Relaxed I-SHOT trust-region algorithm for solving multi-objective economic emission load dispatch problem

B. El-sobky, Y. Abo-elnaga, A. Elsaïd and A. Foda

Abstract

In this paper, a new algorithm for tackling nonlinear constrained multi-objective optimization problems is introduced. This algorithm, which we call relaxed the interactive sequential hybrid optimization technique (relaxed I-SHOT), is based on a hybrid method between the I-SHOT method and step method (STEM) to transform a multi-objective problem into a single-objective one. An active set strategy is used together with a suggested penalty method to transform a single-objective constrained optimization problem into unconstrained one. A trust-region globalization strategy is added to the algorithm to solve the obtained unconstrained problem to ensure global convergence. The suggested approach is utilized to solve the multi-objective economic emission load dispatch (EELD) problems to assert the effectiveness of the proposed algorithm. The proposed algorithm is tested on the standard IEEE 30-bus 6-generator test system. The results of the proposed approach are compared against those reported in the literature. The comparison asserts the potential of the proposed algorithm to solve the EELD problem.

Keywords

Multi-objective optimization; economic emission load dispatch; relaxed I-SHOT; trust-region; penalty method; active set

1. Introduction

A majority of the engineering decision-making problems have multiple design objectives and these problems are called multi-objective problems. There are many methods for solving such problems, which are usually classified into four classes: interactive methods, a posteriori methods, a priori methods and no-preference methods. Over the years, variant optimization methods from the previous classes were used to solve multi-objective optimization problems [1–5]. To improve the resultant solution from these methods, researchers tend to make hybrid methods from methods in one class or different classes so as to combine the advantages of these methods.

We are concerned with a new hybrid method which is used to convert a multi-objective optimization problem to a single one. This method (which we call relaxed I-SHOT) is a hybrid method of interactive sequential hybrid optimization technique (I-SHOT) and step method (STEM). In the interactive methods, the designer’s preferences take part in the formation of a sample of the Pareto solutions, then an interaction between the designer and the analyst takes place to generate more points on the Pareto set until a designed solution is selected by the designer. In the relaxed I-SHOT method, the relaxation technique in the STEM is added to the main steps of the I-SHOT method, which provides an interaction between the decision-maker (DM) and the analyst to reach a sufficient Pareto-optimal solution. The I-SHOT method deals with all objectives with the same strategy, but with the relaxation technique, the DM is able to classify the objectives in every iteration and improve some unacceptable objective values by relaxation or increment of the other acceptable objectives’ values. The relaxed I-SHOT method inherits the ability of the classical I-SHOT to detect non-convex regions of the Pareto set.

The multi-objective optimization problem is solved using a trust-region globalization strategy. The globalized strategy converges from any starting point and this is a modification to the local strategy. By using this approach, there is no limitation to the number of objective functions. The proposed globalization trust-region method is efficient for solving non-convex multi-objective optimization problems and ill-defined systems.

In this work, we convert the single-objective constrained problem to an unconstrained problem using an active set strategy in [6] combined with the penalty method. The main advantage of the suggested active set is that it is identified and updated naturally by the step. Two penalty parameters are used in this work: one for the active set transformed inequality constraints and the other for equality constraints. Some penalty
functions have been suggested and many contributions addressing the convergence of these methods have been made [7].

The trust-region strategy used for solving the single-objective constrained and unconstrained optimization problems and multi-objective problems has proved to be very successful both theoretically and practically [6,8,9].

To ensure the effectiveness of our proposed algorithm, the algorithm is applied to solve the EELD problem. The EELD problem is the process of satisfying the required power load demand using the available generating units while minimizing the cost of operation and all unit operational constraints are also satisfied. In addition, the increasing public awareness of the environmental protection has forced the utilities to modify their design to reduce atmospheric emissions of the thermal power plants [1,10]. Hence, the EELD problem is formulated as a constrained multi-objective problem with conflicting objectives, minimize the fuel cost and emission level simultaneously.

We are concerned with the emission as an objective function, and the multi-objective EELD problem is converted to a corresponded single-objective problem using our new hybrid approach.

The following notations are used in this paper. The sequence of points generated by the algorithm is denoted . A subscripted function means function values at a particular point. For example, , , and so on. Finally, all norms are -norms.

This paper is organized as follows. In Section 2, we introduce in detail the new hybrid method for solving multi-objective problems. Section 3 is devoted to the solution of the single-objective problem obtained from the proposed multi-objective algorithm. The mathematical formulation of the EELD problem is presented in Section 4. In Section 5, implementation of the proposed algorithm for treatment of the EELD problem is presented. In Section 6, numerical results and discussions are presented. Finally, the conclusion is given in Section 7.

2. Relaxed I-SHOT for solving multi-objective problems

2.1. Formulation and terminology

A general mathematical form for multi-objective optimization problem with objective functions can be formulated as follows:

\[
\begin{align*}
\text{minimize} & \quad f_j(x), \quad j = 1, \ldots, n_j, \\
\text{subject to} & \quad h_e(x) = 0, \quad e = 1, \ldots, n_e, \\
& \quad g_i(x) \leq 0, \quad i = 1, \ldots, n_i,
\end{align*}
\]

where is a design vector, is the th objective function to be minimized, is the th equality constraints and is the th inequality constraints.

One of the important methods for solving such problems is the interactive method. The main idea in interactive algorithms is that the DM interacts with the analyst or with the utilized computer programme. The analyst solves to find an initial solution and then, with the help of the DM, obtains the new solution. If the DM is satisfied with the new solution (or one of them), then the process is stopped. Otherwise the analyst makes a new interaction with the DM.

We present two of the important algorithms for solving multi-objective optimization problems, which are based on the interactive approach, the I-SHOT method and STEM. Then, we introduce a new algorithm based on the interactive approach, which is a hybrid method from these two methods.

