

“© 2010 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.”

# Economic Load Dispatch using Differential Evolution with Double Wavelet Mutation Operations

J.C.Y. LAI, *Member, IEEE*, F.H.F. Leung, *Senior Member, IEEE* and S.H.Ling, *Member, IEEE*

**Abstract**— in this paper, a modified Differential Evolution (DE) that incorporates double wavelet-based operations is proposed to handle a load flow problem. The wavelet based operation is embedded in the DE mutation and crossover operation. In the DE mutation operation, the scaling factor is controlled by a wavelet function. In the DE crossover operation, a wavelet-based mutation operation is embedded in it. The trial population vectors are thus modified by the wavelet function. The double wavelet mutations are applied in order to enhance DE in exploring the high-dimension solution space more effectively for better solution quality and stability. The proposed DE algorithm is employed to solve the Economic Load Dispatch with Valve-Point Loading (ELD-VPL) Problem. It is shown empirically that the proposed method out-performs significantly the conventional methods in terms of convergence speed, solution quality and solution stability.

## I. INTRODUCTION

Differential Evolution (DE) is a population based stochastic optimization algorithm that searches the solution space to find out the solution. DE has been well accepted as a powerful algorithm for handling optimization problems during the last decade [1]. It uses the weighted difference between two population vectors to determine a third vector. No separate probability distribution has to be used so that the scheme is completely self-organizing [1]. It is accepted as a new member to the class of Evolutionary Algorithms (EA). Owing to the population based strategy, EAs are less possibly getting trapped in a local optimum in the solution space. As a result, many researchers view EAs as global optimization algorithms. Important examples of EAs include the Genetic Algorithm (GA) [2] and Evolutionary Programming (EP) [3].

The same as GA, DE guides the population towards the global solution within the given solution space by using evolutionary operations. Comparing with other optimization algorithms, DE is relatively easy to implement. It requires fewer parameters for tuning, and has a relatively fast convergence speed. A simple vector subtraction is able to generate a random direction of movement for the population to search the solution space. DE can also offer a high degree of variations for the population's searching. It has been successfully applied in a wide range of optimization

problems such as data clustering [4], power plant control [5], optimization of non-linear functions [6], etc. However, for maintaining the diversity from one generation of the population to the next, mutation takes an important role in the evolution process. The presence of mutation can help assuring the reached solution is a global optimum; but a too vigorous mutation in every iteration step may slow down or even destroy the convergence of the algorithm.

DE can explore the solution space more widely in the early stage of the search by setting a larger searching space for the mutation and crossover operations. It can obtain a fine-tuned global solution in the later stage of the search by setting a smaller searching space, based on the properties of wavelet [7]. The wavelet is a tool to model seismic signals by combining dilations and translations of a simple, oscillatory function (mother wavelet) of a finite duration [8]. Its properties enable us to improve the performance of DE. In this paper, mutation and crossover operations with a dynamic searching space by incorporating some wavelet function [7] are proposed to handle the problem of Economic Load Dispatch with Valve-Point Loading (ELD-VPL).

The ELD is a method to schedule or control the power generator outputs with respect to the load demands, and to operate the power system economically [11]. We aim at minimizing the operation cost of the power generating system. Because of the valve-point loadings and rate limits, the input-output characteristics of modern generators are nonlinear by nature. As a result, the characteristics of ELD-VPL problems are multimodal, discontinuous, and highly nonlinear.

The double mutation operations in the proposed method aid the DE to perform more efficiently and provide a faster convergence than the standard DE in finding the solution for the ELD-VPL problem. In addition, it can achieve better solution quality and higher solution stability.

This paper is organized as follows. Section II presents the operations of DE and the DE with double wavelet mutations. The problem formulation of ELD-VPL is given in Section III. The experiment details and the testing results are given in Section IV. A conclusion will be drawn in the last Section.

