial step. More details can be found in [8].

4.2 Testing the Step and Updating δ_k

Once the trial step is computed, it needs to be tested to determine whether it will be accepted. To test the step, estimates for the two Lagrange multipliers λ_{k+1} and μ_{k+1} are needed. Our way of evaluating the two Lagrange multipliers λ_{k+1} and μ_{k+1} is presented in Step 5 of Algorithm (4.3) below.

Let λ_{k+1} and μ_{k+1} be estimates of the two Lagrange multiplier vectors. We test whether the point $(x_k + s_k, \lambda_{k+1}, \mu_{k+1})$ will be taken as a next iterate.

The actual reduction in the merit function is defined as

$$\begin{aligned} Ared_k &= l(x_k, \lambda_k, \mu_k) - l(x_{k+1}, \lambda_k, \mu_k) - \Delta\lambda_k^T h_{k+1} - \Delta\mu_k^T g_{k+1} \\ &+ \frac{\rho_k}{2}[g_k^T Z_k g_k - g_{k+1}^T Z_{k+1} g_{k+1}] + \frac{r_k}{2}[\|h_k\|^2 - \|h_{k+1}\|^2], \end{aligned} \tag{7}$$

where $\Delta\lambda_k = (\lambda_{k+1} - \lambda_k)$ and $\Delta\mu_k = (\mu_{k+1} - \mu_k)$.

The predicted reduction in the merit function is defined to be

$$\begin{aligned} Pred_k &= q_k(0) - q_k(s_k) - \Delta\lambda_k^T (h_k + \nabla h_k^T s_k) - \Delta\mu_k^T Z_k g_k \\ &+ \frac{r_k}{2}[\|h_k\|^2 - \|h_k + \nabla h_k^T s_k\|^2], \end{aligned} \tag{8}$$

where

$$q_k(s) = l_k + \nabla_x l_k^T s + \frac{1}{2} s^T H_k s + \frac{\rho_k}{2} \| Z_k (g_k + \nabla g_k^T s)\|^2. \tag{9}$$

After computing a trial step and updating the Lagrange multipliers, the penalty parameter is updated to ensure that $Pred_k \geq 0$. To update r_k, we use a scheme that has the flavor of the scheme proposed by [10]. This scheme is described in Step 6 of Algorithm 4.3 below. After that, the step is tested to know whether it is accepted. This is done by comparing $Pred_k$ against $Ared_k$.

If $\frac{Ared_k}{Pred_k} < \eta_1$ where $\eta_1 \in (0,1)$ is a small fixed constant, then the step is rejected. In this case, the radius of the trust region δ_k is decreased by setting $\delta_k = \alpha_1 \|s_k\|$, where $\alpha_1 \in (0,1)$, and another trial step is computed using the new trust-region radius.

If $\frac{Ared_k}{Pred_k} \geq \eta_1$, then the step is accepted. Our theory requires that at the beginning of the next iteration, δ_{k+1} must be greater than or equal to δ_{min}, where δ_{min} is a positive constant chosen at the beginning of the algorithm. That is, δ_k can be reduced below δ_{min} while finding an acceptable step. But, $\delta_{k+1} \geq \delta_{min}$ is required at the beginning of the next iteration after accepting the step s_k.

Our way of evaluating the trial steps and updating the trust-region radius is presented in Step 7 of Algorithm (4.3) below. After accepting the step, we update the parameter ρ_k and the Hessian matrix H_k. To update ρ_k, we use a scheme suggested by [24]. This scheme is described in Step 8 of Algorithm 4.3 below.

Finally, the algorithm is terminated when either $\|s_k\| \leq \varepsilon_1$ or $\|\nabla_x l_k\| + \|\nabla g_k Z_k g_k\| + \|h_k\| \leq \varepsilon_2$, for some $\varepsilon_1, \varepsilon_2 > 0$,

4.3 Main Algorithm

A formal description of our trust-region algorithm for solving problem (2) is presented in the following algorithm.