2.2. I-SHOT method

The I-SHOT method is a hybrid method combines the advantage upon the weighting method and -constrained method, see ([11,12]). In the weighting method, the multi-objective problem is transformed into a single-objective problem by making a linear combination of the objective functions with weights, and in the -constrained method by constraining all objective functions except one and then solve the problem by minimizing that remaining objective. The weighting method and the -constrained cannot detect points in a non-convex Pareto set. In this method, by using the interactive approach, problem (1) is transformed into the following form:

\[
\begin{align*}
\text{minimize} & \quad F(x) = \sum_{j=1}^{n_j} w_j \left( f_j(x) - \hat{e}_{j}^{\text{good}} \right) \\
\text{subject to} & \quad f_j(x) \leq \hat{e}_j, \quad j = 1, \ldots, n_j, \\
& \quad h(x) = 0, \\
& \quad g(x) \leq 0, \\
& \quad \hat{e}_j \in [e_{j}^{\text{good}}, \hat{e}_{j}^{\text{bad}}],
\end{align*}
\]

where \(e_{j}^{\text{good}}, \hat{e}_{j}^{\text{bad}}\) are calculated using payoff table [12] and

\[
\begin{align*}
w_j = \begin{cases} 
W_j + \frac{1 - \hat{W}}{\hat{W}} w_j & \text{if } \hat{W} > 1, \\
w_j & \text{if } \hat{W} = 1, \\
\frac{W_j + \frac{1 - \hat{W}}{\hat{W}} (1 - w_j)}{n_j - W} & \text{if } \hat{W} < 1,
\end{cases}
\end{align*}
\]

such that \(W_j = \text{random}(0, 1)\) and \(\hat{W} = \sum_{j=1}^{n_j} w_j\).

If the solution of (2) is acceptable for the DM, then the process is stopped, else an interaction between the analyst and the DM is established to make a reduction.
to the range \( [\epsilon^\text{good}, \epsilon^\text{bad}] \) by using the following formula:

\[
\begin{align*}
\epsilon^\text{new, good} &= \epsilon^\text{good} + \frac{\Delta \epsilon_j}{\tilde{r}}, \\
\epsilon^\text{new, bad} &= \epsilon^\text{bad} - \frac{\Delta \epsilon_j}{\tilde{r}},
\end{align*}
\]

where \( \Delta \epsilon_j = \epsilon^\text{bad} - \epsilon^\text{good} \) and \( \tilde{r} \) is the reduction factor. Then Equation (2) is solved again until the DM is satisfied by the Pareto-optimal solution.

The main advantage of the I-SHOT method is that it can obtain Pareto solutions for a non-convex problem. The method gives the DM the opportunity to control the trade-off analysis to a designed solution (final solution). Also, by using this method, proper Pareto points are detected. But in real-world multi-objective problems, this method has a major defect. The defect is that, in the solution algorithm of the method as described in [11], the algorithm interacts with all objectives at the same way in reduction of the range of the Pareto set by using the same value of the reduction factor \( \tilde{r} \) and in updating \( \hat{\epsilon}_j \). In real problems, the range of the objectives \( [\epsilon^\text{good}, \epsilon^\text{bad}] \) is different in width from one objective function to another and the solutions occur in every iteration of the I-SHOT algorithm; DM can be satisfied with some objective values and need to modify only the other objectives in the next iteration. The I-SHOT method in this form does not allow the DM to classify the objective functions to another and the solutions occur in every iteration of the I-SHOT algorithm; DM can be satisfied with the generated Pareto set then the process is stopped. Otherwise the DM makes a new classification to the objective functions and the problem is solved for the next iteration. The procedure is repeated until the DM is satisfied by the current objective vector and all of its components. The procedure must also be stopped if the DM is not satisfied with any of the components of the current objective vector. In this case, the STEM cannot find a satisfactory solution. The main defects of the STEM are that it is classified as an ad hoc method and the solutions calculated using it are not necessarily Pareto optimal, but it can obtain weakly Pareto-optimal solutions.

### 2.3. STEM method

STEM is one of the first interactive methods developed for multi-objective optimization problems. In this method, after obtaining the initial solution by using the weighted Tchebycheff method [12], the DM classifies the objective functions into two categories, objective functions with acceptable values which are satisfactory to the DM \( f_j(j \in \mathcal{J}^+, \mathcal{J}^-) \) and other objective functions with unacceptable values, \( f_j(j \in \mathcal{J}^-, \mathcal{J}^-) \), where \( \mathcal{J}^+ \) and \( \mathcal{J}^- \) is the set that contains the numbers of the objective functions with acceptable and unacceptable values, respectively such that \( \mathcal{J}^+ \cup \mathcal{J}^- = \{1, \ldots, nj\} \). So the problem can be solved by making an increment in values of \( f_j(j \in \mathcal{J}^+) \) to achieve a decrement in the unacceptable objective functions values \( f_j(j \in \mathcal{J}^-) \). This process is called the relaxation technique, which is formulated in [12] as follows:

\[
\begin{align*}
\text{minimize} \quad & \max \{w_j | f_j(x) - z_j^*| \} \quad j = 1, \ldots, nj, \\
\text{subject to} \quad & f_j(x) \leq \hat{\epsilon}_j \quad \text{for all} \quad j \in \mathcal{J}^-, \\
& f_j(x) \leq \hat{f}_j(x_k) \quad \text{for all} \quad j \in \mathcal{J}^-, \\
& x \in D,
\end{align*}
\]

where \( w_j \) is the weight coefficient (which is set equal to zero for the relaxed functions \( f_j(j \in \mathcal{J}^-) \)), \( z_j^* \) is the ideal objective vector (which should be known to apply this method) and \( x_k \) is the solution of the previous iteration.

By solving Equation (5), we calculate a new solution; if the DM is satisfied, then the process is stopped. Otherwise the DM makes a new classification to the objective functions and the problem is solved for the next iteration. The procedure is repeated until the DM is satisfied by the current objective vector and all of its components. The procedure must also be stopped if the DM is not satisfied with any of the components of the current objective vector. In this case, the STEM cannot find a satisfactory solution. The main defects of the STEM are that it is classified as an ad hoc method and the solutions calculated using it are not necessarily Pareto optimal, but it can obtain weakly Pareto-optimal solutions.

### 2.4. Relaxed I-SHOT algorithm

Relaxed I-SHOT method is a new hybrid method from the I-SHOT and STEM, which has the main structure of I-SHOT method but it adopts the relaxation technique in the STEM. Generally, Pareto-optimal solutions set generated is not predicted. So in some cases, the DM may not be satisfied with all solutions introduced because the solutions generated are all satisfied for some objectives’ value and the other is not. By adding the relaxation technique to the I-SHOT method, the DM has the ability to improve the unsatisfied objectives’ value by an acceptable increment in the satisfied objectives’ value, as described in step 7 in the following algorithm.

