Solving Economic Load Dispatch problems using Differential Evolution with Opposition Based Learning

SUREKHA P, Dr.S.SUMATHI
Department of Electrical and Electronics Engineering
PSG College of Technology
Peelamedu, Coimbatore - 641004
INDIA
surekha_3000@yahoo.com

Abstract: - This paper presents a Differential Evolution algorithm combined with Opposition Based Learning (DE-OBL) to solve Economic Load Dispatch problem with non-smooth fuel cost curves considering transmission losses, power balance and capacity constraints. The proposed algorithm varies from the Standard Differential Evolution algorithm in terms of three basic factors. The initial population is generated through the concept of Opposition Based Learning, applies tournament based mutation and uses only one population set throughout the optimization process. The performance of the proposed algorithm is investigated and tested with two standard test systems, the IEEE 30 bus 6 unit system and the 20 unit system. The experiments showed that the searching ability and convergence rate of the proposed method is much better than the standard differential evolution. The results of the proposed approach were compared in terms of fuel cost, computational time, power loss and individual generator powers with existing differential evolution and other meta-heuristics in literature. The proposed method seems to be a promising approach for load dispatch problems based on the solution quality and the computational efficiency.

Key-Words: - Differential Evolution with Opposition Based Learning, Standard Differential Evolution, Economic Load Dispatch, solution quality, robustness

1 Introduction

Economic Load Dispatch (ELD) is one of the most significant optimization problems in modern computer aided power system design. The ELD problem finds the optimum allocation of load among the committed generating units subject to satisfaction of power balance and capacity constraints, such that the total cost of operation is kept at a minimum [1]. Various methods and investigations are being carried out until date in order to produce a significant saving in the operational cost. Conventional techniques like Lambda Iteration method [2], dynamic programming [3], mixed integer programming [4], branch and bound [5], gradient-based method, [6] and Newton's method [7] were used earlier to obtain optimal dispatch to the ELD problems.

In lambda iteration and gradient based methods, the solution to ELD is obtained by approximately representing the cost function for individual generators in terms of single quadratic function. These techniques require incremental fuel cost curves which are piecewise linear and monotonically increasing to find the global optimal solution [8]. For generators with non-monotonically incremental cost curves, conventional methods ignores or flattens out portions of incremental cost curve that are not continuous or monotonically increasing [9], [10]. Newton-based methods are not capable of obtaining quality solutions for ELD problems due to highly non-linear characteristics and large number of constraints. Though dynamic programming is capable of solving non-linear and discontinuous problems, it suffers from the problem of curse of dimensionality with large computational time [11].

These limitations of conventional methods were overcome by modern meta-heuristic approaches like Artificial Neural Networks (ANN) [12], Genetic Algorithms (GA) [13], Tabu Search (TS) [14], Simulated Annealing (SA) [15], Particle Swarm Optimization (PSO) [16], Ant colony optimization (ACO) [17], Artificial immune systems (AIS) [18], Differential Evolution (DE) [19], Bacterial Foraging Algorithm (BFA) [20], Intelligent Waterdrop (IWD) [8] and Bio-geography based optimization (BBO) [21] [22] algorithms. Though these methods are not capable in attaining global best optimal solutions to the ELD problems, to a great extent they produce near optimal solutions. Later several hybridizations and improvements were imposed on the meta-heuristics to obtain faster convergence and quality solutions for ELD problems. Some of these approaches in literature include Simulated Annealing – Particle Swarm Optimization (SA-PSO) [23], Quantum-inspired version of the PSO using the harmonic oscillator (HQPSO) [24], Self-organizing hierarchical particle swarm optimization (SOH-PSO) [25], Bacterial foraging with Nelder–Mead algorithm (BFA-NM) [20], Adaptive Particle Swarm Optimization (APSO) [26], Uniform design with the genetic algorithm (UHGA) [27], Particle swarm optimization with chaotic and Gaussian approach (PSO-CG) [28], Self Tuning Hybrid Differential Evolution (STHDE) [29], variable Scaling Hybrid Differential Evolution (VSHDE) [30], Improved genetic algorithm with multiplier updating (IGAMU) [31], Differential evolution with sequential quadratic programming (DEC-SQP) [32], and Improved fast evolutionary programming (IFEP) [33].