Algorithm (The Main Algorithm)

Step 0. (Initialization)

Given $x_1 \in \Re^{2n+m}$. Compute Z_1. Evaluate μ_1 and λ_1 (see Step 5 with $k = 0$ and $\lambda_0 = (0, 0, ..., 0)^T$). Set $\rho_1 = 1$, $r_0 = 1$, $\sigma_1 = 1$, and $\beta = 0.1$. Choose ε_1, ε_2, α_1, α_2, η_1, and η_2 such that $\varepsilon_1 > 0$, $\varepsilon_2 > 0$, $0 < \alpha_1 < 1 < \alpha_2$, and $0 < \eta_1 < \eta_2 < 1$. Choose δ_{min}, δ_{max}, and δ_1 such that $\delta_{min} \leq \delta_1 \leq \delta_{max}$. Set $k = 1$.

Step 1. (Test for convergence)

If $\|\nabla_x l_k\| + \|\nabla g_k Z_k g_k\| + \|h_k\| \leq \varepsilon_2$, then terminate the algorithm.

Step 2. (Compute a trial step)

a)Compute the step s_k by solving (6)
b) Set $x_{k+1} = x_k + s_k$.

Step 3. (Test for termination)

If $\|s_k\| \leq \varepsilon_1$, then terminate the algorithm.

Step 4. (Update the active set)

Compute Z_{k+1}.

Step 5. (Compute the Lagrange multipliers μ_{k+1} and λ_{k+1})

a) Compute μ_{k+1} by solving

$$\begin{array}{ll} \textit{minimize} & \|(\nabla f_{k+1} + \nabla h_{k+1}\lambda_k + \nabla g_{k+1} Z_{k+1}\mu)\|^2 \\ \textit{subject to} & Z_{k+1}\mu \geq 0, \end{array} \tag{10}$$

and set the rest of the components of μ_{k+1} to zero.
b) If $\|\nabla f_{k+1} + \nabla h_{k+1}\lambda_k + \nabla g_{k+1} Z_{k+1}\mu_{k+1}\| \leq \varepsilon$,
then set $\lambda_{k+1} = \lambda_k$.
Else, compute λ_{k+1} by solving

$$\textit{minimize} \;\; \|\nabla f_{k+1} + \nabla g_{k+1}\mu_{k+1} + \nabla h_{k+1}\lambda\|^2. \tag{11}$$

End if.

Step 6. (Update the penalty parameter r_k)

a) Set $r_k = \max(r_{k-1}, \rho_k^2)$.
b) If $Pred_k \leq \frac{r_k}{4}[\|h_k\|^2 - \|h_k + \nabla h_k^T s_k\|^2]$,then set

$$r_k = \frac{4[q_k(s_k) - q_k(0) + \Delta\lambda_k^T(h_k + \nabla h_k^T s_k) + \Delta\mu_k^T Z_k g_k]}{\|h_k\|^2 - \|h_k + \nabla h_k^T s_k\|^2} + \beta.$$

End if.

Step 7. (Test the step and update the trust-region radius)

If $\frac{Ared_k}{Pred_k} < \eta_1$.

Reduce the trust-region radius by setting $\delta_k = \alpha_1\|s_k\|$ and go to step 2.

Else if $\eta_1 \leq \frac{Ared_k}{Pred_k} < \eta_2$, then

Accept the step: $x_{k+1} = x_k + s_k$.
Set the trust-region radius: $\delta_{k+1} = \max(\delta_k, \delta_{min})$.

Else

Accept the step: $x_{k+1} = x_k + s_k$.
Set the trust-region radius: $\delta_{k+1} = \min\{\delta_{max}, \max\{\delta_{min}, \alpha_2\delta_k\}\}$.

End if.

Step 8. (Update the parameters ρ_k and σ_k)

a) Set $\rho_{k+1} = \rho_k$ and $\sigma_{k+1} = \sigma_k$.
b) If

$$Tpred_k - \Delta\lambda_k^T(h_k + \nabla h_k^T s_k) - \Delta\mu_k^T Z_k g_k < \sigma_k \|\nabla g_k Z_k g_k\| \min\{\|\nabla g_k Z_k g_k\|, \delta_k\},$$

then set $\rho_{k+1} = 2\rho_k$ and $\sigma_{k+1} = \frac{1}{2}\sigma_k$.
End if.