The following algorithm presented a main description of the proposed relaxed I-SHOT method:

#### Algorithm 1 (Relaxed I-SHOT Algorithm):

**Step 1** (Compute \( \epsilon_j \) and \( \hat{\epsilon}_j \))

Solve problem (1) for all \( j = 1, \ldots, nj \) to construct the payoff table.

\( \epsilon_j \) and \( \hat{\epsilon}_j \) are calculated using the payoff table such that \( \epsilon_j = z^*_j \) and \( \hat{\epsilon}_j = \text{upper bound of the objective} \ j \text{ values which is the maximal value of the column} \ j \text{ in the payoff table.} \)

**Step 2** (Ask the DM)

The DM is required to choose number of weight vectors (the number of Pareto optimal solutions to be generated in every iteration), to select \( \epsilon^\text{good, new} \in [\epsilon_j, \hat{\epsilon}_j] \), \( \epsilon^\text{bad, new} \in [\epsilon_j, \hat{\epsilon}_j] \) and to select \( \hat{\epsilon}_j \in [\epsilon^\text{good, new}, \epsilon^\text{bad, new}] \).

**Step 3** (Generate weighting vectors)

Solve problem (3) to generate the weighting vectors.

**Step 4** (Solve the I-SHOT hybrid problem)

Solve the hybrid I-SHOT problem (2).

**Step 5** (Test for stopping)

If the DM is satisfied with the generated Pareto set then the process is stopped. If the DM is not satisfied with all objective functions value in the Pareto set, go to Step 6.
If the DM is satisfied with only some objective functions value in the Pareto set, go to Step 7.

**Step 6 (Compute a new value for \( \epsilon_{\text{good}} \) and \( \epsilon_{\text{bad}} \))**

Update \( \epsilon_{\text{good}} \) and \( \epsilon_{\text{bad}} \) using (4).

Set \( \epsilon_{\text{good}} = \epsilon_{\text{new}}^{\text{good}} \) and \( \epsilon_{\text{bad}} = \epsilon_{\text{new}}^{\text{bad}} \).

Go to step 4.

**Step 7 (Ask the DM)**

The DM is asked to classify the objective functions into two categories: \( f_j \in J^+ \) which have acceptable values in solving problem (2) in the previous iteration, \( f_j \in J^- \) which have unacceptable values in solving problem (2) in the previous iteration and the relaxation value \( p \) for the acceptable objective values.

If \( j \in J^+ \) then \( \tilde{\epsilon}_j = \bar{\epsilon}_j + p \epsilon_{\text{new}}^{\text{good}} = \epsilon_{\text{good}} \) and \( \epsilon_{\text{new}}^{\text{bad}} = \epsilon_{\text{bad}} \),

where \( p \) is a relaxation parameter and \( \tilde{\epsilon}_j = \tilde{\epsilon}_j \).

Update \( \epsilon_{\text{good}} \) and \( \epsilon_{\text{bad}} \) using Equation (4).

Set \( \epsilon_{\text{good}} = \epsilon_{\text{new}}^{\text{good}} \) and \( \epsilon_{\text{bad}} = \epsilon_{\text{new}}^{\text{bad}} \).

**Step 8 (Solve the relaxed I-SHOT problem)**

Solve the Relaxed I-SHOT single objective constrained problem which has the following form

\[
\begin{align*}
\text{minimize} & \quad F(x) = \sum_{j=1}^{n_j} w_j \left( \frac{f_j(x) - \epsilon_{\text{good}}}{\epsilon_{\text{bad}} - \epsilon_{\text{good}}} \right), \\
\text{subject to} & \quad f_j(x) \leq \tilde{\epsilon}_j^h \quad \text{for all } j \in J^+, \\
& \quad f_j(x) \leq \tilde{\epsilon}_j \quad \text{for all } j \in J^-, \\
& \quad h(x) = 0, \\
& \quad g(x) \leq 0,
\end{align*}
\]

where \( \tilde{\epsilon}_j = \min(f_j(x_k)) \) for all \( j \in J^- \) and \( \tilde{\epsilon}_j = [\tilde{\epsilon}_j^h, \tilde{\epsilon}_j] \).

Go to step 5.

Repeat until the DM is satisfied with the proper Pareto set generated.

Using the above algorithm, the constrained multi-objective optimization problem is converted to a constrained single-objective problem.

3. Solving single-objective problem

The equality and inequality single-objective constrained problem (6) is converted into an unconstrained problem using an active set strategy together with the penalty method.

3.1. Active set strategy

The main idea of the active set strategy is to create the active inequality constraints and convert them to equalities at every iteration. Then, by using well-developed methods, the equality constrained problem is solved (see, for example, [6]). The main idea in the penalty method used for solving the equality constrained optimization problem is to replace the problem with a sequence of unconstrained optimization problems while the penalty parameter does not tend to infinity [13,14].

The single-objective constrained optimization problem (6) can be formulated as follows:

\[
\begin{align*}
\text{minimize} & \quad F(x), \\
\text{subject to} & \quad h(x) = 0, \\
& \quad g(x) \leq 0,
\end{align*}
\]

where \( F(x) = \sum_{j=1}^{n_j} w_j \left( \frac{f_j(x) - \epsilon_{\text{good}}}{\epsilon_{\text{bad}} - \epsilon_{\text{good}}} \right), \) and \( \tilde{g}(x) = [f_j(x) - \tilde{\epsilon}_j, g(x)] \). The functions \( F(x) : \mathbb{R}^n \rightarrow \mathbb{R}, h(x) : \mathbb{R}^n \rightarrow \mathbb{R}^m, \) and \( \tilde{g}(x) : \mathbb{R}^n \rightarrow \mathbb{R}^{n+1} \) are twice continuously differentiable. Following the active set strategy in [6], we define a \( 0-1 \) diagonal indicator matrix \( D(x) \in \mathbb{R}^{nj+n+1} \), whose diagonal entries are

\[
d_e(x) = \begin{cases} 
1 & \text{if } \tilde{g}_e(x) \geq 0, \\
0 & \text{if } \tilde{g}_e(x) < 0.
\end{cases}
\]