Differential Evolution (DE) is one of the most significant optimization technique proposed by Storn and Price [34] to reveal consistent and reliable performance in non-linear and multimodal environment. They have proved to be efficient for constrained optimization problems [35]. In [19], the authors proposed the classical DE for solving ELD problems with specialized constraint handling mechanisms. Khamsawang et. Al.,[36] proposed the original DE for ELD with regenerated population technique and tuning of parameters. Wang et. Al., [29] used the concept of the 1/5 success rule of evolutionary strategies in the original Hybrid DE (HDE) to accelerate the search for the global optimum in ELD problems. The need for fixed and random scale factors in HDE was overcome by the work of Chiou et. Al., [30], in which a variable scaling factor was added to HDE thus improving the search for the global solution for ELD problems. Mariani et. Al., [32] proposed a hybrid technique that combined the differential evolution algorithm with the generator of chaos sequences and sequential quadratic programming technique. Aniruddha et. Al.,[22] offered a hybrid combination of DE with BBO to accelerate the convergence speed and to improve the quality of the ELD solutions.

In this paper, we propose an Differential Evolution with Opposition Based Learning (DE-OBL) algorithm for solving the ELD problems. The major improvements made to the exisiting standard DE (SDE) are:

- *Initialization* –Population initialization is based on opposition based learning rather than the random method
- *Mutation* – The mutant individual is selected based on tournament selection
- *Population* – Parent and the individuals after reproduction are compared based on fitness and the better ones are maintained in one population, in contrast to two sets in SDE

The idea of Opposition Based Learning (OBL) for DE was proposed by Rahnamayan et.Al., [37]. For a problem under consideration, the estimated and the opposite of estimated solutions are chosen and it has been mathematically proved that opposite numbers to the initial set of random numbers are more likely to be closer to the optimal solution rather than purely random solutions. The advantages of the proposed method are convergence speed, robustness, and the ease in application of opposite points rather than random ones. This paper presents the application of DE-OBL to solve the ELD problems of two test systems namely IEEE 30 bus 6 unit and 20 unit systems, whose generating units are characterized by non-convex operational features including transmission losses. Solving this practical optimization problem leads to a minimized total generation cost of operating the two respective power systems in the presence of generator capacity and power balance constraints.

Section II of this paper provides the nomenclature of symbols used and section III presents a brief mathematical description of the ELD problem. The basic DE, concept of OBL, and proposed DE-OBL are explained in Section IV. The experimental results and comparative analysis for the two test systems are detailed in Section V. The conclusion and future scope are presented in Section VI.

2 Nomenclature

F_T	Fuel cost of the system
F_i	Fuel cost of the i^{th} generating unit of the system
P_{Gi}	Power generated in the i^{th} generating unit
N	Number of generators
a_i, b_i, c_i	Cost coefficients of the i^{th} generator
P_D	Power demand
P_L	Transmission losses
$P_{Gi\min}$	Minimum value of the real power
$P_{Gi\max}$	Maximum value of the real power
$X_j^{\min}$	Lower bound of initial population for jth component

X_j^{max} Upper bound of initial population for j[th] component
N_P Number of individuals in population P
$rand[0,1]$ Uniform random number in the interval [0,1]
D Dimension
P Initial population
P_{add} Additional population to create new population for DE-OBL
P_{new} New population for DE-OBL
X_{ra}, X_{rb} and X_{rc} Random individuals for mutation
F Scaling factor for mutation
C_r Crossover constant
$f(x)$ Fitness function

3 ELD Problem Formulation

The principal objective of the economic load dispatch problem is to find a set of active power delivered by the committed generators to satisfy the required demand subject to the unit technical limits at the lowest production cost. The optimization of the ELD problem is formulated in terms of the fuel cost expressed as,

$$F_T = \sum_{i=1}^{n} F_i(P_{Gi}) = \sum_{i=1}^{n} a_i + b_i P_{Gi} + c_i P_{Gi}^{\ 2} \quad (1)$$

