Multi-objective economic emission load dispatch problem with trust-region strategy

Bothina El-sobkya,b,∗, Yousria Abo-elnagab

a Department of Mathematics, Faculty of Science, Alexandria University, Alexandria, Egypt
b Department of basic science, Higher Technological Institute, Tenth of Ramadan City, Egypt

A R T I C L E   I N F O

Article history:
Received 21 February 2013
Received in revised form 27 October 2013
Accepted 3 November 2013

Keywords:
Economic emission load dispatch
Multi-objective optimization
Trust region
Weighting approach

A B S T R A C T

In this paper, we present a trust region algorithm for solving multi-objective economic emission load dispatch problem (EELD). The trust region algorithm has proven to be a very successful globalization technique for solving a single objective constrained optimization problems. The proposed approach is suitable for multi-objective problem (EELD) such that its objective functions may be ill-defined or having a non convex 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 problem (EELD). Our results with the proposed approach have been compared to those reported in the literature. The comparison demonstrates the superiority of the proposed approach and confirm its potential to solve the multi-objective problem (EELD).

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

Published by Elsevier B.V.

1. Introduction

The multi-objective problem (EELD) is of great interest to many researchers and several local methods have been proposed to solve it. By local method we mean that the method is designed to converge to optimal solution from closest starting point whether it is local or global one. For a local method, there is no guarantee that it converges if it starts from remote.

The purpose of multi-objective problem (EELD) is to figure out the optimal amount of the generated power of the fossil-based generating units in the system by minimizing the fuel cost and emission level simultaneously, subject to various equality and inequality constraints including the security measures of the power transmission/distribution. Various optimization techniques have been proposed by many researchers to deal with this multi-objective nonlinear programming problem with varying degree of success. In [1,2] the problem has been reduced to a single objective problem by treating the emission as a constraint with a permissible limit. This formulation, however, has severe difficulty in getting the trade-off relation between cost and emission.

Goal programming method was also proposed for the multi-objective problem (EELD) (see [3]). In this method a target or a goal to be achieved for each objective is assigned and the objective function will then try to minimize the distance from the targets to the objectives. Although the method is computationally efficient, it will yield an inferior solution rather than a non-inferior one if the goal point is chosen in the feasible domain.

Heuristic algorithms such as genetic algorithms have been recently proposed for solving multi-objective problem (EELD) (see for example [4–6]). The results reported were promising and encouraging for further research. Moreover the studies on heuristic algorithms over the past few years, have shown that these methods can be efficiently used to eliminate most of difficulties of classical methods (see for example [7–11]). Further more, these methods cannot be used to find pareto-optimal solutions in problems having a non convex pareto-optimal front or ill defined problems.

In this paper, we will use a trust-region globalization strategy to solve the multi-objective problem (EELD). Globalizing strategy means modifying the local method in such a way that it is guaranteed to converge at all even if the starting point is far away from the solution. This approach is applied to solve multi-objective problem (EELD) with no limitation to the number of objective functions and is efficient for solving ill-defined systems and non-convex multi-objective optimization problems.

In this work, we convert the multi-objective problem (EELD) to a single-objective constrained optimization problem by using a weighting approach. The weighting approach is considered as one of the most useful algorithms in treating multi-objective...
optimization problems to generate a wide set of optimal solutions (pareto set) [12]. Also, the effect of changing the weights on cost and emission were studied to show the degree of satisfaction of each objective function.

In this paper, an active set strategy is used together with a multiplier method to convert the single-objective constrained optimization problem to unconstrained optimization problem. The trust-region strategy for solving the single-objective constrained optimization problem and unconstrained optimization problem has proved to be very successful, both theoretically and practically (see for example [13–18]). Also the trust-region technique for the multi-objective problem has proved to be very successful (see for example [19–21]).

Here, we introduce some notations for subscripted functions denote function values at particular points; for example, \( f_k = f(x_k) \), \( \nabla f_k = \nabla f(x_k) \), \( f_k = f(x_k) \). 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.

This paper is organized as follows: In Section 2 we introduce in details the description of economic emission load dispatch problem. Section 3 is devoted for the mathematical formulation of multi-objective problem (EELD). In Section 4 we give a detailed discussion of the trust region algorithm problem (EELD). Furthermore, we then discuss in detail the implementation of the proposed approach in Section 5. The results and discussions are presented in Section 6. Finally, the conclusion and future works are given in Section 7.