Step 9. Set $k = k + 1$ and go to Step 1.

5 Implementation of the Proposed Approach

To investigate the effectiveness of the proposed approach the described methodology is applied to the standard IEEE 30-bus 6-generator test system, see figure (1). In [25] we can see a detailed presentation of the data for this system. In this work, our program was written in MATLAB and run under MATLAB Version 7 with machine epsilon about 10^{-16}.

Given a starting point x_1, we chose the initial trust-region radius to be $\delta_1 = max(\|s_1^{cp}\|, \delta_{min})$, where δ_{min} was taken to be $\delta_{min} = 10^{-3}$. We chose the maximum trust-region radius to be $\delta_{max} = 10^5\delta_1$.

The values of the constants that are needed in Step 0 of Algorithm (4.3) were set to be $\eta_1 = 10^{-4}$, $\eta_2 = 0.5$, $\alpha_1 = 0.5$, $\alpha_2 = 2$, $\varepsilon_1 = 10^{-8}$, $\varepsilon_2 = 10^{-8}$, and $\beta = 0.1$. For computing the two components of the trial steps, we used the dogleg algorithm. Successful termination with respect to our trust-region algorithm means that the termination condition of the algorithm is met with $\varepsilon_2 = 10^{-8}$. On the other hand, unsuccessful termination means that the number of iterations is greater than 300, the number of function evaluations is greater than 500, or the length of the trial step is less than $\varepsilon_1 = 10^{-8}$.

6 Results and Discussions

Here we discuss the effects of changing weights on active power losses, the compensation cost, as well as the average voltage deviation. As one weight is changed linearly in each case, the other two weights are generated randomly thus that: $w_1 + w_2 + w_3 = 1$. To obtain the best compromise for the operating point we observed the corresponding values of the weights for the values of $f_1(.)$, $f_2(.)$ and $f_3(.)$.

The three tables (1), (2) and (3) show the values of the weights for the three cases which were studied, where in each case, one weight is changed

