Improved differential evolution algorithms for handling economic dispatch optimization with generator constraints

Leandro dos Santos Coelho a,*, Viviana Cocco Mariani b

a Production and Systems Engineering Graduate Program, Pontifical Catholic University of Parana, PUCPR/PPGEPS, Imaculada Conceicao, 1155, Zip code 80215-901, Curitiba, Parana, Brazil
b Mechanical Engineering Graduate Program, Pontifical Catholic University of Parana, PUCPR/PPGEM, Imaculada Conceicao, 1155, Zip code 80215-901, Curitiba, Parana, Brazil

Received 3 May 2006; accepted 12 November 2006
Available online 12 January 2007

Abstract

Global optimization based on evolutionary algorithms can be used as the important component for many engineering optimization problems. Evolutionary algorithms have yielded promising results for solving nonlinear, non-differentiable and multi-modal optimization problems in the power systems area. Differential evolution (DE) is a simple and efficient evolutionary algorithm for function optimization over continuous spaces. It has reportedly outperformed search heuristics when tested over both benchmark and real world problems. This paper proposes improved DE algorithms for solving economic load dispatch problems that take into account nonlinear generator features such as ramp rate limits and prohibited operating zones in the power system operation. The DE algorithms and its variants are validated for two test systems consisting of 6 and 15 thermal units. Various DE approaches outperforms other state of the art algorithms reported in the literature in solving load dispatch problems with generator constraints.

© 2006 Elsevier Ltd. All rights reserved.

Keywords: Differential evolution algorithm; Economic dispatch; Generator constraints; Optimization

1. Introduction

Electrical power systems and their operation are among the most complex problems of engineering due to their highly nonlinear and computationally difficult environments [1]. The objective of the economic dispatch problem (EDP) of electric power generation is to schedule the committed generating unit outputs so as to meet the required load demand at minimum operating cost while satisfying all unit and system equality and inequality constraints [2].

In traditional EDPs, the cost function of each generator is approximately represented by a simple quadratic function and is solved using mathematical programming methods, such as the lambda iterative method, dynamic programming, linear programming, nonlinear programming, Lagrangian relaxation and gradient techniques [3–9]. In reality, a generating unit cannot exhibit a convex fuel cost function, as there are various practical limitations in operation and control. However, none of these methods may be able to provide a global optimal solution, for they usually get stuck at a local optimum.

When compared with conventional techniques, modern heuristic optimization techniques such as simulated annealing [10], neural networks [11–13], particle swarm optimization [14–16] and taboo search [17,18] have been given much attention by many researchers due to their ability to find an almost global optimal solution for EDPs with operating constraints.

* Corresponding author. Fax: +55 413271 1345.
E-mail address: leandro.coelho@pucpr.br (Leandro dos Santos Coelho).
In this context, recently, evolutionary algorithms (EAs) have emerged as powerful methods to achieve the goals in EDPs [1,19–24]. EAs are general purpose methods for optimization, belonging to a class of meta-heuristics inspired by the evolution of living beings and genetics [25,26]. EAs do not usually require in depth mathematical knowledge of the problem and do not guarantee the optimal solution in a finite time. However, they are useful for large scale optimization problems, dealing efficiently with huge irregular search spaces. EAs use a population of structures (individuals) in which each one is a candidate solution for the optimization problem. Since they are population based methods, they make a parallel search of the space of possible solutions and are less susceptible to local minima. Therefore, EAs are suitable for solving a broad range of complex problems characterized by discontinuity, nonlinearity and multi-variability. The usefulness of a given solution is obtained from the environment by means of a fitness function. The population of solutions evolves through generations based on probabilistic transitions using cooperation and auto-adaptation of individuals. There are many variants of EAs, but the main differences rely on how individuals are represented, the genetic operators that modify individuals (especially mutation and crossover) and the selection procedure. Most current approaches of EAs descend from the principles of the main methodologies: genetic algorithms, evolutionary programming, evolution strategy, and differential evolution (DE).

Storn and Price [27] first introduced the DE algorithm a few years ago. DE was successfully applied to the optimization of some well-known nonlinear, non-differentiable and non-convex functions in Storn [28]. DE is a population based and direct stochastic search algorithm (minimizer or maximizer) whose simple, yet powerful and straightforward, features make it very attractive for numerical optimization. DE uses a rather greedy and less stochastic approach to problem solving compared to EAs. DE combines simple arithmetic operators with the classical operators of crossover, mutation and selection to evolve from a randomly generated starting population to a final solution.

In this paper, two economic dispatch problems with 6 and 15 thermal units with ramp rate limits and prohibited operating zones in the power system operation and transmission loss are employed to demonstrate the performance of the proposed improved DE approaches. The results obtained through the improved DE approaches are analyzed and compared with those reported in the recent literature. The proposed improvements in the design of the control parameters $F$ (mutation factor) and CR (crossover rate) in DE is a powerful strategy to diversify the DE population and improve the DE’s performance in preventing premature convergence to local minima.

The remainder of the paper is organized as follows: Section 2 describes the formulation of an economic dispatch problem, while Section 3 explains the standard DE and the new improved DE approaches. Section 4 then details the procedure of constraint handling in DE and Section 5 presents the results of the optimization and compares methods to solve the case studies of economic dispatch problems with 6 and 15 thermal units. Lastly, Section 6 outlines our conclusions and future research.