Using the above matrix, problem (7) is transformed into the following equality constrained optimization problem:

\[
\begin{align*}
\text{minimize} & \quad F(x), \\
\text{subject to} & \quad h(x) = 0, \\
& \quad \frac{1}{2} \tilde{g}(x)^T D(x) \tilde{g}(x) = 0.
\end{align*}
\]

Using the proposed penalty method, the equality constrained optimization problem (9) is transformed into the following unconstrained optimization problem with two penalty active inequality constrains and equality constrains parameters, \( \rho > 0, r > 0 \), respectively. Two penalty parameters used in this algorithm to reduce the number of computations by ensuring that the predicted reduction is positive before testing the step. Using the proposed penalty method approach, the equality constrained optimization problem (9) is transformed into the following unconstrained optimization problem:

\[
\begin{align*}
\text{minimize} & \quad \phi(x; \rho; r) = F(x) + \frac{\rho}{2} \| D(x) \tilde{g}(x) \|_2^2 \\
& \quad + \frac{r}{2} \| h(x) \|_2^2,
\end{align*}
\]

subject to \( x \in \mathbb{R}^{2n} \).

In the following section, we present a detailed description of the main steps of our trust-region algorithm for solving multi-objective problem formulated as unconstrained optimization problem (10).

3.2. Trust-region algorithm outline

In this section, we present the description of the active set trust-region algorithm which is used to solve the single-objective problem (10). Then we present the main algorithm that is used to solve the multi-objective optimization problem.
3.2.1. Computing a trail step
By solving the following trust-region subproblem, we compute the trail step \( s_k \).

\[
\begin{align*}
\text{minimize} & \quad F_k + \nabla F_k^T s + \frac{1}{2} s^T H_k s \\
\quad & + \frac{\rho_k}{2} \|D_k (\tilde{g}_k + \nabla g_k^T s_k)\|^2 \\
\text{subject to} & \quad \|s\| \leq \delta_k,
\end{align*}
\]

(11)

where \( H_k \) is the Hessian matrix of the function \( F(x) \) or an approximation to it. Since our convergence theory is based on the fraction of cauchy decrease condition, a generalized dogleg algorithm introduced by Toint [15] and Steihaug [16] can be used to compute the step.

3.2.2. Testing the trail step and updating \( \delta_k \)
Once \( s_k \) is computed, it needs to be tested to determine whether it is accepted. We use \( \phi(x; \rho; r) \) as a merit function.

The actual reduction in the merit function according to change form \( x_k \) to \( x_{k+1} \) is defined as

\[
A_{red_k} = F(x_k) - F(x_{k+1})
\]

\[
= \frac{\rho_k}{2} \left[ \|D_k \tilde{g}_k - (\tilde{g}_k^T D_k + D_k \tilde{g}_k) s_k \|^2 \right]
\]

\[
+ \frac{\rho_k}{2} \left[ \|D_k (\tilde{g}_k + \nabla g_k^T s_k)\|^2 \right]
\]

\[
+ \frac{\rho_k}{2} \|\tilde{h}_k\|^2 - \|\tilde{h}_k + \nabla h_k^T s_k\|^2.
\]

(12)

The predicted reduction in the merit function is defined as

\[
Pred_k = -\nabla F_k^T s_k - \frac{1}{2} s_k^T H_k s_k
\]

\[
+ \frac{\rho_k}{2} \left[ \|D_k \tilde{g}_k\|^2 - \|D_k (\tilde{g}_k + \nabla g_k^T s_k)\|^2 \right]
\]

\[
+ \frac{\rho_k}{2} \|\tilde{h}_k\|^2 - \|\tilde{h}_k + \nabla h_k^T s_k\|^2.
\]

(13)

After computing a trial step, to ensure that \( Pred_k \geq 0 \), the penalty parameter \( r_k \) is updated using the scheme presented in [17] and it is described in step 4 of algorithm 2. After that, we use the following algorithm to test the calculated step to know whether it is accepted.

Algorithm 2 (Test the computing step):
0 < \( \alpha_1 < 1 < \alpha_2 \) and 0 < \( \eta_1 < \eta_2 < 1 \).

\( A_{red_k} < \eta_1 \), then:

Reduce the trust-region radius by setting \( \delta_k = \alpha_1 \|s_k\| \) and then calculate a new trail step using the new trust region radius.

Else if \( \eta_1 \leq \frac{A_{red_k}}{Pred_k} < \eta_2 \), then:

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

Set the trust-region radius: \( \delta_{k+1} = \max(\delta_k, \delta_{\text{min}}) \).

Else

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

Set the trust-region radius: \( \delta_{k+1} = \min(\delta_{\text{max}}, \max(\delta_{\text{min}}, \alpha_2 \delta_k)) \).

End if.

After accepting the step, we update the Hessian matrix and the penalty parameter \( \rho_k \). To update \( \rho_k \), we use the scheme presented in [18], which is described in step 6 of algorithm 2. The penalty parameter \( r_k \to r^* \) and the positive number \( \rho_k \to \rho^* \) as \( k \to \infty \).

Finally, the algorithm is terminated when either \( \|\nabla F_k\| + \|\nabla g_k D_k \tilde{g}_k\| + \|\tilde{h}_k\| \leq \varepsilon_2 \) or \( \|s_k\| \leq \varepsilon_1 \), for some \( \varepsilon_1, \varepsilon_2 > 0 \).

3.2.3. Active set trust-region algorithm
The following algorithm formally describes the proposed trust-region algorithm for solving problem (7).

Algorithm 3 (Active set trust-region algorithm):
Step 0 (Initialization)
Given \( x_0 \in \mathbb{R}^n \). Compute \( D_0 \). Set \( \rho_0 = 1, r_0 = 1, \sigma_0 = 1, \alpha, \beta, \delta \) such that \( \varepsilon_1 > 0, \varepsilon_2 > 0, 1 < \alpha_1 < 1 < \alpha_2 \), and \( 0 < \eta_1 < \eta_2 < 1 \).