2. Economic emission load dispatch problem

The economic emission load dispatch involves the simultaneous optimization of fuel cost and emission objectives which are conflicting ones. The deterministic problem is formulated as follows:

2.1. Objective functions

There are two objective functions which are described in details as follows:

(1) Fuel cost objective function. The classical economic dispatch problem of finding the optimal combination of power generation, which minimize the total fuel cost while satisfying the total required demand can be mathematically stated in [22] as follows:

\[
f_1 = \sum_{i=1}^{n} C_i(P_{Gi}) = \sum_{i=1}^{n} (a_i + b_i P_{Gi} + c_i P_{Gi}^2)
\]

(2.1)

where \( f_1 \) is the total fuel cost($/h), \( C_i \) is the fuel cost of generator \( i \), \( P_{Gi} \) is the power generated by generator \( i \), \( n \) is the number of generator, and \( a_i, b_i, c_i \) are the fuel cost coefficients of generator \( i \).

(2) Emission objective function. The emission function can be expressed as the sum of all types of emission considered as \( NO_2, SO_2 \), thermal emission, etc., with suitable pricing or weighting on each pollutant emitted. In the present study, only one type of emission \( NO_2 \) is given as a function of generator output, that is, the sum of a quadratic and exponential function:

\[
f_2 = \sum_{i=1}^{n} \left[ 10^{-2}(\tilde{a}_i + \tilde{b}_i P_{Gi} + \tilde{c}_i P_{Gi}^2) + \tilde{\xi}_i e^{\tilde{\psi}_i P_{Gi}} \right],
\]

(2.2)

where \( f_2 \) is the amount of \( NO_2 \) emission (ton/h) and \( \tilde{a}_i, \tilde{b}_i, \tilde{c}_i, \tilde{\xi}_i, \) and \( \tilde{\psi}_i \) are the coefficients of the \( i \)th generator’s \( NO_2 \) emission characteristic.

2.2. Constraints

The optimization problem is bounded by the following constraints:

(1) Power balance constraint. The total power generated must supply the total load demand and the transmission losses

\[
\sum_{i=1}^{n} P_{Gi} - P_D - P_{Loss} = 0,
\]

(2.3)

where \( P_D \) is a total load demand, and \( P_{Loss} \) represents a transmission losses. The transmission losses are given by [23] as follows:

\[
P_{Loss} = \sum_{i=1}^{n} \sum_{j=1}^{n} [A_{ij}(P_i P_j + Q_i Q_j) + B_{ij}(Q_i P_j - P_i Q_j)],
\]

(2.4)

where \( P_i = P_{Gi} - P_{Di}, Q_i = Q_{Gi} - Q_{Di}, A_{ij} = (R_{ij}/V_i V_j) \cos(\delta_i - \delta_j), \) and \( B_{ij} = (R_{ij}/V_i V_j) \sin(\delta_i - \delta_j) \) such that \( n \) is a number of buses, \( R_{ij} \) is a series resistance connecting buses \( i \) and \( j \), and \( P_i \) is a real power injection at bus \( i \), \( Q_i \) a reactive power injection at bus \( i \); \( V_i \) is a voltage magnitude at bus \( i \); \( \delta_i \) is a voltage angle at bus \( i \).

2 Maximum and minimum limits of power generation.

The power generated \( P_{Gi} \) by each generator is constrained between its minimum and maximum limits, i.e., \( P_{Gi}^{min} \leq P_{Gi} \leq P_{Gi}^{max} \).\( Q_{Gi}^{min} \leq Q_{Gi} \leq Q_{Gi}^{max} \), and \( V_{min} \leq V_i \leq V_{max} \).

3 Security constraints. A mathematical formulation of the security constrained EELD problem would require a very large number of constraints to be considered. However, for typical systems the large proportion of lines has a rather small possibility of becoming overloaded. The EELD problem should consider only the small proportion of lines in violation, or near violation of their respective security limits which are identified as the critical lines. We consider only the critical lines that are binding in the optimal solution. The detection of the critical lines is assumed done by the experiences of the DM. An improvement in the security can be obtained by minimizing the following objective function,

\[
S = \sum_{q=1}^{m} \left( \frac{T_q(P_{Ci})}{T_q^{max}} \right)^2,
\]

where \( T_q(P_{Ci}) \) is the real power flow, \( T_q^{max} \) is the maximum limit of the real power flow of the \( q \)th line, and \( m \) is the number of monitored lines. The line flow of the \( q \)th line is expressed in terms of the control variables \( P_{Ci} \) by utilizing the generalized generation distribution factors (GGDF) in [24] and is given as follows:

\[
T_q(P_{Ci}) = \sum_{i=1}^{n} (D_{qi} P_{Ci}),
\]

where \( D_{qi} \) is the generalized GGDF for line \( q \) due to generator \( i \).