2. Formulation of economic dispatch problem with generator constraints

The objective of the economic dispatch problem is to minimize the total fuel cost at thermal power plants subjected to the operating constraints of a power system with a defined interval (typically 1 h). Therefore, it can be formulated mathematically as the minimization of an objective function with constraints. The equality and inequality constraints are represented by Eqs. (1) and (2) given by

$$\sum_{i=1}^{n} P_i - P_L - P_D = 0$$

(1)

$$p_{\text{min}}^i \leq P_i \leq p_{\text{max}}^i$$

(2)

In the power balance criterion, an equality constraint must be satisfied, as shown in Eq. (1). The generated power should be the same as the total load demand plus total line losses. The generating power of each generator (capacity limits constraint) should lie between the maximum and minimum limits represented by Eq. (2), where $P_i$ is the power of generator $i$ (in MW); $n$ is the number of generators in the system; $P_D$ is the system’s total demand (in MW); $P_L$ represents the total line losses (in MW) and $p_{\text{min}}^i$ and $p_{\text{max}}^i$ are, respectively, the outputs of the minimum and maximum operation of the generating unit $i$ (in MW). The total fuel cost function is formulated as follows:

$$\text{min } f = \sum_{i=1}^{n} F_i(P_i)$$

(3)

where $F_i$ is the total fuel cost for the $i$th generator (in $/h), which is defined by the following equation:

$$F_i(P_i) = a_iP_i^2 + b_iP_i + c_i$$

(4)

where $a_i$, $b_i$ and $c_i$ are the cost coefficients of generator $i$.

In this study, the ramp rate limits, prohibited operating zone constraints and transmission losses are considered [16,20,22,29]. The constraints of the EDP at specific operating intervals can be represented by Eqs. (5)–(8) given by

(i) ramp rate limit constraints:

$$\text{max}(P_{\text{min}}^i, P_L^i - DR_i) \leq P_i' \leq \text{min}(P_{\text{max}}^i, P_L^i + UR_i)$$

(5)

where $P_i'$ is the present output power and $P_L^i$ is the previous output power. UR$_i$ is the up-ramp limit of the $i$th generator (in units of MW/time period) and DR$_i$ is the down-ramp limit of the $i$th generator (in units of MW/time period).

(ii) prohibited operating zones constraints:
work losses are expressed as a quadratic function of the variables adopted for this work. In the method of Storn and Price in 1995 [27] and was successfully applied in the optimization of some well-known nonlinear, non-differentiable and non-convex functions by Storn[28].

The fundamental idea behind DE is a scheme by which it generates the trial parameter vectors. In each time step, DE mutates vectors by adding weighted random vector differentials to them. If the cost of the trial vector is better than that of the target, the target vector is replaced by the trial vector in the next generation.

A number of alternative versions of basic configurations of DE have been proposed by Refs. [28,34,35]. The different variants are classified according to the following notation: DE/\alpha/\beta/\delta, where \alpha indicates the method for selecting the parent chromosome that will form the base of the mutated vector, \beta indicates the number of difference vectors used to perturb the base chromosome and \delta indicates the recombination mechanism used to create the offspring population. The bin acronym indicates that the recombination is controlled by a series of independent binomial experiments.

The variant implemented here was the DE/rand/1/bin, which is given by the following steps:

3.1. Initialization of the parameter setup

The user must choose the key parameters that control the DE, i.e. population size, boundary constraints of optimization variables (M), mutation factor (f_m), crossover rate (CR), and the stopping criterion (t_{max}).

3.2. Initialization of population

Set generation \( t = 0 \). Initialize a population of \( i = 1, \ldots , M \) individuals (real valued \( n \)-dimensional solution vectors) with random values generated according to a uniform probability distribution in the \( n \)-dimensional problem space. Initialize the entire solution vector population within the given upper and lower limits of the search space. In this work, the power of generators \( P \) are represented by individuals \( x \) in DE.

3.3. Evaluation of population

Evaluate the fitness value of each individual (in this work, the goal is to minimize the cost function).

3.4. Mutation operation (or differential operation)

Mutation is an operation that adds a vector differential to a population vector of individuals, according to the following equation:

\[
z_i(t + 1) = x_{i,j}(t) + f_m(t)[x_{i,j}(t) - x_{i,j}(t)]
\]  

where \( j = 1, 2, \ldots , n \) is the position in \( n \) of the dimensional individual; \( t \) is the time (generation); \( x_i(t) = [x_{i,1}(t), x_{i,2}(t), \ldots , x_{i,n}(t)]^\top \) stands for the position of the \( i \)th individual of a population of real valued \( n \)-dimensional vectors; \( z_i(t) = [z_{i,1}(t), z_{i,2}(t), \ldots , z_{i,n}(t)]^\top \) stands for the position of the \( i \)th individual of a mutant vector and \( r_1, r_2 \text{ and } r_3 \) are mutually different integers that are also different from the running index, \( i \), randomly selected with uniform distribution from the set \( \{1, 2, \ldots , i - 1, i + 1, \ldots , N\} \). The mutation factor \( f_m(t) > 0 \) is a real parameter, which controls the amplification of the difference between two individuals with indexes \( r_2 \text{ and } r_3 \) so as to avoid search stagnation and
is usually a constant value taken from the range [0.4, 1] (see comments in Refs. [27,28]).

The mutation operation using the difference between two randomly selected individuals may cause the mutant individual to escape from the search domain. If an optimized variable for the mutant individual is outside of the domain search, then this variable is replaced by its lower bound or its upper bound so that each individual can be restricted to remain within the search domain [36].