Choose \( \delta_{\text{min}}, \delta_{\text{min}} \) and \( \delta_0 \) such that \( \delta_{\text{min}} \leq \delta_0 \leq \delta_{\text{max}} \). Set \( k = 0 \).

Step 1 (Test for convergence)
If \( \|\nabla F_k\| + \|\nabla g_k D_k \tilde{g}_k\| + \|\tilde{h}_k\| \leq \varepsilon_2 \), then stop.

End if.

Step 2 (Compute a trail step)
Compute the step by solving Equation (11).

If \( \|s_k\| \leq \varepsilon_1 \), then stop.

Else set \( x_{k+1} = x_k + s_k \).

End if.

Step 3 (Update the active set)
Compute \( D_{k+1} \).

Step 4 (Update \( r_k \))
If \( \|s_k\| \leq \frac{r_k}{4} (\|\tilde{h}_k\|^2 - \|\tilde{h}_k + \nabla h_k^T s_k\|^2) \), then

\[
4 \left[ \nabla F_k^T s_k + \frac{1}{2} s_k^T H_k s_k \right] + \frac{\rho_k}{2} \left[ \|D_k \tilde{g}_k\|^2 - \|D_k (\tilde{g}_k + \nabla g_k^T s_k)\|^2 \right]
\]

Set \( r_k = \frac{\sigma_k}{2} \). \( \sigma_k \).

Else Set \( r_k = \max(r_{k-1}, \rho_k^2) \).

End if.

Step 5 (Test the step and update the trust-region radius)
Test and update using algorithm 2.

Step 6 (Update \( \rho_k \))
If

\[
-\nabla F_k^T s_k - \frac{1}{2} s_k^T H_k s_k + \frac{\rho_k}{2} (\|D_k \tilde{g}_k\|^2 - \|D_k \tilde{g}_k + \nabla g_k^T s_k\|^2) \leq \sigma \|\nabla g_k D_k \tilde{g}_k\| + \|\nabla g_k D_k \tilde{g}_k\|, \delta_k),
\]

then set \( \rho_{k+1} = 2 \rho_k \) and \( \sigma_{k+1} = \frac{1}{2} \sigma_k \).

Else

Set \( \rho_{k+1} = \rho_k \) and \( \sigma_{k+1} = \sigma_k \).

End if.

Step 7 Set \( k = k + 1 \) and go to step 1.

A global convergence theorem of the trust-region algorithm (3) is presented in [17].
3.3. The main algorithm

The main steps of our approach for solving problem (1) are described in detail in algorithm 4.

\textbf{Algorithm 4 (Main algorithm)}:

\textbf{Step 1} (Transform the multi-objective problem into a single-objective problem)

Run algorithm 1.

\textbf{Step 2} (Construct the payoff table)

Run algorithm 3 to solve problem (2) for \( j = 1, 2 \).

\textbf{Step 3} (Solve the hybrid relaxed I-SHOT problem)

Run algorithm 3 to solve problem (6).

Continuously until the DM is satisfied from the Pareto-optimal set generated.

To ensure the effectiveness of the proposed algorithm, we apply it in solving the EELD problem.

4. EELD problem formulation

The EELD problem is to minimize two simultaneous non-commensurable and conflicting objective functions, fuel cost and emission, while satisfying several equality and inequality constraints [1]. This means that the minimization of emission is contrary to the maintenance of cost economy. Generally the deterministic EELD problem is formulated as follows.

4.1. Problem objectives

In the EELD problem, there are two objective functions to be minimized, as described in detail as follows.

4.1.1. Fuel cost objective function

The generator cost curves are usually represented by quadratic functions. The total fuel cost \( f_1 \) in ($/h) stated in [20] as follows:

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

where \( n \) is the number of generators; \( P_{Gi} \) is the real output power of the \( i \)th generator \( a_i; b_i \) and \( c_i \) are the cost coefficients of the \( i \)th generator.

4.1.2. Emission dispatch objective function

The total emission \( f_2 \) in (ton/h) of environmental pollutants such as nitrogen oxides NO\(_x\) and sulphur oxides SO\(_x\), due to burning fuels in fossil-fueled thermal units for production of power to meet the load demand can be expressed as [20]

\[ f_2 = \sum_{i=1}^{n} (10^{-2} (\alpha_i + \beta_i P_{Gi} + \gamma_i P_{Gi}^2) + \zeta_i e^{k_i P_{Gi}}), \]

where \( \alpha_i, \beta_i, \gamma_i, \zeta_i \) and \( \lambda_i \) are coefficients of the \( i \)th generator emission characteristics.

4.2. Problem constraints

The optimization problem is bounded by the following equality and inequality constraints.

4.2.1. Power balance constraint

The total power generated should cover the total load demand and the real power losses in the transmission lines. Hence,

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

where \( P_D \) is the total load demand and \( P_{\text{loss}} \) represents the transmission losses. The real power loss in transmission lines in Equation (16) is given in [21] as follows:

\[ P_{\text{loss}} = \sum_{i=1}^{n} \sum_{j=1}^{n} (A_{ij}(P_{Pi} + Q_i) + B_{ij}(Q_{Pi} - P_{Pi})), \]

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

4.2.2. Maximum and minimum limits of power generation

For stable operation, real power generated by each generator is restricted by minimum and maximum limits, i.e.

\[ P_{\text{Gmin}} \leq P_{Gi} \leq P_{\text{Gmax}}, Q_{\text{Gmin}} \leq Q_{Gi} \leq Q_{\text{Gmax}}, V_{\text{Gmin}} \leq V_{Gi} \leq V_{\text{Gmax}}. \]

4.2.3. Security constraints

A very large number of constraints are to be considered in the security-constrained EELD problem. But for typical systems, there is a small possibility of becoming overloaded for the large proportion of lines. So the EELD problem should consider only the critical lines which are in violation, or near violation of their respective security limits. We consider only the critical lines which are binding in the optimal solution. The detection of the critical lines is assumed to be done by the experiences of the DM. Minimizing the following objective function, we can obtain an improvement in the security

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

where \( m \) is the number of monitored lines, \( T_q(P_G) \) and \( T_{q,\text{max}} \) is the real power flow and the maximum limit of the real power flow of the \( q \)th line, respectively. By utilizing the generalized generation distribution factors (GGDF), the line flow of the \( q \)th line is formulated in terms of the
control variables $P_{Gi}$ in [22] as follows:

$$T_q(P_G) = \sum_{i=1}^{n} (D_{q1}P_{Gi}),$$

where $D_{q1}$ is the GGDF for the line $q$ due to generator $i$.