## II. DE WITH DOUBLE WAVELET MUTATIONS

To perform DE, a randomly generated population over the solution space is first obtained. The population of solution vectors are then successively updated and swapped, until the population converges to the optimum. The pseudo code for

J.C.Y. Lai is with the Centre for Signal Processing, Dept. of Electronic and Information Engg., The Hong Kong Polytechnic University, Hung Ham, Hong Kong. (e-mail: 08900438r@polyu.edu.hk).

F.H.F. Leung is with the Centre for Signal Processing, Dept. of Electronic and Information Engg., The Hong Kong Polytechnic University, Hung Ham, Hong Kong. (e-mail: enfrank@inet.polyu.edu.hk).

S.H. Ling is with the Centre for Health Technologies, Faculty of Engineering and Information Technology, University of Technology, Sydney, NSW, Australia. (e-mail: steve.ling@uts.edu.au).

the standard DE process is shown in Fig. 1. In this paper, a DE with double wavelet mutation (DWM-DE) is proposed and the pseudo code is shown in Fig. 2. The details of both the standard DE and the DWM-DE are discussed as follows.

### A. Standard DE Differential Evolution

DE maintains a population of  $N_p$  vectors for each generation of evolution, with each vector contains  $D$  elements of parameters. Let  $P_{x,g}$  be the population of the current generation  $g$ , and  $\mathbf{x}_{i,g}$  be the  $i$ -th vector in this population:

$$\begin{aligned} P_{x,g} &= (\mathbf{x}_{i,g}), i = 0, 1, \dots, N_p - 1; g = 0, 1, \dots, g_{\max} \\ \mathbf{x}_{i,g} &= (x_{j,i,g}), j = 0, 1, \dots, D - 1. \end{aligned} \quad (1)$$

where  $g_{\max}$  is the maximum generation number. Before the population can be initialized over the solution space, the boundary of the searching space should be specified. The population should be uniformly and randomly distributed in the searching space. Once initialized, DE creates a mutated vector,  $\mathbf{v}_{i,g}$  for each target vector  $\mathbf{x}_{i,g}$  by using the mutation operation. In particular, DE adds a scaled, randomly sampled, vector difference to form a third vector. The mutated vector is realized by the following equation:

$$\mathbf{v}_{i,g} = \mathbf{x}_{i,g} + F \cdot (\mathbf{x}_{r_1,g} - \mathbf{x}_{r_2,g}) \quad (2)$$

where  $F$  is the scaling factor;  $r_1$  and  $r_2$  are two different integers which are randomly generated from  $\{0, 1, \dots, N_p - 1\}$ . All vectors in the population perform mutations in each generation. To complement the differential mutation search strategy and increase the diversity of the perturbed parameter vectors, DE employs a method called uniform crossover for the mutated vectors. Each vector element pair  $x_{j,i,g}$  and  $v_{j,i,g}$  generates a new trial vector element  $u_{j,i,g}$ . The crossover operation is realized by the following equation:

$$\mathbf{u}_{i,g} = (u_{j,i,g}) = \begin{cases} (v_{j,i,g}) & \text{if } (\text{rand}_j(0,1) \leq C_r) \\ (x_{j,i,g}) & \text{otherwise.} \end{cases} \quad (3)$$

where  $C_r \in [0, 1]$  is called the crossover rate, which is a user-defined value that controls the fraction of parameters that are copied from the mutant.  $\text{rand}_j(0,1)$  generates a random value between 0 and 1 for the  $j$ -th parameter. The algorithm also ensures  $u_{j,i,g}$  gets at least one parameter value as  $x_{j,i,g}$  [1]. Then the population is updated by comparing each trial vector  $\mathbf{u}_{i,g}$  to the corresponding target vector  $\mathbf{x}_{i,g}$ . If the fitness function value of the trial vector is lower than that of the target vector, replace the target vector in the next generation; otherwise the target vector retains its place in the population for at least one generation of iteration. The selection operation is therefore realized by the following equation:

$$\mathbf{x}_{i,g+1} = \begin{cases} \mathbf{u}_{i,g} & \text{if } f(\mathbf{u}_{i,g}) \leq f(\mathbf{x}_{i,g}) \\ \mathbf{x}_{i,g} & \text{otherwise.} \end{cases} \quad (4)$$

where  $f(\cdot)$  is the fitness function. Because of this selection operation, DE is expected to have high optimization ability. When the condition to stop further evolution is satisfied, for example, a preset maximum number of iteration has been reached, the algorithm ends with the best solution as the final solution (see Fig. 1).

```

begin
  Initialize the population
  While (not termination condition) do
    begin
      Mutation operation by equation (2)
      Crossover operation by equation (3)
      Evaluation of the function
      Select the best vector by equation (4)
    end
  end

```

Fig. 1. Pseudo code for standard DE.

```

begin
  Initialize the population
  While (not termination condition) do
    begin
      Update the new value of F by equation (14)
      Mutation operation by equation (11)
      Crossover operation by equation (3)
      Modifying the trial population vectors by equation (16)
      Evaluation of the function
      Select the best vector by equation (4)
    end
  end

```

Fig. 2. Pseudo code for the DWM-DE.

### B. Differential Evolution with Double Wavelet Mutation (DWM-DE)

In the standard DE mutation operation, the value of  $F$  in (2) is a fixed value within the range of  $[0, 1]$  determined based on the application or the problem nature. The choice of this value relies very much on experience or expert knowledge. Yet, a fixed value of  $F$  takes no advantage of the benefit brought by the evolution. We propose the value of  $F$  to diminish with the increase of the number of iteration. Moreover, for some complex optimization problems like finding the minimum point of a multimodal functions with many local minima, a large number of iteration for solving the problem is required. It reduces the efficiency of the DE. This leads to the proposed DWM-DE in which the value  $F$  is determined by a wavelet function. The freedom for the movement of the trial vector will then be increased. More different vector directions would be generated during the mutation operation. Moreover, in the crossover operation,

we proposed a second wavelet mutation that varies the searching space based on the wavelet function. As the wavelet function output is inversely proportional to the number of iteration; when the searching population is approaching the optimal solution, the effect of the double wavelet mutations will be decreasing until the DE ends eventually (see Fig. 4.) By adopting this method, the effort on searching and evaluating those local optima, which are far away from the global optimum, in the later iteration is reduced. The total number of iteration should also decrease. Thanks to the property of the wavelet function, the solution stability is enhanced in a statistical sense, i.e. the performance of the DE on converging to the optimal point is relatively stable despite the presence of many random factors during the evolution. The result is a hybrid DE with double wavelet mutation operations (DWM-DE).

### C. Double Wavelet Mutation

#### 1. Wavelet theory

Certain seismic signals can be modelled by combining translations and dilations of an oscillatory function with a finite duration called a “wavelet”. A continuous function  $\psi(x)$  is called a “mother wavelet” or “wavelet” if it satisfies the following properties:

*Property 1:*

$$\int_{-\infty}^{+\infty} \psi(x) dx = 0 \quad (5)$$

In other words, the total positive momentum of  $\psi(x)$  is equal to the total negative momentum of  $\psi(x)$ .

*Property 2:*

$$\int_{-\infty}^{+\infty} |\psi(x)|^2 dx < \infty \quad (6)$$

Hence, most of the energy in  $\psi(x)$  is confined to a finite duration and bounded. The Morlet wavelet [2], as shown in Fig. 3, is an example mother wavelet:

$$\psi(x) = e^{-x^2/2} \cos(5x) \quad (7)$$

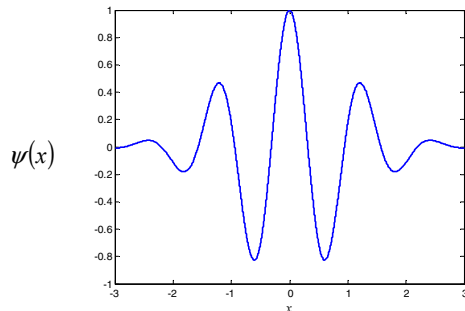


Fig. 3. Morlet wavelet.