For secure operation, the transmission line loading \( S_l \) is restricted by its upper limit as

\[
S_l \leq S_{l\text{max}}, \quad l = 1, \ldots, n_l,
\]

where \( n_l \) is the number of transmission line.
In the following section we rewrite the EELD problem in mathematical formulation and using a weighting approach to transform it to a single objective optimization problem.

3. Multi-objective formulation of problem (EELD)

The mathematical formulation of the multi-objective problem (EELD) with \( n \)-buses and \( m \)-generator is the following multi-objective optimization problem:

minimize \( f_1 = \sum_{i=1}^{n} (a_i + b_i P_{Gi} + c_i P_{Gi}^2) \)

minimize \( f_2 = \sum_{i=1}^{n} [10^{-2}(\tilde{a}_i + \tilde{b}_i P_{Gi} + \tilde{c}_i P_{Gi}^2) + \xi \epsilon^P_{Gi}] \)

subject to \( \sum_{i=1}^{n} P_{Gi} - P_D - P_{Loss} = 0, \)

\[ P_{G_{\text{min}}} \leq P_{Gi} \leq P_{G_{\text{max}}}, \]

\[ Q_{G_{\text{min}}} \leq Q_{Gi} \leq Q_{G_{\text{max}}}, \]

\[ V_{i_{\text{min}}} \leq V_i \leq V_{i_{\text{max}}}, \]

\[ S_i \leq S_{\text{max}}, \]

where \( i = 1, \ldots, n \) and \( l = 1, \ldots, n_l \).

Using a weighting approach to transform problem (3.1) to a single-objective optimization problem which has the following form:

minimize \( f(x) = w_1 f_1 + w_2 f_2 \)

subject to \( \sum_{i=1}^{n} P_{Gi} - P_D - P_{Loss} = 0, \)

\[ P_{G_{\text{min}}} \leq P_{Gi} \leq P_{G_{\text{max}}}, \]

\[ Q_{G_{\text{min}}} \leq Q_{Gi} \leq Q_{G_{\text{max}}}, \]

\[ V_{i_{\text{min}}} \leq V_i \leq V_{i_{\text{max}}}, \]

\[ S_i \leq S_{\text{max}}, \]

where \( x = [P_{G_1}, \ldots, P_{G_n}, Q_{G_1}, \ldots, Q_{G_n}, V_1, \ldots, V_n, S_1, \ldots, S_n]^T, \)

\( w_1 + w_2 = 1, \) and \( w_1, w_2 \geq 0. \) The above problem can be written as follows:

minimize \( f(x) \)

subject to \( h(x) = 0, \)

\[ g(x) \leq 0, \]

where \( h(x) = \sum_{i=1}^{n} P_{Gi} - P_D - P_{Loss} \) and \( g(x) = [P_{G_{\text{min}}} - P_{Gi}, P_{G_{\text{max}}}, Q_{G_{\text{min}}} - Q_{Gi}, Q_{G_{\text{max}}}, V_{i_{\text{min}}} - V_i, \]

\[ V_{i_{\text{max}}} - S_i, S_{\text{min}}]^T. \]

The functions \( f(x) : \mathbb{R}^n \rightarrow \mathbb{R}, h(x) : \mathbb{R}^n \rightarrow \mathbb{R}, \) and \( g(x) : \mathbb{R}^{2n+m} \rightarrow \mathbb{R}^{6n+m} \) are twice continuously differentiable.

The Lagrangian function associated with problem (3.3) is the function

\[ L(x, \mu, \nu) = f(x) + \mu^T h(x) + \nu^T g(x), \]

where \( \mu \in \mathbb{R} \) and \( \nu \in \mathbb{R}^{6n+m} \) are the Lagrangian multiplier vectors associated with equality and inequality constraints, respectively.