For secure operation, the transmission line loading $S_l$ is constrained by its upper limit $S_{l\text{max}}$ as

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

where $n_l$ is the number of transmission lines.

Aggregating the objectives and constraints, the EELD problem can be mathematically formulated as a non-linear equality and inequality constrained multi-objective optimization problem as follows:

minimize $f_1 = \sum_{i=1}^{n} (a_i + b_iP_{Gi} + c_iP_{Gi}^2),$

minimize $f_2 = \sum_{i=1}^{n} (10^{-2}(a_i + b_iP_{Gi} + c_iP_{Gi}^2) + \xi_{ie}P_{Gi}),$

subject to

$$\sum_{i=1}^{n} P_{Gi} - P_D - P_{\text{loss}} = 0,$$

$$P_{Gi\min} \leq P_{Gi} \leq P_{Gi\max},$$

$$Q_{Gi\min} \leq Q_{Gi} \leq Q_{Gi\max},$$

$$V_{Gi\min} \leq V_{Gi} \leq V_{Gi\max},$$

$$S_l \leq S_{l\text{max}},$$

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

The above problem can be formulated as follows:

minimize $f_j(x),$  
subject to $h(x) = 0,$  $g(x) \leq 0,$

where $j = 1, 2, h(x) = \sum_{i=1}^{n} P_{Gi} - P_D - P_{\text{loss}}$ and $g(x) = |P_{Gi\min} - P_{Gi}|, P_{Gi\max} - P_{Gi}, Q_{Gi\min} - Q_{Gi}, Q_{Gi\max} - Q_{Gi}, V_{Gi\min} - V_{Gi}, V_{Gi\max} - V_{Gi} S_l - S_{l\text{max}}|.$

The functions $f_j(x) : \mathbb{R}^n \to \mathbb{R}, h(x) : \mathbb{R}^n \to \mathbb{R}$, and $g(x) : \mathbb{R}^{3n+n_l} \to \mathbb{R}^{n+n+l}$ are twice continuously differentiable where $j = 1, 2.$

Algorithm (4) is applied in the next section for solving the EELD problem formulated as Equation (19).

5. Implementation of the proposed algorithm

We apply Algorithm 4 to the EELD problem and it is tested on the standard IEEE 30-bus system having 6 generating units. The values of fuel cost and emission coefficient are described in Table 1. The parameters of the proposed algorithm implementation are selected as follows: $\alpha_1 = 0.05, \alpha_2 = 2, \eta_1 = 1 \times 10^{-4}, \eta_2 = 0.5, \epsilon_1 = 1 \times 10^{-8}$ and $\epsilon_2 = 1 \times 10^{-8}.$

In the proposed algorithm, we begin from any starting point $x_0.$ The initial trust-region radius selected to be $\delta_0 = \max(\|s_{cp}\|, \delta_{\text{min}}),$ where $\delta_{\text{min}} = 1 \times 10^{-3}.$ The maximum trust-region radius is selected to be $\delta_{\text{max}} = 10^2 \delta_0.$

Successful termination with respect to the proposed active set trust-region algorithm indicates that the termination condition of the algorithm is met with $\epsilon_2.$ In contrast, 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 trail step is less than $\epsilon_1.$

6. Results and discussions

To investigate the effectiveness of the proposed algorithm, relaxed I-SHOT active set Trust-region algorithm (RIAT), we compare the obtained results with different previous algorithms that handled the same problem. These algorithms include the strength Pareto evolutionary algorithm (SPEA) [1], the non-dominated sorting genetic algorithm (NSGA) [1], the Niched Pareto genetic algorithm (NPGA) [1], the multi-objective particle swarm optimization (MOPSO) [2], the fuzzy clustering-based particle swarm optimization (FCPSO) [3], the trust-region-based augmented Lagrangian method (TRALM) [23] and the $\epsilon$-constraint algorithm (EC) [24].

Table 2 is presented as a comparison of the best fuel costs of the previously mentioned algorithms. This table shows that the fuel cost obtained by the RIAT algorithm is less than the ones of the other algorithms and the corresponding emission value of the best cost of our approach is less than the other algorithms except in EC. These results are visualized in Figure 1.

Table 3 is presented to compare the best emissions of the previous mentioned algorithms. This table shows that the emission amount obtained by the RIAT algorithm is less than the ones of the other algorithms except in EC approach. The best emission corresponding cost in our approach is less than all corresponding costs in other algorithms. The enhancement in the best emission of EC algorithm upon our approach is small when compared to the enhancement of the
Table 2. Comparison of best fuel costs of some algorithms to the considered EELD problem.

|       | SPEA [1] | NPGA [1] | NSGA [1] | MOPSO [2] | FCPSO [27] | EC [23] | TRALM [28] | RIAT [proposed] |
|-------|----------|----------|----------|-----------|------------|---------|------------|-----------------|
| $P_{G1}$ | 0.1319   | 0.1127   | 0.1358   | 0.1524    | 0.2183     | 0.1648  | 0.1799     |                 |
| $P_{G2}$ | 0.3654   | 0.3747   | 0.3151   | 0.3472    | 0.3535     | 0.3554  | 0.3456     |                 |
| $P_{G3}$ | 0.7791   | 0.8057   | 0.8418   | 0.7857    | 0.7974     | 0.5776  | 0.6619     |                 |
| $P_{G4}$ | 0.9282   | 0.9031   | 1.0431   | 1.0180    | 0.9719     | 0.7590  | 1.1079     |                 |
| $P_{G5}$ | 0.1308   | 0.1347   | 0.0631   | 0.0995    | 0.0862     | 0.5393  | 0.1691     |                 |
| $P_{G6}$ | 0.5292   | 0.5331   | 0.4664   | 0.4669    | 0.4961     | 0.4080  | 0.4147     | 0.3987          |
| Cost ($/h) | 619.60   | 620.46   | 620.87   | 618.54    | 620.18     | 611.22  | 613.93     | 610.0006        |
| Emission (ton/h) | 0.2244   | 0.2243   | 0.2368   | 0.2308    | 0.2283     | 0.2043  | 0.2322     | 0.21157         |