3.5. Recombination operation

Following the mutation operation, recombination is applied to the population. Recombination is employed to generate a trial vector by replacing certain parameters of the target vector by the corresponding parameters of a randomly generated donor vector.

For each vector, $\mathbf{z}_i(t+1)$, an index $\text{rnbr}(i) \in \{1, 2, \ldots, n\}$ is randomly chosen using a uniform distribution, and a trial vector, $u_i(t+1) = [u_{i1}(t+1), u_{i2}(t+1), \ldots, u_{in}(t+1)]^T$, is generated with

$$u_{ij}(t+1) = \begin{cases} z_{ij}(t+1), & \text{if } (\text{randb}(j) \leq \text{CR}) \text{ or } (j = \text{rnbr}(i)), \\ x_{ij}(t), & \text{if } (\text{randb}(j) > \text{CR}) \text{ or } (j \neq \text{rnbr}(i)) \end{cases}$$

where $\text{randb}(j)$ is the $j$th evaluation of a uniform random number generation with [0, 1] and CR is the crossover or recombination rate in the range [0, 1]. Usually, the performance of a DE algorithm depends on three variables: the population size, the mutation factor $f_m(t)$ and the CR.

3.6. Selection operation

Selection is the procedure whereby better offspring are produced. To decide whether or not the vector $u_i(t+1)$ should be a member of the population comprising the next generation, it is compared with the corresponding vector $x_i(t)$. Thus, if $f$ denotes the cost function under minimization, then

$$x_{i}(t+1) = \begin{cases} u_i(t+1), & \text{if } f(u_i(t+1)) < f(x_i(t)), \\ x_i(t), & \text{otherwise} \end{cases}$$

In this case, the cost of each trial vector $u_i(t+1)$ is compared with that of its parent target vector $x_i(t)$. If the cost, $f_i$, of the target vector $x_i(t)$ is lower than that of the trial vector, the target is allowed to advance to the next generation. Otherwise, the target vector is replaced by a trial vector in the next generation [26].

3.7. Verification of the stopping criterion

Set the generation number for $t = t + 1$. Proceed to Step 3.3 until a stopping criterion is met, usually a maximum number of iterations (generations), $t_{\text{max}}$. The stopping criterion depends on the type of problem.

3.8. Improved differential evolution algorithms

The parameters $M$, CR and $f_m$ of the DE are generally the key factors affecting the DE’s convergence. The utilization of improvements in the DE can be useful to escape more easily from local minima than with the traditional DE. In this context, the three new improved DE algorithms (IDE) based on sinusoidal and Gaussian functions for the $f_m$ setup are described as follows:

**Approach 1 – IDE(1).** The parameter $f_m$ of Eq. (9) is modified by formula (12) through the following equation based on a sinusoidal function given by

$$z_i(t+1) = x_{i,r_1}(t) + [x + A \sin(\omega \beta)] [x_{i,r_2}(t) - x_{i,r_3}(t)]$$

where $x + A \sin(\omega \beta)$ represents $f_m$; $z$ is a DC component of the signal, $A$ is the amplitude of the signal, $\omega$ is the angular frequency of the signal and $\beta$ is a gain. The choice of values was $\alpha = 0.4$, $A = 0.2$, $\omega = 0.2 \pi \max$ and $\beta$ are values linearly spaced between the initial value $-180^\circ$ and the final value of $180^\circ$ with increments based on $t_{\text{max}}$.

**Approach 2 – IDE(2).** The parameter $f_m$ of Eq. (9) is modified by formula (13) through the following equation based on random numbers with Gaussian distribution given by

$$z_i(t+1) = x_{i,r_1}(t) + [x + 0.1 \text{Gauss}] [x_{i,r_2}(t) - x_{i,r_3}(t)]$$

where $x + 0.1 \text{Gauss}$ represents $f_m$ of Eq. (9); $z$ is a DC component of signal and Gauss is a Gaussian distributed zero mean random number with unit variance.

**Approach 3 – IDE(3).** This approach is a combination of the IDE(1) and IDE(2) approaches. The parameter $f_m$ of Eq. (9) is modified by formula (14) through the following equation given by

$$z_i(t+1) = x_{i,r_1}(t) + \{x + [A \sin(\omega \beta)] [0.1 \text{Gauss}]\} \times [x_{i,r_2}(t) - x_{i,r_3}(t)]$$

4. Constraints handling with differential algorithms

A key factor in the application of DE algorithms to optimization of an EDP is how the algorithm handles the constraints relating to the problem. Over the last few decades, several methods have been proposed to handle constraints in EAs [37], such as DE algorithms. These methods can be grouped into four categories: methods that preserve the feasibility of solutions, penalty based methods, methods that clearly distinguish between feasible and unfeasible solutions and hybrid methods.

When DE algorithms are used for constrained optimization problems, it is common to handle constraints using concepts of penalty functions (which penalize unfeasible solutions), i.e. one attempts to solve an unconstrained problem in the search space $\mathcal{S}$ using a modified cost function $f$ such as
\[
\min f = \begin{cases} 
  f(x_i), & \text{if } x_i \in F \\
  f(x_i) + \text{penalty}(x_i), & \text{otherwise}
\end{cases} 
\] (15)

where \( \text{penalty}(x_i) \) is zero and the constraints are not violated; otherwise it is positive. The penalty function is usually based on a distance measured to the nearest solution in the feasible region \( F \) or to the effort to repair the solution.