Following [13], we define a 0–1 diagonal indicator matrix \( U(x) \in \mathbb{R}^{6n+m \times 6n+m} \) whose diagonal entries are

\[ u_e(x) = \begin{cases} 1 & \text{if } g_e(x) \geq 0, \\ 0 & \text{if } g_e(x) < 0 \end{cases} \]

Using the above matrix, we transform problem (3.3) to the following equality constrained optimization problem

minimize \( f(x) \)

subject to \( h(x) = 0, \)

\( \frac{1}{2} g(x)^T U(x) g(x) = 0. \)

Using a multiplier method, we transform the equality constrained optimization problem (3.6) to the following unconstrained optimization problem

minimize \( \Phi(x, \mu, \nu; r) = L(x, \mu, \nu) + \frac{r}{2} \parallel U(x) g(x) \parallel_2^2 \]

\[ + \frac{r}{2} \| h(x) \|_2^2, \]

subject to \( x \in \mathbb{R}^{3n+m}, \)

where \( r > 0 \) is a parameter usually called the penalty parameter. A detailed description of the main steps of the trust-region algorithm for solving the above problem and its an algorithmic framework is presented in the following section.

4. Trust-region algorithm outline

This section presents in details the description of our trust-region algorithm for solving problem (3.7).

4.1. Computing a trial step

We compute the trial step \( z_k \) by solving the following trust-region subproblem

minimize \( l_k + \nabla l_k^T z_k + \frac{1}{2} z_k^T H_k z_k + \frac{\rho_k}{2} \| U_k (g_k + \nabla g_k^T z_k) \|^2 \]

\[ + \frac{r_k}{2} \| h_k + \nabla h_k^T z_k \|^2 \]

subject to \( \| z \| \leq \Delta_k, \)

where \( H_k \) is the Hessian matrix of the Lagrangian function \( L(x_k, \mu_k, \nu_k) \) or an approximation to it. Since our convergence theory is based on the fraction of Cauchy decrease condition, therefore a generalized dogleg algorithm introduced by Steihaug [25] and Toint [26] can be used to compute the trial step.

4.2. Testing the step and updating \( \Delta_k \)

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 Lagrangian multipliers \( \mu_{k+1} \) and \( \nu_{k+1} \) are needed. Our way of evaluating the two Lagrangian multipliers \( \mu_{k+1} \) and \( \nu_{k+1} \) is presented in Step 5 of Algorithm (4.1) below.

Let \( \mu_{k+1} \) and \( \nu_{k+1} \) be the estimation of the two Lagrangian multiplier vectors. We test whether the point \( (x_k + z_k, \mu_{k+1}, \nu_{k+1}) \) will be taken as a next iterate.
The actual reduction in the merit function is defined as

\[
A_{redk} = L(x_k, \mu_k, v_k) - L(x_{k+1}, \mu_k, v_k) - \Delta \mu_k h_{k+1} + \Delta v_k^T U_k g_k
+ \frac{\rho_k}{2} [L_k^T U_k g_k - L_{k+1}^T U_k g_k + \frac{r_k}{2} ||h_k||^2 - ||h_{k+1}||^2].
\]  

(4.2)

where \(\Delta \mu_k = (\mu_{k+1} - \mu_k)\) and \(\Delta v_k = (v_{k+1} - v_k)\).

The predicted reduction in the merit function is defined to be

\[
Pred_k = \hat{q}_k(0) - \hat{q}_k(z_k) - \Delta \mu_k^T h_k + \nabla h_k^T g_k - \Delta v_k^T U_k g_k
+ \frac{r_k}{2} [||h_k||^2 - ||h_k + \nabla h_k^T z_k||^2].
\]  

(4.3)

where

\[
\hat{q}_k(z) = L_k + \nabla s^T z + \frac{1}{2} z^T H_k z + \frac{\rho_k}{2} ||U_k g_k + \nabla g_k^T z||^2.
\]  

(4.4)

After computing a trial step and updating the Lagrangian multipliers, the penalty parameter is updated to ensure that \(Pred_k \geq 0\). To update \(\rho_k\), we use a scheme that has the flavor of the scheme proposed by [14]. This scheme is described in Step 6 of Algorithm 4.1. After that, the step is tested to know whether it is accepted. This is done by comparing \(A_{redk}\) against \(A_{redk}\).