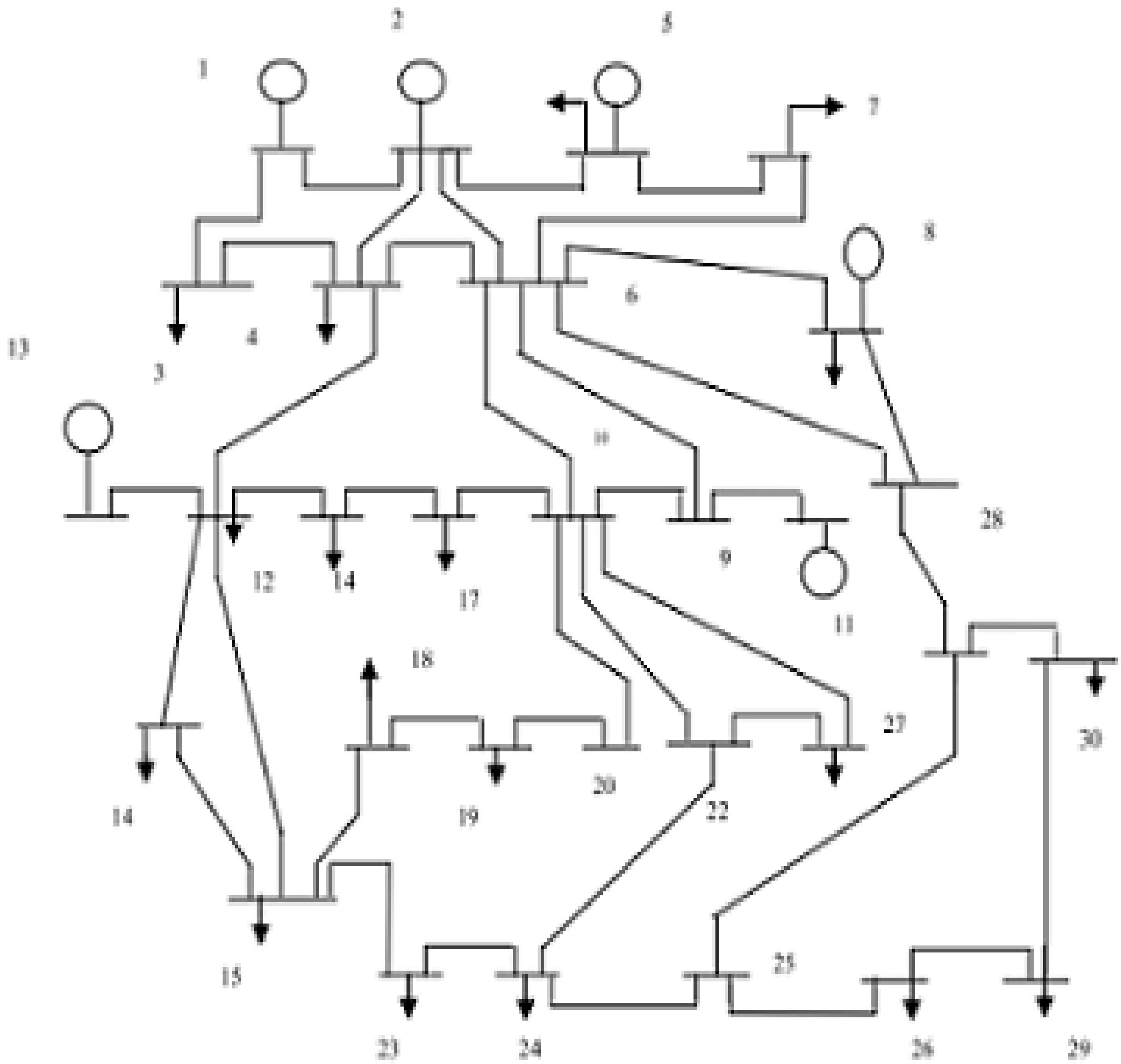


Figure 1: IEEE 30-bus Test System.

linearly taking six values, while the other two weights are obtained using the relation $w_1+w_2+w_3=1$. Figures (2), (3) and (4) show the objective functions obtained from the six solutions corresponding to the six weights compared to the weights for the three cases. Thus, we observe that:

1- We obtained the highest cost at the lowest w_1, while the highest value of w_1 gave us the lowest cost.

2- The change of weight in w_2 and w_3 has less effect on active power losses and average voltage division, respectively, than the change of the weight w_1 which has a stronger effect on compensation cost.

3- In the light of these results we recommended to choose w_1 around 0.6 because the change of the cost corresponding to values of w_1 higher than 0.6 is not significant.

7 Conclusions

In this paper we use a trust region method to solve the reactive power compensation problem formulated as multiobjective optimization problem with competing amounts of reactive compensation devices, active power losses and voltage deviation. We have arrived at the conclusion that this approach is a new technique for the numerical, parametric study of RPCs, when the problem is complex and has many real world applications. We have solved the RPC, by applying the proposed technique while considering three objectives simultaneously. Non-convex multiobjective optimization problems have been efficiently solved using the proposed approach. Our approach is interactive because it allows the decision maker to specify the relative weights of criterion importance which show the degree to which the objectives have been satisfied.

8 Acknowledgements

First and foremost, we give thanks to God for the grace and mercy he has shown us. We depend on Him and we could not have accomplished this work without Him.

Words cannot express our deep gratitude to The Deanship of Scientific Research, Taibah University, KSA for supporting this project by grant No. 1432/79.