The methodology proposed for constraint handling is divided into two steps. The first step involves finding solutions for the decision variables that lie within user defined upper (\( \text{lim}_{\text{upper}} \)) and lower (\( \text{lim}_{\text{lower}} \)) bounds, that is, \( x \in [\text{lim}_{\text{lower}}, \text{lim}_{\text{upper}}] \). Whenever a lower bound or an upper bound restriction fails to be satisfied, a repair rule applied according to Eqs. (16) and (17), respectively [38]:

\[
x_i(t+1) = x_i(t) + w \cdot \text{rand}\{\text{lim}_{\text{upper}}(x_i) - \text{lim}_{\text{lower}}(x_i)\} 
\] (16)

\[
x_i(t+1) = x_i(t) + w \cdot \text{rand}\{\text{lim}_{\text{upper}}(x_i) - \text{lim}_{\text{lower}}(x_i)\} 
\] (17)

where \( w \in [0, 1] \) is a user defined parameter (\( w \) is set to 0.01 in this work) and \( \text{rand}[0, 1] \) is an uniformly distributed random value between 0 and 1.

In the second step, if the inequalities of Eq. (1) and the constraints of Eqs. (5)–(8) are not satisfied, Eq. (3) is rewritten as

\[
\min f = \sum_{j=1}^{n} F_i(x_i) + q_1 \left( \sum_{j=1}^{n} x_i - P_L - P_D \right)^2 + q_2 \sum_{j=1}^{n} V_{k,j} 
\] (18)

or

\[
\min f = \sum_{i=1}^{n} F_i(P_i) + q_1 \left( \sum_{i=1}^{n} P_i - P_L - P_D \right)^2 + q_2 \sum_{j=1}^{n} V_j 
\] (19)

where \( q_1 \) and \( q_2 \) are positive constants (penalty factors) associated with the power balance and prohibited zones constraints, respectively. These penalty factors were tuned empirically, and their values are \( q_1 = 5 \) and \( q_2 = 1 \) in the studied cases. The \( V_j \) is expressed as follows:

\[
V_j = \begin{cases} 
  1, & \text{if } x_j \text{ violates the prohibited zones} \\
  0, & \text{otherwise}
\end{cases} 
\] (20)

5. Simulation results

In this paper, to assess the efficiency of the proposed differential evolution algorithms, two case studies (6 and 15 thermal units or generators) of economic dispatch problems were applied in which the cost functions constrained by ramp rate limits, prohibited operating zones in the power system operation and transmission losses are employed to demonstrate the efficiency.

Each optimization method (DE and IDE) was implemented in Matlab (MathWorks). All the programs were run on a 3.2 GHz Pentium IV processor with 2 GB of RAM.

In each case study, 50 independent runs were made for each of the optimization methods involving 50 different initial trial solutions for each optimization method. The setup of classical DE approaches used was the following:

- DE(1): classical DE using a constant mutation factor given by \( f_m = 0.4 \) and a crossover rate of \( CR = 0.8 \);
- DE(2): DE using a linear reduction of \( f_m \) with initial and final values of 0.8 and 0.3, respectively;
- DE(3): DE using a linear increase of \( f_m \) with initial and final values of 0.3 and 0.8, respectively.

In these case studies, the population size \( N \) was 50 and the stopping criterion \( t_{\text{max}} \) was 100 generations for the DE and IDE algorithms. All the \( B \) coefficients are given in per unit (p.u.) on a 100 MVA base capacity.

5.1. Case study I: six unit system

This case study consisted of six thermal units, 26 buses, and 46 transmission lines [39]. All thermal units are within the ramp rate limits and prohibited operating zones. The data shown in Tables 1 and 2 are also available in Refs. [16,22,40]. In this case, the load demand expected to be satisfied was \( P_D = 1263 \) MW. The \( B \) matrix of the transmission loss coefficient is given by

\[
B_{ij} = 10^{-3}, 
\]

\[
\begin{bmatrix}
1.7 & 1.2 & 0.7 & -0.1 & -0.5 & -0.2 \\
1.2 & 1.4 & 0.9 & 0.1 & -0.6 & -0.1 \\
0.7 & 0.9 & 3.1 & 0.0 & -1.0 & -0.6 \\
-0.1 & 0.1 & 0.0 & 0.24 & -0.6 & -0.8 \\
-0.5 & -0.6 & -0.1 & -0.6 & 12.9 & -0.2 \\
0.2 & -0.1 & -0.6 & -0.8 & -0.2 & 15.0 
\end{bmatrix}
\] (21)

\[
B_{ii} = 10^{-3}, 
\]

\[
\begin{bmatrix}
-0.3908 & -0.1297 & 0.7047 & 0.0591 & 0.2161 & -0.6635 \\
0.2161 & -0.6635 & -0.3908 & -0.1297 & 0.7047 & 0.0591 \\
0.0591 & 0.2161 & -0.6635 & -0.3908 & -0.1297 & 0.7047 \\
-0.6635 & 0.0591 & 0.2161 & -0.6635 & -0.3908 & -0.1297 \\
-0.3908 & -0.1297 & 0.7047 & 0.0591 & 0.2161 & -0.6635 \\
0.2161 & -0.6635 & -0.3908 & -0.1297 & 0.7047 & 0.0591 
\end{bmatrix}
\] (22)

\[
B_{ii} = 0.0056
\] (23)

The results obtained for case study I are given in Table 3, which shows that IDE(1) succeeded in finding the best solution of the tested methods. The low standard deviations indicated a good convergence of the DE and IDE methods in the 50 runs. From Table 3, the mean of fitness found by DE(2), DE(3), and IDE(1) for the 50 runs performed was better than the results of the other tested methods.