If \(A_{redk}/Pred_k < \eta_1\) 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 \(\Delta_k\) is decreased by setting \(\Delta_k = \alpha_1 \Delta_k\), where \(\alpha_1 \in (0, 1)\), and another trial step is computed using the new trust-region radius.

If \(A_{redk}/Pred_k \geq \eta_1\), then the step is accepted. Our theory requires that at the beginning of the next iteration, \(\Delta_k + 1\) must be greater than or equal to \(\Delta_{min}\), where \(\Delta_{min}\) is a positive constant chosen at the beginning of the algorithm. That is, \(\Delta_k\) can be reduced below \(\Delta_{min}\) 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 \(z_k\).

Our way of evaluating the trial steps and updating the trust-region radius is presented in Section 7 of Algorithm (4.1). After accepting the step, we update the parameter \(\rho_k\) and the Hessian matrix \(H_k\). To update \(\rho_k\), we use a scheme suggested by [27]. This scheme is described in Step 8 of Algorithm 4.1.

Finally, the algorithm is terminated when either \(\{z_k\} \leq \varepsilon_1\) or \(||\nabla s_k + \nabla g_k + U_k g_k + \nabla g_k^T z_k|| \leq \varepsilon_2\), for some \(\varepsilon_1, \varepsilon_2 > 0\).

4.3. Main algorithm

A formal description of our trust-region algorithm for solving problem (3.3) is presented in the following algorithm:

Algorithm 4.1. (The Main Algorithm)

Step 0. (Initialization)

Given \(x_0 \in \mathbb{R}^{n+1}\), Compute \(U_0\). Evaluate \(v_1\) and \(\mu_1\) (see Step 5 with \(k = 0\) and \(\rho_0 = (0, 0, \ldots, 0)\)). Set \(\rho_1 = 1, \rho_2 = 1, \sigma_1 = 1, \) and \(\beta = 0.1\). Choose \(\varepsilon_1 = \varepsilon_2 = 10^{-4}, \alpha_1 = 0.05, \alpha_2 = 2, \eta_1 = 10^{-4}, \) and \(\eta_2 = 0.5\) such that \(\varepsilon_1 > 0, \varepsilon_2 > 0, \alpha_1 < 1 < \alpha_2, \) and \(0 < \eta_1 < \eta_2 < 1\). Set \(\Delta_{min} = 10^{-3}\) and \(\Delta_{max} = 10^{-1}\), such that \(\Delta_{min} \leq \Delta_k \leq \Delta_{max}\). Set \(k = 1\).

Step 1. (Test for convergence)

If \(||\nabla s_k + \nabla g_k + U_k g_k + \nabla g_k^T z_k|| \leq \varepsilon_2\), then terminate the algorithm.

Step 2. (Compute a trial step)

a) Compute the step \(z_k\) by solving (4.1)

b) Set \(x_{k+1} = x_k + z_k\).

Step 3. (Test for termination)

If \(||z_k|| \leq \varepsilon_1\), then terminate the algorithm.

Step 4. (Update the active set)

Compute \(U_{k+1}\).

Step 5. (Compute the Lagrangian multipliers \(\mu_{k+1}\) and \(v_{k+1}\))

a) Compute \(v_{k+1}\) by solving

\[
\min_{U_{k+1}\in\Omega} \frac{1}{2} ||U_{k+1} g_k + \nabla g_k^T z_k||^2,
\]

subject to \(U_{k+1}, v_{k+1} \geq 0\), and set the rest of the components of \(v_{k+1}\) to zero.

b) If \(||\nabla f_k + \nabla s_k + \nabla g_k + U_k g_k + \nabla g_k^T z_k|| \leq \varepsilon_2\), then set \(\mu_{k+1} = \mu_k\).

Else, compute \(\mu_{k+1}\) by solving

\[
\min ||\nabla f_{k+1} + \nabla s_{k+1} + v_{k+1} + h_{k+1}||^2.
\]

Step 6. (Update the penalty parameter \(\rho_k\))

a) Set \(\rho_k = \max(\rho_k, \rho_0)\).

b) If \(\rho_{k-1} \leq \frac{1}{4} \rho_k \leq \rho_0 \leq \rho_{k-1}\), then set \(\rho_k = \rho_{k-1} + \alpha_1 \rho_{k-1}\), where \(\alpha_1 \in (0, 1)\), and another trial step is computed using the new trust-region radius.