Subject to the equality constraint,

$$\sum_{i=1}^{N} P_{Gi} = P_D + P_L \quad (2)$$

Subject to the inequality constraint,

$$P_{Gi\min} \le P_{Gi} \le P_{Gi\max} \quad (3)$$

4 Proposed Methodology

The basic function of the SDE algorithm and the concept of the Opposition based learning are described in this section. Followed by the brief introduction to the concepts, the implementation of DE-OBL and its application to ELD problem is explained in detail.

4.1 Standard Differential Evolution

The SDE algorithm is a stochastic population based algorithm similar to Genetic Algorithms (GA) using the operators; crossover, mutation and selection. The key dissimilarity between GA and SDE is that GAs rely mostly on crossover while SDE relies on mutation operation. The algorithm uses mutation operation as a search mechanism and selection operation to direct the search toward the prospective regions in the search space [34]. Mutation in SDE uses differences of randomly sampled pairs of solutions in the population and greediness may be embedded in it. The SDE algorithm also uses a non-uniform crossover that can take child vector parameters from one parent more often than it does from others. By using the components of the existing population members to construct trial vectors, the recombination (crossover) operator efficiently shuffles information about successful combinations, enabling the search for a better solution space. An optimization problem consisting of N parameters can be represented by an N-dimensional vector. In SDE, a population of Np solution vectors is randomly created at the initialization stage. This population is successfully improved by applying mutation, crossover and selection operators thus evaluating the objective function or the fitness function. A brief description of different steps of SDE is given below.

Initialization - An initial population of candidate solutions is formed by assigning random values to each decision parameter of every individual in the population, dimension of each vector being N, according to the rule,

$$X_{i,j}^{(0)} = X_j^{\min} + rand\,[0,1] \times \left(X_j^{\max} - X_j^{\min}\right), \quad i = 1,2,\ldots,N_P \ and \ j = 1,2,\ldots,D \quad (4)$$

Mutation – Three distinct individuals are chosen in random from the population such that $ra \ne rb \ne rc \ne i$ and mutation is performed according to

$$V_i^{G+1} = X_{ra}^{G} + F \times \left[X_{rb}^{G} - X_{rc}^{G}\right], i = 1,2,\ldots N_P \quad (5)$$

where X_{ra}^{G} can be any random individual among the selected three and F is the scaling factor.

Crossover – The current population member $X_{i,j}^{G}$ and the mutated member $V_{i,j}^{G+1}$ are subject to crossover, to generate a set of trial vectors as follows:

$$U_{i,j}^{G+1} = \begin{cases} V_{i,j}^{G+1}, if \ rand[0,1] \le C_r \\ X_{i,j}^{G}, otherwise \end{cases} \quad (6)$$

Selection – Compute the fitness function value of the new individual and select the best individual for the next generation.

4.2 Opposition Based Learning

In general, heuristic optimization methods start with few initial solutions in a population and try to improve them towards optimal solutions during generations. The optimization process terminates when some predefined criteria are satisfied. Without any a priori information about the solutions to the problem under consideration, the optimization starts with a set of random presumptions. The chance of obtaining a fitter solution can be attained through the opposite solution. By monitoring the opposite solution, the fitter presumed solution can be chosen as an initial solution. In fact, according to probability theory, 50% of the time a presumption is further from the solution than its opposite presumption. Therefore, based on the fitness, two close presumption has the potential to accelerate convergence. This approach is not only applied to initial solutions but also continuously to each solution in the current population.