The Morlet wavelet integrates to zero (*Property 1*). Over 99% of the total energy of the function is contained in the interval of  $-2.5 \leq x \leq 2.5$  (*Property 2*). In order to control the magnitude and the position of  $\psi(x)$ , a function  $\psi_{a,b}(x)$  is defined as follows.

$$\psi_{a,b}(x) = \frac{1}{\sqrt{a}} \psi\left(\frac{x-b}{a}\right) \quad (8)$$

where  $a$  is the dilation parameter and  $b$  is the translation parameter. Notice that

$$\psi_{1,0}(x) = \psi(x) \quad (9)$$

$$\psi_{a,0}(x) = \frac{1}{\sqrt{a}} \psi\left(\frac{x}{a}\right). \quad (10)$$

It follows that  $\psi_{a,0}(x)$  is an amplitude-scaled version of  $\psi(x)$ . Fig. 4 shows different dilations of the Morlet wavelet. The amplitude of  $\psi_{a,0}(x)$  will be scaled down as the dilation parameter  $a$  increases. This property is used to do the mutation operation in order to enhance the searching performance.

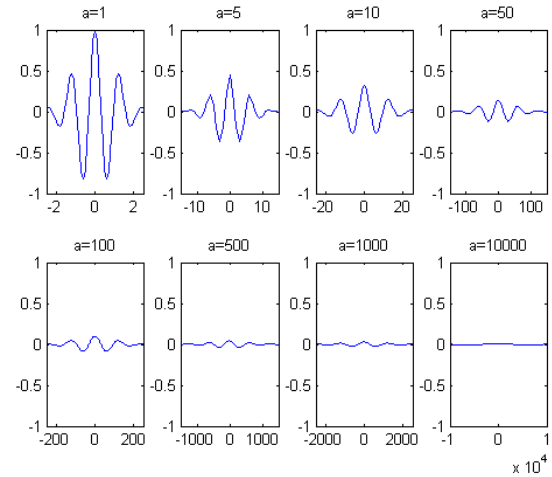


Fig. 4. Morlet wavelet dilated by different values of the parameter  $a$  (x-axis:  $a$ , y-axis:  $\psi_{a,0}(x)$ .)

#### 2. Operation of wavelet mutation

The vectors in the population are mutated based on a proposed wavelet mutation (WM) operation, which exhibits a fine-tuning property. The details of the operation are as follows. Modify the mutation operation (2) as follows.

$$\mathbf{v}_{i,g} = \mathbf{x}_{i,g} + F \cdot \left( \mathbf{x}_{r_1,g} - \mathbf{x}_{r_2,g} \right), \quad (11)$$

where

$$F = \psi_{a,0}(\varphi), \quad (12)$$

$$F = \frac{1}{\sqrt{a}} \psi\left(\frac{\varphi}{a}\right) \quad (13)$$

By using the Morlet wavelet of (7) as the mother wavelet,

$$F = \frac{1}{\sqrt{a}} e^{-\left(\frac{\varphi}{a}\right)^2 / 2} \cos\left(5\left(\frac{\varphi}{a}\right)\right) \quad (14)$$

where

$$a = e^{-\ln(\lambda) \times \left(1 - \frac{t}{T}\right)^{\zeta_{wm}} + \ln(\lambda)} \quad (15)$$

$T$  is the total number of iteration and  $t$  is the current iteration number;  $\zeta_{wm}$  is the shape parameter of the monotonic increasing function,  $\lambda$  is the upper limit of the parameter  $a$ . If  $F$  is positive approaching 1 or  $F$  is negative approaching  $-1$ , the mutation will tend to a maximum. Conversely, when  $F$  approaches 0, the mutation will tend to a minimum. A larger values of  $|F|$  gives a larger searching space for the solution. When  $|F|$  is small, it gives a smaller searching space for fine-tuning.