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

If \(\|z_k\| < \varepsilon_1\), then reduce the trust-region radius by setting \(\Delta_k = \alpha_1 \|z_k\|\), and go to step 2.

Else if \(\|z_k\| \geq \varepsilon_1\) then

Accept the step: \(x_{k+1} = x_k + z_k\).

Set the trust-region radius: \(\Delta_k+1 = \max(\Delta_k, \Delta_{min})\).

Else

Accept the step: \(x_{k+1} = x_k + z_k\).

Set the trust-region radius: \(\Delta_k+1 = \min(\Delta_{max}, \max(\Delta_k, \alpha_1 \Delta_k))\).

End if.

Step 8. (Update the parameters \(\rho_k\) and \(\sigma_k\))

a) Set \(\rho_k+1 = \rho_k + \alpha_1 \rho_k\).

b) If \(\hat{q}_k(0) - \hat{q}_k(z_k) - \Delta \mu_k^T h_k + \nabla h_k^T g_k - \Delta v_k^T U_k g_k \leq \sigma_1 (||U_k g_k + \nabla g_k^T z_k|| + \|\nabla s_k + \nabla g_k + U_k g_k + \nabla g_k^T z_k|| + \Delta_k),\) then set \(\rho_{k+1} = \rho_k + \alpha_1 \rho_k\) and \(\sigma_{k+1} = \frac{1}{4} \sigma_k\).

End if.

Step 9. Set \(k = k + 1\) and go to Step 1.

5. Implementation of the proposed approach

The proposed approach is applied to the standard IEEE 30-bus 6-generator test system to investigate the effectiveness of this approach. The values of fuel cost and emission coefficient are given in Table 1. The values of minimum fuel cost and minimum NOx emission are given in Tables 2 and 3, respectively.
In this work, our program was written in MATLAB and run under MATLAB 7 with machine epsilon about $10^{-16}$. 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

We compare our obtained results with different previous works, which introduced different algorithms handle the same problem (the standard IEEE 30-bus 6-generator test system) to investigate the effectiveness of this approach. These approaches are (a) Non-dominated Strong Genetic Algorithm (NSGA) [7], (b) Niched Pareto Genetic Algorithm (NPGA) [8], (c) Strength Pareto Evolutionary Algorithm (SPEA) [9], and (d) Hybrid Multi-objective Evolutionary Algorithm (HMEA) [28]. For comparison purposes with the reported results, the system is considered as losses and the security constraints is released. The results declare the implementation of trust-region globalization strategy to solve the multi-objective problem (EELD) modifies the local method in such a way that it is guaranteed to converge at all even if the starting point is far away from the solution and improves the solution quality for the same approach. Also, our approach could discuss the effects of changing weights on fuel cost, as well as the emission. As one weight is changed linearly such as $\{w_1 = 0.0, 0.1, 0.2, \ldots, 1\}$ and $\{w_2 = 1.0, 0.9, 0.8, \ldots, 0.0\}$, then we can obtain the best range of the weights to help the decision maker to choose the most prefer weights for the objective functions. We do this idea as a some sort of parametric study to help the D.M. if he/she not experience enough in this field. That mean that the D.M. can choose the weights from the following ranges $0.4 \leq w_1 \leq 0.6$ and $0.4 \leq w_2 \leq 0.6$. where $w_1 + w_2 = 1$. These ranges guarantee that minimum of both fuel cost and $NO_2$ emission. Finally our algorithm the fuel cost function and keep the $NO_2$ emission function in its ranges comparable with all reported results. The values of minimum fuel cost and the minimum $NO_2$ emission are given in Tables 2 and 3, respectively (Fig. 1).