Table 4 shows the performance comparison among the proposed algorithms, an evolution strategy (ES) [22], a particle swarm optimization (PSO) approach [16] and the
The genetic algorithm (GA) method [16]. The best results obtained for solution vector with IDE(1) with minimum cost of 15,356.1809 $/h is given in Table 4. The simulation results of the DE and IDE methods outperformed the ES, PSO and GA methods presented in the literature in terms of solution quality and mean cost of convergence.

5.2. Case study II: fifteen unit system

This case study consisted of 15 thermal units. All thermal units are within the ramp rate limits and prohibited operating zones. The data shown in Tables 5 and 6 are also available in Refs. [16,22,29,40]. In this case, the load demand expected to be satisfied was $P_D = 2630$ MW. The $B$ matrix of the transmission loss coefficient is given by

$$\begin{align*}
B_{ij} &= 10^{-3} \cdot \\
&= \begin{bmatrix}
1.4 & 1.2 & 0.7 & -0.1 & -0.3 & -0.1 & -0.1 & -0.1 & -0.3 & -0.5 & -0.3 & -0.2 & 0.4 & 0.3 & -0.1 \\
1.2 & 1.5 & 1.3 & 0.0 & -0.5 & -0.2 & 0.0 & 0.1 & -0.2 & -0.4 & -0.4 & 0.0 & 0.4 & 1.0 & -0.2 \\
0.7 & 1.3 & 7.6 & -0.1 & -1.3 & -0.9 & -0.1 & 0.0 & -0.8 & -1.2 & -1.7 & 0.0 & -2.6 & 11.1 & -2.8 \\
-0.1 & 0.0 & -0.1 & 3.4 & -0.7 & -0.4 & 1.1 & 5.0 & 2.9 & 3.2 & -1.1 & 0.0 & 0.1 & 0.1 & -2.6 \\
-0.3 & -0.5 & -1.3 & -0.7 & 9.0 & 1.4 & -0.3 & -1.2 & -1.0 & -1.3 & 0.7 & -0.2 & -0.2 & -2.4 & -0.3 \\
-0.1 & -0.2 & -0.9 & -0.4 & 1.4 & 1.6 & 0.0 & -0.6 & -0.5 & -0.8 & 1.1 & -0.1 & -0.2 & -1.7 & 0.3 \\
-0.1 & 0.0 & -0.1 & 1.1 & -0.3 & 0.0 & 1.5 & 1.7 & 1.5 & 0.9 & -0.5 & 0.7 & 0.0 & -0.2 & -0.8 \\
-0.1 & 0.1 & 0.0 & 5.0 & -1.2 & -0.6 & 1.7 & 16.8 & 8.2 & 7.9 & -2.3 & -3.6 & 0.1 & 0.5 & -7.8 \\
-0.3 & -0.2 & -0.8 & 2.9 & -1.0 & -0.5 & 1.5 & 8.2 & 12.9 & 11.6 & -2.1 & -2.5 & 0.7 & -1.2 & -7.2 \\
-0.5 & -0.4 & -1.2 & 3.2 & -1.3 & -0.8 & 0.9 & 7.9 & 11.6 & 20.0 & -2.7 & -3.4 & 0.9 & -1.1 & -8.8 \\
-0.3 & -0.4 & -1.7 & -1.1 & 0.7 & 1.1 & -0.5 & -2.3 & -2.1 & -2.7 & 14.0 & 0.1 & 0.4 & -3.8 & 16.8 \\
-0.2 & 0.0 & 0.0 & 0.0 & -0.2 & -0.1 & 0.7 & -3.6 & -2.5 & -3.4 & 0.1 & 5.4 & -0.1 & -0.4 & 2.8 \\
0.4 & 0.4 & -2.6 & 0.1 & -0.2 & -0.2 & 0.0 & 0.1 & 0.7 & 0.9 & 0.4 & -0.1 & 10.3 & -10.1 & 2.8 \\
0.3 & 1.0 & 11.1 & 0.1 & -2.4 & -1.7 & -0.2 & 0.5 & -1.2 & -1.1 & -3.8 & -0.4 & -10.1 & 57.8 & -9.4 \\
-0.1 & -0.2 & -2.8 & -2.6 & -0.3 & 0.3 & -0.8 & -7.8 & -7.2 & -8.8 & 16.8 & 2.8 & 2.8 & -9.4 & 128.3
\end{bmatrix}
\end{align*}$$

$$B_{ii} = 10^{-3} \cdot \begin{bmatrix}
-0.1 & -0.2 & 2.8 & -0.1 & 0.1 & -0.3 & -0.2 & -0.20.6 & 3.9 & -1.7 & 0.0 & -3.2 & 6.7 & -6.4
\end{bmatrix}$$

$$B_{0i} = 0.0055$$
6. Conclusion and future research

This paper proposes improved DE algorithms to solve economic dispatch problems of electric energy that takes into account nonlinear generator features such as ramp rate limits and prohibited operating zones.