Run	w_1	w_2	w_3
1	0	0.5721	0.4279
2	0.2000	0.6705	0.1295
3	0.4000	0.2118	0.3882
4	0.6000	0.2636	0.1364
5	0.8000	0.0759	0.1241
6	1.0000	0	0

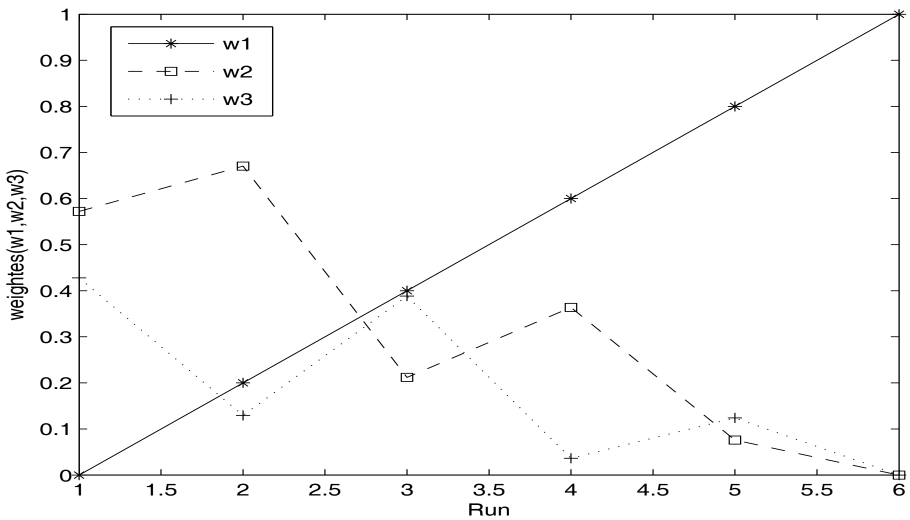
Table 1: Different weights (w_1 is changed linearly).

Run	w_1	w_2	w_3
1	0.6028	0	0.3972
2	0.5676	0.2	0.2324
3	0.4573	0.4	0.1427
4	0.2218	0.6	0.1782
5	0.1468	0.8	0.0532
6	0	1.0000	0

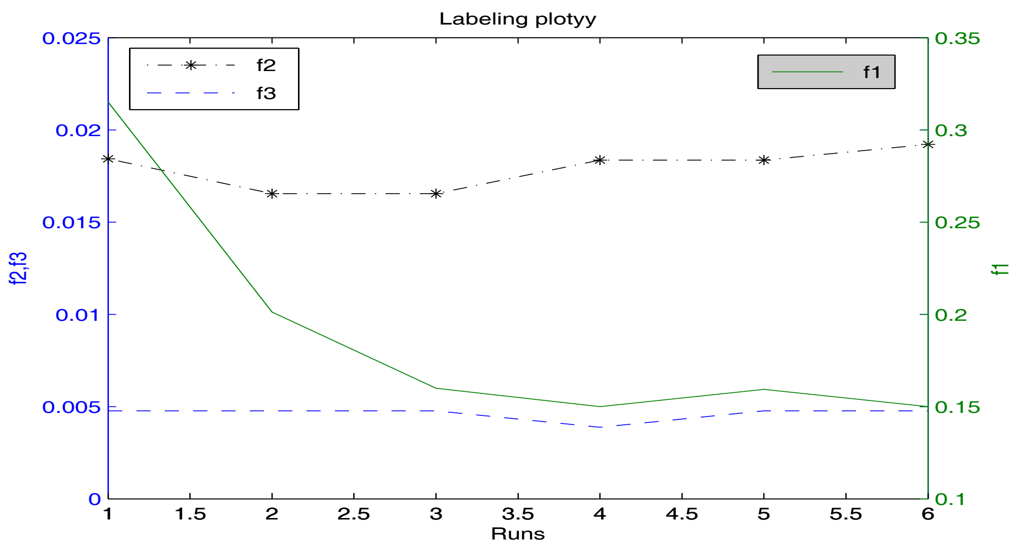
Table 2: Different weights (w_2 is changed linearly).

Run	w_1	w_2	w_3
1	0.7477	0.2523	0
2	0.2994	0.5006	0.2
3	0.1576	0.4424	0.4
4	0.1354	0.2646	0.6
5	0.1976	0.0024	0.8
6	0	0	1

Table 3: Different weights (w_3 is changed linearly).