**Figure 1.** Best cost solutions of some algorithms to the considered EELD problem.

Table 3. Comparison of best emissions of some algorithms to the considered EELD problem.

|       | SPEA [1] | NPGA [1] | NSGA [1] | MOPSO [2] | FCPSO [27] | EC [23] | TRALM [28] | RIAT [proposed] |
|-------|----------|----------|----------|-----------|------------|---------|------------|-----------------|
| $P_{G1}$ | 0.4419   | 0.4753   | 0.4403   | 0.4589    | 0.4796     | 0.4122  | 0.4649     | 0.2805          |
| $P_{G2}$ | 0.4598   | 0.5162   | 0.4940   | 0.5121    | 0.5287     | 0.4667  | 0.5164     | 0.4278          |
| $P_{G3}$ | 0.6944   | 0.6513   | 0.7509   | 0.6524    | 0.6711     | 0.5514  | 0.6201     | 0.5238          |
| $P_{G4}$ | 0.4616   | 0.4363   | 0.5060   | 0.4331    | 0.5318     | 0.4059  | 0.4764     | 0.6902          |
| $P_{G5}$ | 0.1952   | 0.1896   | 0.1375   | 0.1981    | 0.1257     | 0.5731  | 0.2091     | 0.5323          |
| $P_{G6}$ | 0.6131   | 0.3988   | 0.3364   | 0.6129    | 0.5299     | 0.4550  | 0.5771     | 0.4378          |
| Cost ($/h) | 651.71   | 657.59   | 649.24   | 656.87    | 651.62     | 642.70  | 651.57     | 623.0762        |
| Emission (ton/h) | 0.2019   | 0.2017   | 0.2048   | 0.2014    | 0.2047     | 0.1944  | 0.2008     | 0.19896         |

The best compromise solution is calculated using the membership function given in [25] and [3]. This membership function is used to calculate each member of the Pareto-optimal set for each method. After this calculation, the best compromise solution is the one that has the maximum value of membership function.

Table 4 is presented to compare the existing best compromise solutions of the previously considered algorithms. This table shows a reasonable compromise solution for the proposed RIAT algorithm. These results are visualized in Figure 3.

To investigate the effectiveness of using relaxed I-SHOT approach, we compare our obtained results with the results obtained using the I-SHOT method and we use the same algorithm for solving the obtained single-objective problem described in Section 3.2.3 and we call this algorithm as IAT.

Table 5 is presented to compare the best fuel cost, the best emission and the best compromise solution of RIAT and IAT algorithms. As can be seen from this table, the best emission of IAT algorithm is better than the one obtained using RIAT. By using the relaxation technique, we make an increment in emission value to make a sufficient decrement in fuel cost value. So the best fuel cost and best compromise solution of RIAT algorithm are better than those of the IAT algorithm.

The Pareto front of RIAT and IAT algorithms is presented in Figure 4. The graph asserts the fact that the algorithm makes an increment in emission value to make a decrement in fuel cost value.
Figure 2. Best emission solutions of some algorithms to the considered EELD problem.

Table 4. Comparison of best compromise solutions of some algorithms to the considered EELD problem.

|        | SPEA [1] | NPGA [1] | NSGA [1] | MOPSO [2] | FCPSO [27] | EC [23] | TRALM [28] | RIAT (proposed) |
|--------|----------|----------|----------|-----------|-------------|---------|------------|----------------|
| $P_{G1}$ | 0.3052  | 0.2998  | 0.2712  | 0.2882  | Compromise solution was not considered in the paper | 0.3149 | 0.3126 | 0.2467 |
| $P_{G2}$ | 0.4389  | 0.4325  | 0.3670  | 0.3956  | 0.4076  | 0.4262 | 0.4044 | 0.5198 |
| $P_{G3}$ | 0.7163  | 0.7342  | 0.8099  | 0.7320  | 0.5613  | 0.6508 | 0.7822 | 0.5198 |
| $P_{G4}$ | 0.6978  | 0.6853  | 0.7550  | 0.7520  | 0.5795  | 0.7994 | 0.5198 | 0.4111 |
| $P_{G5}$ | 0.1552  | 0.1560  | 0.1357  | 0.1489  | 0.5544  | 0.1809 | 0.5198 | 0.4111 |
| Cost ($/h) | 629.59  | 630.06  | 625.71  | 626.10  | 623.58  | 622.78 | 617.9484 | 617.9484 |
| Emission (ton/h) | 0.2079  | 0.2079  | 0.2136  | 0.2106  | 0.1969  | 0.2094 | 0.2032 | 0.2032 |

Figure 3. Best compromise solutions of some algorithms to the considered EELD problem.
Table 5. Comparison of results of RIAT and IAT algorithms to the considered EELD problem.

|     | Best Cost | Best emission | Best Compromise solution |
|-----|-----------|--------------|--------------------------|
|     | IAT       | RIAT         | IAT                      | RIAT        | IAT       | RIAT        |
| PG1 | 0.2060    | 0.1799       | 0.3321                   | 0.2805      | 0.2559    | 0.2467      |
| PG2 | 0.4036    | 0.3552       | 0.4731                   | 0.4278      | 0.4344    | 0.4044      |
| PG3 | 0.5070    | 0.5148       | 0.5217                   | 0.5238      | 0.5188    | 0.5198      |
| PG4 | 0.8501    | 0.9073       | 0.5833                   | 0.6902      | 0.7249    | 0.7822      |
| PG5 | 0.5070    | 0.5148       | 0.5217                   | 0.5238      | 0.5188    | 0.5198      |
| PG6 | 0.1401    | 0.3987       | 0.4581                   | 0.4378      | 0.43106   | 0.4111      |
| Cost ($/h) |       | 615.3636     | 610.0006                 | 631.8569    | 623.0762  | 621.1265    |
| Emission (ton/h) | 0.207083 | 0.21157      | 0.195267                 | 0.19896     | 0.20044   | 0.20322     |

Figure 4. Pareto-optimal fronts of RIAT and IAT to the considered EELD problem.