Consider a point $P=(x_1,x_2,\ldots,x_n)$, with D-dimensional space consisting of candidate solutions. Let $f(.)$ be the fitness function used to measure the fitness of the candidate solutions. If $x_i \in [p_i, q_i]\ \forall\ i=1,2,...,D$ represents a real number, then the opposite points of x_i (denoted as $\breve{x}_i$) is defined as

$$\breve{x}_i = p_i + q_i - x_i \qquad (7)$$

Based on Eq. (7), $\breve{P}=(\breve{x}_1,\breve{x}_2,\ldots,\breve{x}_n)$ represents the opposite of $P=(x_1,x_2,\ldots,x_n)$. If $f(\breve{P}) \geq f(P)$, then P can be replaced with $\breve{P}$, otherwise the optimization procedure continues with P. Thus the point and its opposite point are evaluated simultaneously in order to continue the generations with the fitter individuals.

4.4 Proposed DE-OBL for ELD

Though SDE has emerged as one of the most popular technique for solving optimization problem, it has been observed that the convergence rate of SDE does not meet the expectations in case of multi-objective problems. Hence certain modifications using the concept of opposition based learning, and random localization are performed on the SDE. The proposed DE-OBL varies from the basic SDE in terms of the following factors:

- DE-OBL uses the concept of opposition based learning in the initialization phase while SDE uses the uniform random numbers for initialization of population
- During mutation, DE-OBL chooses the best individual among the three points as the mutant individual whereas in SDE, a random choice is made with equal choice of any of the three being selected.
- SDE uses two sets of population – current population and an advanced population for next generation individuals. DE-OBL uses only one population set and the same population is updated as the best individuals are found.

The steps of the proposed algorithm are explained below:

Initialization: The basic step in the DE-OBL optimization is to create an initial population of candidate solutions by assigning random values to each decision parameter of each individual of the population. A population P consisting of N_P individuals is constructed in a random manner such that the values lie within the feasible bounds $X_j^{\min}$ and $X_j^{\max}$ of the decision variable, according to the following rule,

$$X_{i,j}^{(0)} = X_j^{\min} + rand[0,1] \times \left(X_j^{\max} - X_j^{\min}\right), \quad i=1,2,\ldots,N_P \ and\ j=1,2,\ldots,D \qquad (8)$$

where $rand[0,1]$ represents a uniform random number in the interval [0,1], $X_j^{\min}$ and $X_j^{\max}$ are the lower and upper bounds for the j^{th} component respectively, D is the number of decision variables. Each individual member of the population consists of an N-dimensional vector $X_i^{(0)} = \{P_1, P_2, \ldots, P_N\}$ where the i^{th} element of $X_i^{(0)}$ represents the power output of the i^{th} generating unit.

An additional population P_{add} is constructed using the rule,

$$Y_{i,j}^{(0)} = X_j^{\min} + X_j^{\max} - P_{i,j}, \qquad (9)$$

where $P_{i,j}$ denotes the points of population P. The new population P_{new} for the proposed approach is formed by combining the best individuals of both populations P and P_{add} as follows

$$P_{new} = X_{i,j}^{(0)} \cup Y_{i,j}^{(0)} \qquad (10)$$

Mutation: Next generation offspring are introduced into the population through the mutation process. Mutation is performed by choosing three individuals from the population P_{new} in a random manner. Let X_{ra}, X_{rb} and X_{rc} represent three random individuals such that $ra \neq rb \neq rc \neq i$, upon which mutation is performed during the G^{th} generation as,

$$V_i^{G+1} = X_{best}^{G} + F \times \left[X_{rb}^{G} - X_{rc}^{G}\right], i=1,2,\ldots N_P \qquad (11)$$

where V_i^{G+1} is the perturbed mutated individual and X_{best}^{G} represents the best individual among three random individuals. The difference of the remaining two individuals is scaled by a factor F, which controls the amplification of the difference between two individuals so as to avoid search stagnation and to improve convergence.