The DE algorithms and its variants were validated for two test systems consisting of 6 and 15 thermal units. DE algorithms have been found before to be a very successful and robust search paradigm for optimization problems.

DE algorithms offer potential advantages: they find the true global minimum regardless of the initial parameter values, and they display fast convergence and use few control parameters.

In relation to the procedure involved in solving the economic dispatch problem of electric energy, the results achieved with the DE and IDE were better than the results presented in Refs. [16,22]. The results of these simulations with improved DE approaches are very encouraging and

---

### Table 3
Convergence results (50 runs) of a case study I considering a six unit system with $P_D = 1263$ MW

<table>
<thead>
<tr>
<th>Method</th>
<th>Minimum cost ($/h)</th>
<th>Mean cost ($/h)</th>
<th>Standard deviation of cost ($/h)</th>
<th>Maximum cost ($/h)</th>
<th>Mean CPU time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DE(1)</td>
<td>15,355</td>
<td>15,357</td>
<td>$9.772 \cdot 10^{-5}$</td>
<td>15,360</td>
<td>0.45</td>
</tr>
<tr>
<td>DE(2)</td>
<td>15,353</td>
<td>15,356</td>
<td>$1.960 \cdot 10^{-4}$</td>
<td>15,361</td>
<td>0.46</td>
</tr>
<tr>
<td>DE(3)</td>
<td>15,353</td>
<td>15,356</td>
<td>$2.102 \cdot 10^{-4}$</td>
<td>15,362</td>
<td>0.46</td>
</tr>
<tr>
<td>IDE(1)</td>
<td>15,351</td>
<td>15,356</td>
<td>$1.291 \cdot 10^{-4}$</td>
<td>15,359</td>
<td>0.46</td>
</tr>
<tr>
<td>IDE(2)</td>
<td>15,352</td>
<td>15,357</td>
<td>$1.714 \cdot 10^{-4}$</td>
<td>15,362</td>
<td>0.46</td>
</tr>
<tr>
<td>IDE(3)</td>
<td>15,353</td>
<td>15,357</td>
<td>$1.655 \cdot 10^{-4}$</td>
<td>15,361</td>
<td>0.46</td>
</tr>
<tr>
<td>Evolution strategy [22]</td>
<td>15,410</td>
<td>15,430</td>
<td>–</td>
<td>15,470</td>
<td>0.36</td>
</tr>
<tr>
<td>Genetic algorithm [16]</td>
<td>15,459</td>
<td>15,469</td>
<td>–</td>
<td>15,469</td>
<td>41.58</td>
</tr>
</tbody>
</table>

* values obtained from references.

### Table 4
Comparison of five methods: best result for the case study I

<table>
<thead>
<tr>
<th>Unit power output (MW)</th>
<th>DE(2) proposed</th>
<th>IDE(1) proposed</th>
<th>Evolution strategy [22]</th>
<th>Particle swarm optimization</th>
<th>Genetic algorithm [16]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_1$</td>
<td>446,409,821</td>
<td>446,448,481</td>
<td>451.56</td>
<td>447.95</td>
<td>474.81</td>
</tr>
<tr>
<td>$P_2$</td>
<td>173,104,909</td>
<td>172,851,548</td>
<td>173.44</td>
<td>173.32</td>
<td>178.64</td>
</tr>
<tr>
<td>$P_3$</td>
<td>261,591,793</td>
<td>261,806,179</td>
<td>263.99</td>
<td>263.47</td>
<td>262.21</td>
</tr>
<tr>
<td>$P_4$</td>
<td>139,339,380</td>
<td>138,332,872</td>
<td>147.46</td>
<td>139.06</td>
<td>134.28</td>
</tr>
<tr>
<td>$P_5$</td>
<td>161,367,730</td>
<td>162,858,077</td>
<td>164.68</td>
<td>165.47</td>
<td>151.90</td>
</tr>
<tr>
<td>$P_6$</td>
<td>87,054,195</td>
<td>86,576,285</td>
<td>71.32</td>
<td>87.13</td>
<td>74.18</td>
</tr>
<tr>
<td>Total power (MW)</td>
<td>1268,922</td>
<td>1268,875</td>
<td>1272.46</td>
<td>1275.96</td>
<td>1276.03</td>
</tr>
<tr>
<td>$P_L$ (MW)</td>
<td>5.922</td>
<td>5.875</td>
<td>9.46</td>
<td>12.96</td>
<td>13.02</td>
</tr>
<tr>
<td>Total cost ($/h)</td>
<td>15,357.337</td>
<td>15,356.181</td>
<td>15,407.527</td>
<td>15,450.00</td>
<td>15,459.00</td>
</tr>
</tbody>
</table>