### 3. Operation of wavelet crossover

The crossover operation of (3) is done with respect to the elements of the trial vector (after mutation) in DE. In DWM-DE, the second stage wavelet mutation is embedded in the crossover operation. It exhibits a fine-tuning ability. The details are as follows. The crossover after the first mutation takes place according to (3). Let  $\mathbf{u}_{i,g} = (u_{0,i,g}, u_{1,i,g}, \dots, u_{D-1,i,g})$  (where  $g$  is the current generation number and  $D$  is the number of elements in the vector) be the  $i$ -th vector after crossover for the second wavelet mutation. Its element value is inside the vector element's boundary  $[para_{\min}^j, para_{\max}^j]$ . The mutated crossover vector is given by  $\bar{\mathbf{u}}_{i,g} = (\bar{u}_{0,i,g}, \bar{u}_{1,i,g}, \dots, \bar{u}_{D-1,i,g})$ , and

$$u_{j,i,g} = \begin{cases} u_{j,i,g} + \sigma \times (para_{\max}^j - u_{j,i,g}) & \text{if } \sigma > 0 \\ u_{j,i,g} + \sigma \times (u_{j,i,g} - para_{\min}^j) & \text{if } \sigma \leq 0 \end{cases} \quad (16)$$

$$\sigma = \frac{1}{\sqrt{a}} \psi\left(\frac{\varphi}{a}\right) \quad (17)$$

where the same Morlet wavelet in (8) is used as the mother wavelet and the value of  $a$  is governed by (15). Similar to  $F$  of (14), a larger value of  $|\sigma|$  at the early stage of evolution gives a larger searching space for the solution; when  $|\sigma|$  is small at the later stage of evolution, the algorithm gives a smaller searching space for fine-tuning.

After the operations of the double wavelet mutations, the population is updated by comparing each trial vector  $\bar{\mathbf{u}}_{i,g}$  to the corresponding target vector  $\mathbf{x}_{i,g}$  using the method of standard DE as given by (4). A new population is generated

and the same evolution process is repeated. Such an iterative process will be terminated when a defined number of iteration has been met.

## III. ELD-VPL PROBLEM

The ELD is a method to control or schedule a group of power generator outputs with respect to the load demands, and operate a power system economically so as to minimize the operation cost of the power system. Because of the valve-point loadings and rate limits, the input-output characteristics of modern generators are nonlinear by nature. As a result, the characteristics of ELD-VPL problems are multimodal, discontinuous, and highly nonlinear. In this paper, the DWM-DE is employed to solve the ELD-VPL problem, which can be formulated into the following objective function:

$$\min \sum_{i=1}^n C_i(P_{Li}) \quad (18)$$

where  $C_i(P_{Li})$  is the operation fuel cost of generator  $i$ , and  $n$  denotes the number of generators. The problem is subject to balance constraints and generating capacity constraints as follows:

$$D_L = \sum_{i=1}^n P_{Li} - P_{Loss} \quad (19)$$

$$P_{L_{i,\min}} \leq P_{Li} \leq P_{L_{i,\max}}, i = 1, 2, \dots, n \quad (20)$$

where  $D_L$  is the load demand,  $P_{Li}$  is the output power of the  $i$ -th generator,  $P_{Loss}$  is the transmission loss, and  $P_{L_{i,\max}}$  and  $P_{L_{i,\min}}$  are the maximum and minimum output power of the  $i$ -th generator, respectively.

The operation fuel cost function with valve-point loadings of the generators is given by

$$C_i(P_{Li}) = a_i P_{Li}^2 + b_i P_{Li} + c_i + \left| e_i \times \sin(f_i \times (P_{L_{i,\min}} - P_{Li})) \right| \quad (21)$$

where  $a_i$ ,  $b_i$ , and  $c_i$  are the coefficients of the cost curve of the  $i$ -th generator, and  $e_i$  and  $f_i$  are the coefficients of the valve point loadings. The generating units with multivalve steam turbines exhibit a greater variation in the fuel cost functions. The valve-point effects introduce ripples in the heat-rate curves.