Crossover: New offspring members are reproduced through the crossover operation based on binomial distribution. The members of the current population (target vector) $X_{i,j}^{G}$ and the members of the mutated individual $V_{i,j}^{G+1}$ are subject to crossover operation thus producing a trial vector $U_{i,j}^{G+1}$ according to,

$$U_{i,j}^{G+1} = \begin{cases} V_{i,j}^{G+1}, \text{if } rand[0,1] \le C_r \\ X_{i,j}^{G}, \text{otherwise} \end{cases} \quad (12)$$

where C_r is the crossover constant that controls the diversity of the population and prevents the algorithm from getting trapped into the local optima. The crossover constant must be in the range of [0 1]. $C_r = 1$ implies the trial vector will be composed entirely of the mutant vector members and $C_r = 0$ implies that the trial vector individuals are composed of the members of parent vector. Eq. (12) can also be written as

$$\mathrm{U}_{i,j}^{G+1} = \mathrm{X}_{i,j}^{G} \times (1 - \mathrm{C}_r) + \mathrm{V}_{i,j}^{G+1} \times \mathrm{C}_r \quad (13)$$

Selection: Selection procedure is performed with the trial vector and the target vector to choose the best set of individuals for the next generation. In this proposed approach, only one population set is maintained and hence the best individuals replace the target individuals in the current population. The objective values of the trial vector and the target vector are evaluated and compared. For minimization problems like ELD, if the trial vector has better value, the target vector is replaced with the trial vector as per,

$$X_i^G = \begin{cases} U_i^{G+1}, \text{if } f(U_i^{G+1}) \le f(X_i^G) \\ X_i^G, \text{otherwise} \end{cases}; \text{ for } i = 1,2,\ldots,N_P \quad (14)$$

Fitness evaluation: The objective function for the ELD problem based on the fuel cost and power balance constraints is framed as

$$f(x) = \sum_{i=1}^{N} F_i(P_i) + k\left(\sum_{i=1}^{N} Pi - P_D - P_L\right) \quad (15)$$

where k is the penalty factor associated with the power balance constraint, $F_i(P_i)$ is the ith generator cost function for output power P_i, N is the number of generating units, P_D is the total active power demand and P_L represents the transmission losses. For ELD problems without transmission losses, setting k=0 is most rational, while for ELD including transmission losses, the value of k was set to 1.

The pseudocode of the proposed approach is shown below:

Generate an initial population P *randomly with each individual representing the power output of the ith generating unit according to Eqn (8).*
Generate an additional population P_{add} *according to Eqn (9)*
Obtain the new population P_{new} *as per Eqn (10)*
Evaluate fitness for each individual in P_{new} *based on Eqn (15)*
While termination criteria not satisfied
  For i = 1 to N_P
  Mutate random members in P_{new} *to obtain* V_i^{G+1}
  Perform crossover on X_i^G *and* $U_{i,j}^{G+1}$
  Evaluate fitness function of X_i^G *and* U_i^{G+1}
  If $f(U_i^{G+1}) \le f(X_i^G)$
    Replace existing population with U_i^{G+1}
  End if
  End for
End While

5 Experimental Results and Analysis

The efficiency of the proposed algorithm for solving Economic Load Dispatch (ELD) problem has been tested on two different power generating units – the 6 unit and the 20 unit system including the transmission losses. The performances of these algorithms are evaluated and compared with classical Lambda Iteration Method (LIM) and other meta-heuristics available in literature. The algorithms are implemented in MATLAB R2008b platform on i3 processor, 2.53 GHz, 4 GB RAM personal computer.

5.1 Test System I – IEEE 30 bus system

The IEEE 30 bus six unit test system has been adopted from [38], in which the fuel cost coefficients, and power limits are known. The specifications of the system for six generator test system are detailed in Table 1. The system is found to have minimum and maximum generation capacity of 117 MW and 435 MW, respectively.

Table 1 Fuel cost coefficients and power limits for IEEE 30 bus test system

Unit no.	a_i (\$/hr)	b_i (\$/MW hr)	c_i (\$/MW^{2}hr)	P_{Gimax} (MW)	P_{Gimin} (MW)
1	.00375	2	0	50	200
2	.01750	1.75	0	20	80
3	.06250	1	0	15	50
4	.00834	3.25	0	10	35
5	.02500	3	0	10	30
6	.02500	3	0	12	40