### Table 5
Data for the 15 thermal units of generating unit capacity and coefficients

<table>
<thead>
<tr>
<th>Unit</th>
<th>$P_{\min}$ (MW)</th>
<th>$P_{\max}$ (MW)</th>
<th>$a$ ($$/MW^2$$)</th>
<th>$b$ ($$/MW$$)</th>
<th>$c$ ($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>150</td>
<td>455</td>
<td>0.000299</td>
<td>10.1</td>
<td>671</td>
</tr>
<tr>
<td>2</td>
<td>150</td>
<td>455</td>
<td>0.000183</td>
<td>10.2</td>
<td>574</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>130</td>
<td>0.001126</td>
<td>8.8</td>
<td>374</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>130</td>
<td>0.001126</td>
<td>8.8</td>
<td>374</td>
</tr>
<tr>
<td>5</td>
<td>150</td>
<td>470</td>
<td>0.000205</td>
<td>10.4</td>
<td>461</td>
</tr>
<tr>
<td>6</td>
<td>135</td>
<td>460</td>
<td>0.000301</td>
<td>10.1</td>
<td>630</td>
</tr>
<tr>
<td>7</td>
<td>135</td>
<td>465</td>
<td>0.000364</td>
<td>9.8</td>
<td>548</td>
</tr>
<tr>
<td>8</td>
<td>60</td>
<td>300</td>
<td>0.000338</td>
<td>11.2</td>
<td>227</td>
</tr>
<tr>
<td>9</td>
<td>25</td>
<td>162</td>
<td>0.000087</td>
<td>11.2</td>
<td>173</td>
</tr>
<tr>
<td>10</td>
<td>25</td>
<td>160</td>
<td>0.001203</td>
<td>10.7</td>
<td>175</td>
</tr>
<tr>
<td>11</td>
<td>20</td>
<td>80</td>
<td>0.003586</td>
<td>10.2</td>
<td>186</td>
</tr>
<tr>
<td>12</td>
<td>20</td>
<td>80</td>
<td>0.005513</td>
<td>9.9</td>
<td>230</td>
</tr>
<tr>
<td>13</td>
<td>25</td>
<td>85</td>
<td>0.003571</td>
<td>13.1</td>
<td>225</td>
</tr>
<tr>
<td>14</td>
<td>15</td>
<td>55</td>
<td>0.001929</td>
<td>12.1</td>
<td>309</td>
</tr>
<tr>
<td>15</td>
<td>15</td>
<td>55</td>
<td>0.004447</td>
<td>12.4</td>
<td>323</td>
</tr>
</tbody>
</table>

### Table 6
Data for the 15 thermal units of ramp rate limits and prohibited zones of the generating units

<table>
<thead>
<tr>
<th>Unit</th>
<th>$P^0_i$ (MW)</th>
<th>$UR_i$ (MW/h)</th>
<th>$DR_i$ (MW/h)</th>
<th>Prohibited zones</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zone 1</td>
<td>Zone 2</td>
<td>Zone 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>400</td>
<td>80</td>
<td>120</td>
<td>[185–255]</td>
</tr>
<tr>
<td>2</td>
<td>300</td>
<td>80</td>
<td>120</td>
<td>[305–335]</td>
</tr>
<tr>
<td>3</td>
<td>105</td>
<td>130</td>
<td>130</td>
<td>[180–200]</td>
</tr>
<tr>
<td>4</td>
<td>100</td>
<td>130</td>
<td>130</td>
<td>[230–255]</td>
</tr>
<tr>
<td>5</td>
<td>90</td>
<td>80</td>
<td>120</td>
<td>[180–200]</td>
</tr>
<tr>
<td>6</td>
<td>400</td>
<td>80</td>
<td>120</td>
<td>[230–255]</td>
</tr>
<tr>
<td>7</td>
<td>350</td>
<td>80</td>
<td>120</td>
<td>[230–255]</td>
</tr>
<tr>
<td>8</td>
<td>95</td>
<td>65</td>
<td>100</td>
<td>[30–40]</td>
</tr>
<tr>
<td>9</td>
<td>105</td>
<td>60</td>
<td>100</td>
<td>[30–40]</td>
</tr>
<tr>
<td>10</td>
<td>110</td>
<td>60</td>
<td>100</td>
<td>[30–40]</td>
</tr>
<tr>
<td>11</td>
<td>60</td>
<td>80</td>
<td>80</td>
<td>[30–40]</td>
</tr>
<tr>
<td>12</td>
<td>40</td>
<td>80</td>
<td>80</td>
<td>[30–40]</td>
</tr>
<tr>
<td>13</td>
<td>30</td>
<td>80</td>
<td>80</td>
<td>[30–40]</td>
</tr>
<tr>
<td>14</td>
<td>20</td>
<td>55</td>
<td>55</td>
<td>[30–40]</td>
</tr>
<tr>
<td>15</td>
<td>20</td>
<td>55</td>
<td>55</td>
<td>[30–40]</td>
</tr>
</tbody>
</table>
represent an important contribution to DE algorithm set ups.

Methods combining DE and local search can be very effective in solving economic dispatch problems. The search for the best combination of exploitation (convergence speed) and exploration (population diversity) is a constant in the IDE, and our future studies will focus mainly on the conception of IDE approaches with sequential quadratic programming local search for solution of nonlinear economic dispatch problems of electric energy considering the power system transmission losses, line flow constraints and valve point effects modeled as prohibited operating zones.