To solve the ELD-VPL problem by using DWM-DE, the solution representation of elements in the population is defined as follows:

$$\mathbf{P} = [P_{L_1} P_{L_2} P_{L_3} \dots P_{L_{n-1}}] \quad (22)$$

where  $n$  denotes the number of generators. From (18), we have

$$P_{L_n} = D_L - \sum_{i=1}^{n-1} P_{L_i} + P_{Loss} \quad (23)$$

In this paper, the power loss is not considered. Therefore  $P_{Loss} = 0$ , and

$$P_{L_n} = D_L - \sum_{i=1}^{n-1} P_{L_i} \quad (24)$$

To ensure that  $P_{L_n}$  falls within the range  $[P_{L_{n,\min}}, P_{L_{n,\max}}]$ , the following conditions are considered:

$$\text{If } P_{L_n} > P_{L_{n,\max}} \begin{cases} P_{L_i} = P_{L_i} + (P_{L_n} - P_{L_{n,\max}}) \\ P_{L_n} = P_{L_{n,\max}} \end{cases} \quad (25)$$

$$\text{If } P_{L_n} < P_{L_{n,\min}} \begin{cases} P_{L_i} = P_{L_i} - (P_{L_n} - P_{L_{n,\min}}) \\ P_{L_n} = P_{L_{n,\min}} \end{cases} \quad (26)$$

It should be noted that (25) and (26) are used to control each individual power generator. If the value of  $P_{L_i}$  is outside the constraint boundary (i.e. the safety range), the exceeding portion of the power required will be shared by other generators to ensure that the output power of all individual generators are working within the safety range. Based on the problem defined above, the objective of this optimization problem is to minimize the cost function of (21) by using DWM-DE.

#### IV. EXPERIMENT AND RESULT

In this section, different kinds of DE methods are applied to a 40-generator system, which is adopted as an example in [11]. The resulting system is a very large one with nonlinearities. The load demand ( $D$ ) of the problem is 10500 MW. Standard DE [1], DE/local-to-best/1 [10] [9], DE/rand/1 with per-vector-dither [10] and the proposed DWM-DE are used to solve the ELD-VPL problem. The following simulation conditions are used:

- The shape parameter of the wavelet mutation ( $\zeta_{wm}$ ): It is chosen by trial and error through experiments for good performance.  $\zeta_{wm} = 1$  is used in this experiment.
- The parameter  $\lambda$  for the monotonic increasing function: 10000.
- The number of population vectors:  $N_p = 50$
- Initial population: It is generated uniformly at random.
- Crossover rate:  $C_r = 0.5$
- The mutation weight factor (For standard DE, DE/local-to-best/1, DE/rand/1 with per-vector-dither):  $F = 0.5$
- The number of iteration: 600

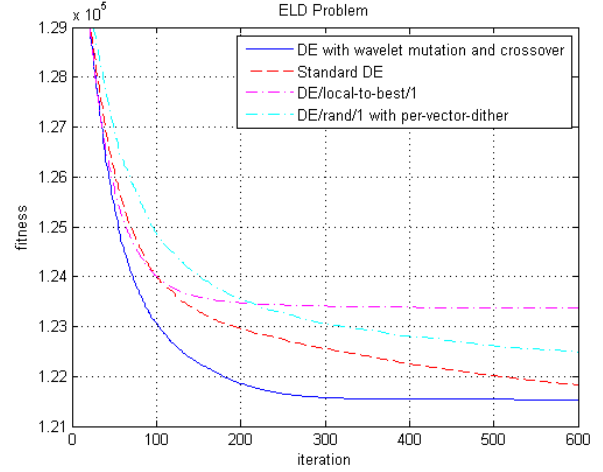


Fig. 5 The fitness curve of ELD Problem.

Table 1. The result.