The results obtained when implementing the proposed trust-region algorithm on the constrained multi-objective EELD optimization problem indicate that it is an efficient tool for this type of problems. The results also illustrate the global convergence of this modified method, which makes it more suitable for practical computations with lower computational complexity and easier implementation.

Also, the proposed multi-objective approach RIAT algorithm allows more control over the optimization process and reduces the feasible region at every iteration with the interaction of the DM preferences so that the most preferred solution (decision vector) can be found.

7. Conclusions

In this work, we introduce a hybrid method between two interactive multi-objective methods, I-SHOT method and STEM (which we called relaxed I-SHOT method). By using the Relaxed I-SHOT method, the multi-objective constrained optimization problem is transformed into a single objective constrained optimization problem. The active set strategy is used together with the penalty method to convert the single objective constrained optimization problem into an unconstrained optimization problem. Then we are concerned with the proposed trust-region algorithm to solve the unconstrained optimization problem. The previous process is collected in RIAT algorithm.

This algorithm is considered as a globally search method to form a Pareto-optimal set where multiple Pareto-optimal solutions can be found in one run. We need to select one operating point, which will satisfy the different goals to some extent. The Relaxed I-SHOT method makes the interaction between the DM and the obtained Pareto-optimal set introduce more preferred solutions to the DM, from which he(she) can select the best compromise solution.

The proposed approach gives the DM great control over the optimization process. The Relaxed I-SHOT method gives the DM the ability to classify the objectives, make an improvement to some objectives values by small relaxation (increment) in the other objective values. Generally, the Pareto-optimal solutions
obtained cannot be predicted. Though the DM can be sufficient for some objectives values and not sufficient with other objectives values, so the relaxation technique improves the unacceptable objective values. The proposed approach with two penalty parameters method reduces the number of computations needed when solving the obtained single-objective optimization problem by ensuring that the predicted reduction is positive before testing the step.

In this work, the proposed algorithm was applied to the EELD problem, which is formulated as a multi-objective optimization problem with two conflicting objectives, fuel cost and emission level. The results obtained by our algorithm is less than most of the other algorithms in the literature for the values of best fuel cost or the best emission level. The obtained results present the efficiency of the proposed RIAT algorithm in solving real problems. The approach is applied to the EELD problem with no limitation to the number of objectives. The proposed algorithm is efficient for solving ill-defined systems and non-convex multi-objective optimization problems.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

[1] Abido MA. Multi-objective evolutionary algorithms for electric power dispatch problem. IEEE Trans Evol Comput. 2006;10:315–329.
[2] Abido MA. Multi-objective particle swarm optimization for environmental/economic dispatch problem. Electr Power Syst Res. 2009;79:1105–1113.
[3] Agrawal S, Panigrahi BK, Tiwari MK. Multi-objective particle swarm algorithm with fuzzy clustering for electrical power dispatch. IEEE Trans Evol Comput. 2008;12:529–541.
[4] El-Ela A, Abido MA. Optimal operation strategy for reactive power control. ASME. 1992;41:19–40.
[5] Kermanshahi BS, Wu Y, Yasuda K, et al. Environmental marginal cost evaluation by non inferiority surface (power systems). IEEE Trans. Power Syst. 1996;5:1151–1159.
[6] Dennis J, El-Alem M, Williamson K. A trust-region approach to nonlinear systems of equalities and inequalities. SIAM J. Optimization. 1999;9:291–315.
[7] Fletcher R. An I penalty method for nonlinear constraints. Numer Optim. 1984;6:26–40.
[8] El-Sobky B, Abo-Elnagag Y. Multi-objective optimal load flow problem with interior-point trust-region strategy. Electr Power Syst Res. 2017;148:127–135.
[9] Yuan Y. Recent advances in trust region algorithms. Math Program Ser. 2015;1:249–281.
[10] Wood A, Woolenbg BF. Power generation, operation and control. New York: Wiley; 1996. p. 29–90.
[11] Azarm S, Narayanan S. A multi-objective interactive sequential hybrid optimization technique for decision making. Eng Optim. 2000;32:485–500.
[12] Miettinen K. Nonlinear multi-objective optimization. WWW.NIMBUS on the Internet Computer Optimization Research; 2000.
[13] Bertsekas D. Constrained optimization and Lagrange multiplier methods. Belmont, MA: Athena Scientific; 1996.
[14] Maciel MC, Sottosanto GN. An augmented penalization algorithm for the equality constrained minimization problem. Tendencias em Matematica Aplicada e Computacional. 2002;3:171–180.
[15] Toint PL. Towards an efficient sparsity exploiting Newton’s method for minimization. New York: Academic Press; 1981. p. 57–87.
[16] Steihaug T. The conjugate gradient method and trust region in large scale optimization. SIAM J Numer Anal. 1983;20:626–637.
[17] El-Alem M. A global convergence theory for a class of trust-region algorithms for constrained optimization [PhD thesis]. Houston (TX); Department of Mathematical Sciences, Rice University; 1988.
[18] Yuan Y. On the convergence of a new trust region algorithm. Numer Math. 1995;70:515–539.
[19] Abo-Elnagag Y, El-Sobkya B, Al-Naser L. An active-set trust-region algorithm for solving warehouse location problem. J Taibah Univ Sci. 2017;11:353–358.
[20] Yokoyama R, Bae SH, Morita T, et al. Multi-objective generation dispatch based on probability security criteria. IEEE Trans Power Syst. 1988;3:317–324.
[21] Hazarika D, Bordoloi PK. Modified loss coefficients in the determination of optimum generation scheduling. IEEE Proc. 2015;138:166–172.
[22] Ng WY. Generalized generation distribution factors for power system security evaluations. IEEE Trans. 1981;PAS-100:1001–1005.
[23] Bishe HM, Kian AR, Esfahani MS. Solving environmental/economic power dispatch problem by a trust region based augmented lagrangian method. Iran J Electr Electron Eng. 2012;8:177–187.
[24] Vahidinasab V, Jadid S. Joint economic and emission dispatch in energy markets: A multi-objective mathematical programming approach. Energy. 2010;35:1497–1504.
[25] Das DB, Patvardhan C. New multi-objective stochastic search technique for economic load dispatch. IET Proc Gener Transm Distrib. 1998;145:747–752.