References


Table 7
Convergence results (50 runs) of case study II considering a fifteen unit system with $P_D = 2630$ MW

<table>
<thead>
<tr>
<th>Method</th>
<th>Minimum cost ($/h)</th>
<th>Mean cost ($/h)</th>
<th>Standard deviation of cost ($/h)</th>
<th>Maximum cost ($/h)</th>
<th>Mean CPU time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DE(1)</td>
<td>32,508.46</td>
<td>32,643.68</td>
<td>$8.9026 \times 10^{-3}$</td>
<td>32,868.35</td>
<td>0.45</td>
</tr>
<tr>
<td>DE(2)</td>
<td>32,425.63</td>
<td>32,638.67</td>
<td>$1.1139 \times 10^{-2}$</td>
<td>32,818.90</td>
<td>0.46</td>
</tr>
<tr>
<td>DE(3)</td>
<td>32,490.46</td>
<td>32,670.16</td>
<td>$8.4388 \times 10^{-3}$</td>
<td>32,821.28</td>
<td>0.45</td>
</tr>
<tr>
<td>IDE(1)</td>
<td>32,418.79</td>
<td>32,638.68</td>
<td>$9.9880 \times 10^{-2}$</td>
<td>32,821.29</td>
<td>0.45</td>
</tr>
<tr>
<td>IDE(2)</td>
<td>32,449.21</td>
<td>32,656.57</td>
<td>$8.2715 \times 10^{-3}$</td>
<td>32,884.91</td>
<td>0.46</td>
</tr>
<tr>
<td>IDE(3)</td>
<td>32,475.96</td>
<td>32,675.77</td>
<td>$8.3224 \times 10^{-3}$</td>
<td>32,886.63</td>
<td>0.46</td>
</tr>
<tr>
<td>Evolution strategy [22]</td>
<td>32,568.54</td>
<td>32,620</td>
<td>0.45</td>
<td>32,710</td>
<td>0.48$^a$</td>
</tr>
<tr>
<td>Particle swarm optimization [16]</td>
<td>33,331</td>
<td>33,039</td>
<td>33,331</td>
<td>26.59$^a$</td>
<td></td>
</tr>
<tr>
<td>Genetic algorithm [16]</td>
<td>33,113</td>
<td>33,228</td>
<td>33,337</td>
<td>49.31$^a$</td>
<td></td>
</tr>
</tbody>
</table>

$^a$ Values obtained from references.

Table 8
Comparison of five methods: best result for the case study II

<table>
<thead>
<tr>
<th>Unit power output (MW)</th>
<th>DE(2) proposed</th>
<th>IDE(2) proposed</th>
<th>Evolution strategy [22]</th>
<th>Particle swarm optimization</th>
<th>Genetic algorithm [16]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_1$</td>
<td>452.538616</td>
<td>451.820142</td>
<td>455.00</td>
<td>439.12</td>
<td>415.31</td>
</tr>
<tr>
<td>$P_2$</td>
<td>376.011372</td>
<td>368.339840</td>
<td>380.00</td>
<td>407.97</td>
<td>359.72</td>
</tr>
<tr>
<td>$P_3$</td>
<td>129.458210</td>
<td>130.000000</td>
<td>130.00</td>
<td>119.63</td>
<td>104.42</td>
</tr>
<tr>
<td>$P_4$</td>
<td>127.406430</td>
<td>124.666587</td>
<td>150.00</td>
<td>129.99</td>
<td>74.98</td>
</tr>
<tr>
<td>$P_5$</td>
<td>152.149464</td>
<td>167.954026</td>
<td>168.92</td>
<td>151.07</td>
<td>380.28</td>
</tr>
<tr>
<td>$P_6$</td>
<td>459.973830</td>
<td>459.922643</td>
<td>459.34</td>
<td>459.99</td>
<td>426.79</td>
</tr>
<tr>
<td>$P_7$</td>
<td>416.784282</td>
<td>419.998757</td>
<td>430.00</td>
<td>425.56</td>
<td>341.32</td>
</tr>
<tr>
<td>$P_8$</td>
<td>68.626514</td>
<td>87.117073</td>
<td>97.42</td>
<td>98.56</td>
<td>124.79</td>
</tr>
<tr>
<td>$P_9$</td>
<td>112.507028</td>
<td>52.948576</td>
<td>30.61</td>
<td>113.49</td>
<td>133.14</td>
</tr>
<tr>
<td>$P_{10}$</td>
<td>129.762104</td>
<td>147.78000</td>
<td>142.56</td>
<td>101.11</td>
<td>89.26</td>
</tr>
<tr>
<td>$P_{11}$</td>
<td>71.599069</td>
<td>75.789536</td>
<td>80.00</td>
<td>33.91</td>
<td>60.06</td>
</tr>
<tr>
<td>$P_{12}$</td>
<td>76.413150</td>
<td>77.707678</td>
<td>85.00</td>
<td>79.96</td>
<td>50.00</td>
</tr>
<tr>
<td>$P_{13}$</td>
<td>25.000000</td>
<td>29.797246</td>
<td>15.00</td>
<td>25.00</td>
<td>38.77</td>
</tr>
<tr>
<td>$P_{14}$</td>
<td>17.262241</td>
<td>21.268395</td>
<td>15.00</td>
<td>41.41</td>
<td>41.94</td>
</tr>
<tr>
<td>$P_{15}$</td>
<td>15.000000</td>
<td>15.000000</td>
<td>15.00</td>
<td>35.61</td>
<td>22.64</td>
</tr>
<tr>
<td>Total power (MW)</td>
<td>2630.54</td>
<td>2630.11</td>
<td>2653.85</td>
<td>2662.41</td>
<td>2688.44</td>
</tr>
<tr>
<td>$P_L$ (MW)</td>
<td>0.54</td>
<td>0.11</td>
<td>23.85</td>
<td>32.42</td>
<td>38.28</td>
</tr>
<tr>
<td>Total cost ($/h)</td>
<td>32,425.63</td>
<td>32,418.79</td>
<td>32,568.54</td>
<td>32,858.00</td>
<td>33,113.00</td>
</tr>
</tbody>
</table>

[16] $^a$ Values obtained from references.