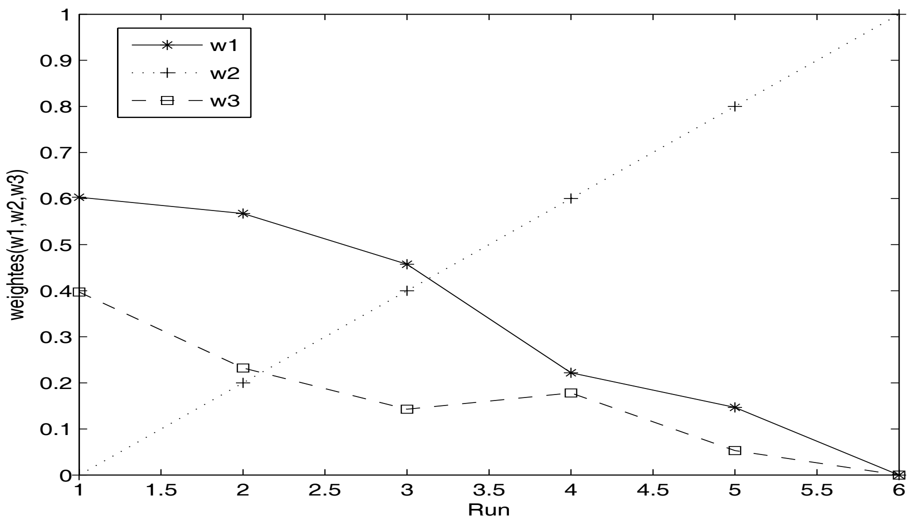


(a) Different weights (W_1 is changed linearly)

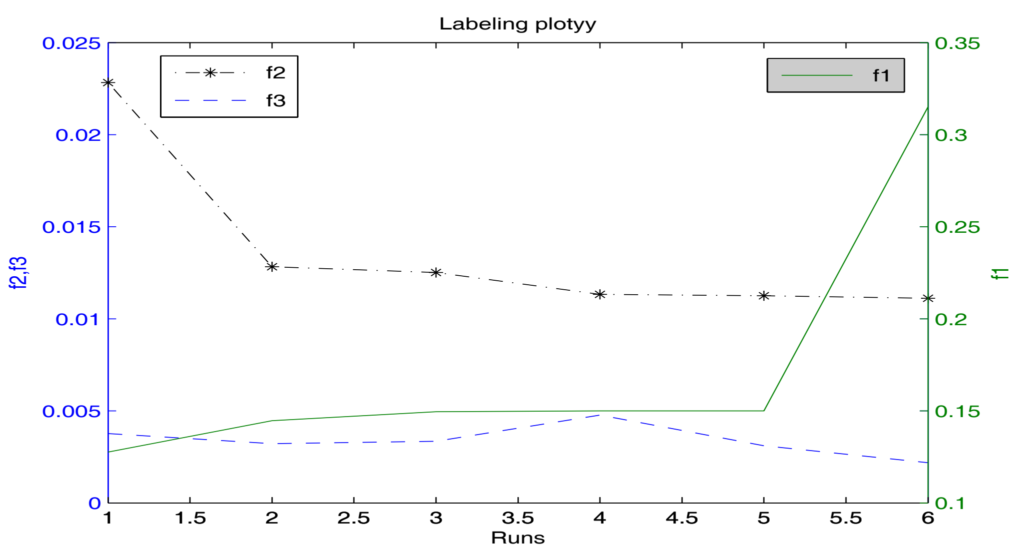


(b) Plot showing (f_1, f_2 and f_3)

Figure 2: Plot showing best compromise solution for different weights in 6 runs of table 1.