	DWM-DE	standard DE	DE/local-to-best/1	DE/rand/1 with per-vector-dither
Mean	<b>121537.822643</b>	121834.625136	123363.297947	122490.902309
Best	<b>121449.229377</b>	121530.992191	121971.298961	122188.146437
Std	<b>54.658543</b>	172.746304	610.103640	89.645727
Dev				

All results shown are averaged data out of 50 trials. The dimension of ELD-VPL problem is 39. The statistical results in terms of the mean cost value, the best cost value, and the standard deviation are shown in Table 1. The convergence rates of different tested methods are shown in Fig. 5. From the result obtained in this experiment, we find that the DWM-DE performs much better than the other methods in terms of convergence rate, cost and standard DE deviations. The DWM-DE can offer the best (minimum) cost among the methods. The average cost for the 40-generator system is \$121,537.82, and the best (minimum) cost is \$121,449.23. The convergence rate of DWM-DE is much higher than that of the other methods. After 500 times of iteration, the DWM-DE has nearly reached the minimum cost. The smallest standard deviation is obtained by using DWM-DE. Thanks to the wavelet properties, the stability of the optimization solution is improved. To conclude, the convergence speed, solution quality and solution stability of the DWM-DE are good.

#### V. CONCLUSION

In this paper, we have proposed an improved Differential Evolution (DE) that incorporates double wavelet-based mutations to handle a Load Flow Problem. The double wavelet mutations are applied in order to enhance DE in exploring the high-dimension solution space more effectively for better solution quality and stability. The ELD-VPL problem is solved by the proposed DE algorithm. It is shown empirically that the proposed method outperforms significantly the conventional methods in terms of convergence speed, solution quality and solution stability.

## ACKNOWLEDGEMENT

The work described in this paper was substantially supported by a grant from The Hong Kong Polytechnic University (Project No. RP9L).

## REFERENCES

- [1] R. Storn and K. Price "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, pp. 341–359, 1997.
- [2] L. Fogel, "Evolutionary programming in perspective: The top-down view," *Computational Intelligence: Imitating Life*. Piscataway, NJ: IEEE Press, 1994.
- [3] D. Goldberg, *Genetic Algorithms in Search Optimization and Machine Learning*. Addison-Wesley, 1989.
- [4] S. Paterlini and T. Krink, "High performance clustering with differential evolution," in *Proc. IEEE Congress on Evolutionary Computation*, vol. 2, 2004, pp. 2004–2011.
- [5] J.H. van Sickle, K.Y. Lee, and J.S. Heo, "Differential evolution and its applications to power plant control," in *Proc. Intelligent Systems Applications to Power Systems 2007*, (ISAP 2007), 5-8 Nov. 2007, pp.1 – 6.
- [6] B. Babu and R. Angira, "Optimization of non-linear functions using evolutionary computation," in *Proc. 12th ISME International Conference on Mechanical Engineering*, India, 2001, pp. 153–157.
- [7] I. Daubechies, *Ten lectures on Wavelets*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1992.
- [8] S.H. Ling, C.W. Yeung, K.Y. Chan, H.H.C. Iu, and F.H.F. Leung, "A new hybrid particle swarm optimization with wavelet theory based mutation operation," in *Proc. 2007 IEEE Congress on Evolutionary Computation (CEC 2007)*, Singapore, Sep. 25-28, 2007, pp. 1977-1984.
- [9] S. Rahnamayan, H.R. Tizhoosh, and M.M.A. Salama, "Opposition-based differential evolution," *IEEE Trans. Evol. Comput.*, vol. 12, no. 1, pp 64-79, Feb. 2008.
- [10] Y. Ao, and H. Chi , "Experimental study on differential evolution strategies," in *Proc. Global Congress on Intelligent Systems.*, vol. 2, pp. 19-24, May. 2009.
- [11] P.H. Chen and H.C. Chang, " Large-scale economic dispatch by genetic algorithms," *IEEE Trans. Power Syst.*, vol. 10, no. 1, pp. 117–124, Feb. 1995.