![Plot showing $f_1$ at $w_1 = 0.0, 0.1, 0.2, \ldots, 1$](image1)

![Plot showing $f_2$ at $w_1 = 0.0, 0.1, 0.2, \ldots, 1$](image2)

7. Conclusions

The proposed approach was applied to economic emission load dispatch optimization problem which formulated as multi-objective optimization problem as well as computing the fuel cost and emission. The presented algorithm considered as globally

<table>
<thead>
<tr>
<th>$p_0$</th>
<th>NSGA</th>
<th>NPGA</th>
<th>SPEA</th>
<th>HMEA</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_{c1}$</td>
<td>0.1168</td>
<td>0.1245</td>
<td>0.1086</td>
<td>0.1737</td>
<td>0.1114</td>
</tr>
<tr>
<td>$p_{c2}$</td>
<td>0.3165</td>
<td>0.2792</td>
<td>0.3056</td>
<td>0.3568</td>
<td>0.3011</td>
</tr>
<tr>
<td>$p_{c3}$</td>
<td>0.5441</td>
<td>0.6284</td>
<td>0.5818</td>
<td>0.5411</td>
<td>0.5218</td>
</tr>
<tr>
<td>$p_{c4}$</td>
<td>0.9447</td>
<td>1.2064</td>
<td>0.9846</td>
<td>0.9890</td>
<td>1.0002</td>
</tr>
<tr>
<td>$p_{c5}$</td>
<td>0.5498</td>
<td>0.4693</td>
<td>0.5288</td>
<td>0.4529</td>
<td>0.5281</td>
</tr>
<tr>
<td>$p_{c6}$</td>
<td>0.3964</td>
<td>0.39993</td>
<td>0.3584</td>
<td>0.3705</td>
<td>0.3613</td>
</tr>
<tr>
<td>Best cost</td>
<td>608.245</td>
<td>608.147</td>
<td>607.807</td>
<td>606.012</td>
<td>602.55</td>
</tr>
<tr>
<td>Corresponding emission</td>
<td>0.21664</td>
<td>0.22364</td>
<td>0.22015</td>
<td>0.20080</td>
<td>0.22202</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$p_0$</th>
<th>NSGA</th>
<th>NPGA</th>
<th>SPEA</th>
<th>HMEA</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_{c1}$</td>
<td>0.4113</td>
<td>0.3923</td>
<td>0.4043</td>
<td>0.3675</td>
<td>0.4002</td>
</tr>
<tr>
<td>$p_{c2}$</td>
<td>0.4591</td>
<td>0.4700</td>
<td>0.4525</td>
<td>0.4904</td>
<td>0.4434</td>
</tr>
<tr>
<td>$p_{c3}$</td>
<td>0.5117</td>
<td>0.5565</td>
<td>0.5525</td>
<td>0.5177</td>
<td>0.5025</td>
</tr>
<tr>
<td>$p_{c4}$</td>
<td>0.3724</td>
<td>0.3695</td>
<td>0.4079</td>
<td>0.4512</td>
<td>0.4009</td>
</tr>
<tr>
<td>$p_{c5}$</td>
<td>0.5810</td>
<td>0.5599</td>
<td>0.5468</td>
<td>0.5215</td>
<td>0.5203</td>
</tr>
<tr>
<td>$p_{c6}$</td>
<td>0.5304</td>
<td>0.5163</td>
<td>0.5005</td>
<td>0.5304</td>
<td>0.5268</td>
</tr>
<tr>
<td>Best emission</td>
<td>0.19432</td>
<td>0.19424</td>
<td>0.19422</td>
<td>0.1880</td>
<td>0.2000</td>
</tr>
<tr>
<td>Corresponding cost</td>
<td>647.251</td>
<td>649.964</td>
<td>642.603</td>
<td>644.5346</td>
<td>643.0562</td>
</tr>
</tbody>
</table>
search technique to get a pareto-optimal solution. The following points are the significant contributions of this paper.

- This approach is applied to solve multi-objective problem (EELD) with no limitation to the no of objective functions.
- The proposed approach is efficient for solving ill-defined systems and non-convex multi-objective optimization problems.
- The presented algorithm considered as globally search technique to get a pareto-optimal solution.
- The proposed approach is efficient for solving no convex multi-objective optimization problems where multiple Pareto-optimal solutions can be found in one run.
- The proposed approach helps the decision maker to choose the most prefer weights for the objective functions.
- For future work, there are many questions that should be answered. Although we have implemented the algorithm and tested it, we believe that the implementation of the algorithm should be refined with efficiency in mind. In particular, a better way of solving the trust-region subproblems that can handle large-scale problems should be used.
- Updating the Lagrangian multiplier is another point that needs to be refined. This indeed will reduce the cost of the computation of the steps.

Acknowledgements

The authors are grateful to the anonymous reviewers for their valuable comments and helpful suggestions which greatly improved the paper’s quality.

References

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