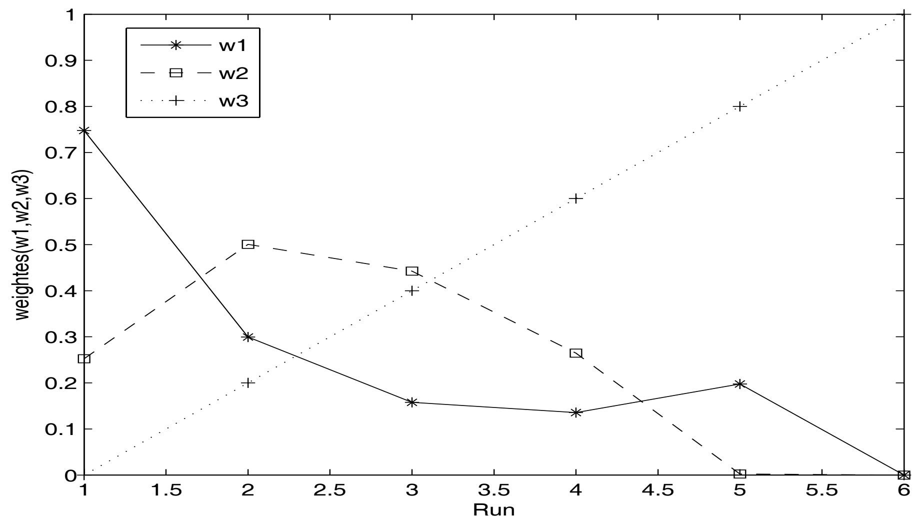


(a) Different weights (W_2 is changed linearly)

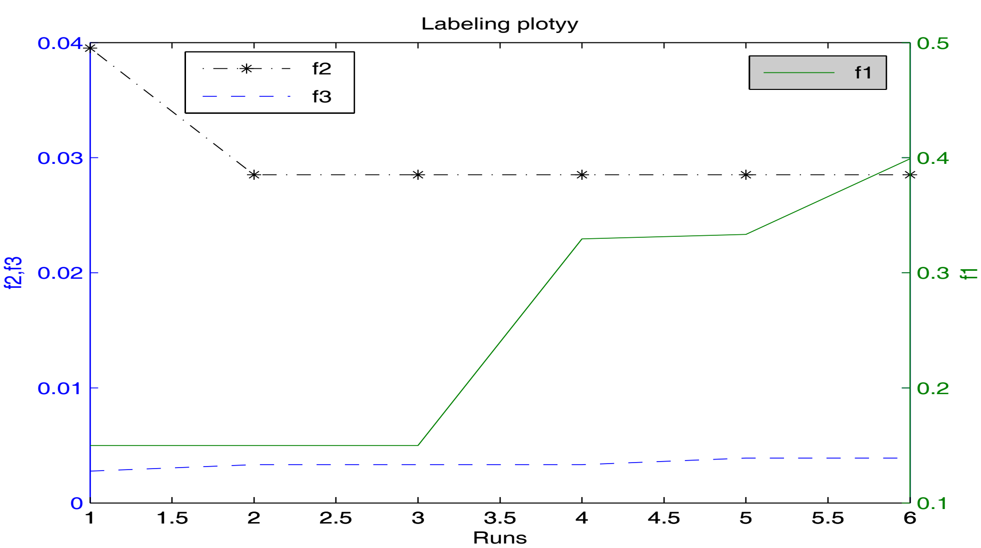


(b) Plot showing (f_1, f_2 and f_3)

Figure 3: Plot showing best compromise solution for different weights in 6 runs of table 2.



(a) Different weights (W_3 is changed linearly)



(b) Plot showing (f_1, f_2 and f_3)

Figure 4: Plot showing best compromise solution for different weights in 6 runs of table 3.

References

[1] M. A. Abido. Multiobjective evolutionary algorithms for electric power dispatch problem. IEEE Trans. On Evolutionary Computation, Vol. 10, No.3, (2006).

[2] B. Baran, J. Vallejos and R. Ramos. Multi-objective reactive power compensation with voltage security. Proc. IEEE Transmission and Distribution Conference and Exposition, Latin America, Sao Paulo, Brazil, (2004).

[3] B. Baran, J. Vallejos, R. Ramos and U. Fernandez. Reactive power compensation using multi-objective evolutionary algorithm. Proc. IEEE Porto Power Tech 2001, Portugal, (2001).

[4] B. Baran, J. Vallejos, R. Ramos and U. Fernandez. Multi-objective reactive power compensation. Proc. IEEE Transmission and Distribution Conference and Exposition, Atlanta, USA, (2001).

[5] J. Carlisle, A. El-Keib, D. Boyd, and K. Nolan. A review of capacitor placement techniques on distribution feeders. in Proc. IEEE 29 Southeastern Symposium on system Theory (SSST'97), (1997).

[6] K. Deb. Multi-objective optimization using evolutionary algorithms. Wiley, NY, USA, (2001).

[7] M. Delfanti, G. Granelli, P. Marannino, and M. Montagna. Optimal capacitor placement using deterministic and genetic algorithms. IEEE Trans. Power Systems, Vol.15, No.3, pp. 1041-1046,(2000).

[8] J. Dennis, M. El-Alem, and K. Williamson. A trust-region approach to nonlinear systems of equalities and inequalities. SIAM J Optimization,9:291-315, (1999).

[9] H. Dommel, and W. Tinney. Optimal Power Flow Solutions. IEEE Trans. Power Apparatus and Systems, vol. PAS-87, No.10, pp.1866-1876, (1968).

[10] M. El-Alem. A global convergence theory for a class of trust-region algorithms for constrained optimization. PhD thesis, Department of Mathematical Sciences, Rice University, Houston, Texas, (1988).

[11] B. El-Sobky. Arobust trust-region algorithm for general nonlinear constrained optimization problems. PhD thesis, Department of Mathematics, Alexandria University, Alexandria, Egypt, (1998).

[12] 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 No.1:127-157,(2003).

[13] R. Fletcher. Practical methods of optimization. John Wiley and Sons,Chichester, (1987),2nd ed.

[14] L. Furon, J. D. Pilgrim, C. Dabeedin, A. Chebbo, and R. K. Aggarwal. Genetic algorithms for optimal reactivepower compensation on national grid system. IEEE Trans. Power Systems, Vol. 20, No.1, pp. 493-500, (2005).

[15] Y. Haimesy. Multiple-Criteria Decisionmaking Aretrospective analysis, IEEE Transactions on systems, Man and Gyberrnetics Vol. 15, No.3, pp. 313-315, (1985).

[16] P. Kundur. Power System Stability and Control. New York: Mc Graw-Hill, NY, USA, (1993).

[17] K. Miettinen. Non-linear multiobjective optimization. Dordrecht: Kluwer Academic Publisher, (2002).

[18] K. Miu, H. Chiang and G. Darling. Capacitor placement, replacement and control in large scale distribution systems by A-based two-stage algorithms. IEEE Trans. Power Systems, Vol. 12, pp. 1160-1166, (1997).

[19] J. Moré and D. Sorensen. Computing a trust-region step. Siam.J on scientific and statistical computing, 4:553-572,(1983).

[20] M. S. Osman, M. A. Abo-Sinna and A. A. Mousa. A solution to the optimal power flow using genetic algorithms. Appl. Math. Comput, Vol. 155, pp. 391-405, (2004).

[21] D. F. Pires, C. H. Antunes and A. G. Martins. A tabu search multiobjective approach to capacitor allocation in radial distribution systems. MIC' 2001, Porto, Portugal, pp. 169-174, (2001).

[22] D. Van Veldhuizen. Multi-objective evolutionary algorithms classifications, analysis and new innovations. PhD. Dissertation, Faculty of Graduate School of Engineering, Air Force Institute of Technology, (1997)

[23] B. Venkatesh, G. Sadasivam and Khan Abdullah. A new optimal reactive power scheduling method for loss minimization and voltage stability margin maximization using successive multi-objective fuzzy LP technique. IEEE Trans. Power Systems, Vol. 15, (2000).

[24] Y. Yuan. On the convergence of a new trust region algorithm. Numer. Math. 70:515-539, (1995).

[25] R. Zimmerman, D. Gan, MATPOWER: A Matlab power system simulation package. Available: http://www.pserc.cornell.edu/ matpower/.

Received: December, 2011