The transmission loss coefficient denoted as B_{ij} is given according to Eq. (16) as,

$$B_{ij} = \begin{bmatrix} 0.000218 & 0.000103 & 0.000009 & -.000010 & 0.000002 & 0.000027 \\ 0.000103 & 0.000181 & 0.000004 & -.000015 & 0.000002 & 0.000030 \\ 0.000009 & 0.000004 & 0.000417 & -.000131 & -.000153 & -.000107 \\ -.000140 & -.000015 & -.000131 & 0.000221 & 0.000094 & 0.000050 \\ 0.000002 & 0.000002 & -.000153 & 0.000094 & 0.000243 & 0.000000 \\ 0.000027 & 0.000030 & -.000107 & 0.000050 & 0.000000 & 0.000358 \end{bmatrix} \quad (16)$$

Table 2 Parameters of DE used to implement ELD for six unit system

Parameters of DE	Notations used	Values
1. No. of members in population	N_P	[20,100]
2. Vector of lower bounds for initial population	X_j^{min}	[-2,-2]
3. Vector of upper bounds for initial population	X_j^{max}	[2,2]
4. Number of iterations	Iter	200
5. Dimension	D	5
6. Crossover Rate	Cr	[0,1]
7. Step size	F	[1,2]
8. Strategy parameter	DE/best/2/bin	9
9. Refresh parameter	R	10
10. Value to Reach	VTR	1.e-6

The generalized DE-OBL parameters and their settings for the ELD problem are listed in Table 2. For optimal parameters, simulations were carried out for 50 trials each time varying the basic parameters like scale factor (F), Crossover rate (Cr) and population size (P). The effect of these parameters on the IEEE 30 bus system for a demand of 283.4 MW is shown below.

Effect of population size

The population size is related with the problem dimension and complexity. The population size was varied between [20,100] and the results are shown in Table 3. Experiments were repeated for 50 trials for each population size and it was found that a size of 80 was more consistent in obtaining the global optimal solution. The corresponding standard deviation was also computed and it was found very low for the population size of 80 which implies that most of the best solutions are very close to the optimal value.

Effect of F and C_r

The parameter F controls the speed and robustness of the search, i.e., a lower value of F increases the convergence rate but also increases the risk of getting stuck into a local optimum. On the other hand, if F > 1.0 then solutions tend to be more time consuming and less reliable. The parameter Cr which controls the crossover operation can also be thought of as a mutation rate, i.e., the probability that a variable will be inherited from the mutated individual. The role of Cr is to provide a means of exploiting decomposability.

Table 3 Effect of population size on IEEE 30 bus system

Population size	Min Cost (\$/hr)	Max cost (\$/hr)	Mean cost (\$/hr)	SD	CPU Time (s)	No. of hits for min cost
20	794.9129	794.9212	794.9188	0.03929	1.326	43
40	794.9129	794.9758	794.9144	0.009042	2.6988	44
60	794.9129	794.9668	794.914	0.007702	3.8064	46
80	**794.9129**	**794.9273**	**794.9133**	**0.002294**	**4.992**	**49**
100	794.9129	794.9385	794.9134	0.003668	5.9592	47

Table 4 Influence of F and C_r on IEEE 30 bus system

C_r	F = 0	0.2	0.4	0.6	**0.8**	1
0.1	795.733	794.9767	794.9728	794.9675	794.9773	794.91309
0.2	795.84346	794.9654	794.9662	794.9189	794.9514	794.913626
0.3	781.54936	794. 9628	794.9735	794.9598	794.9228	794.912854
0.4	796.2055	794. 9422	794.9577	794.9308	794.7974	794.912863
0.5	796.41323	794.9138	794.9423	794.9907	794.9458	794.912904
0.6	800.1263	794.9251	794.9385	794.9522	794.9328	794.912858
0.7	802.30951	795.1291	794.9601	794.9582	794.9185	794.91326
0.8	796.58186	795.4591	794.9809	794.9254	**794.9129**	794.913771
0.9	803.46569	797.4425	794.9190	794.9391	794.9131	794.91919
1	813.48962	802.1391	794.9337	794.9859	794.9138	794.916266

Table 5 Results using DE-OBL for IEEE 30 bus system

P_D(MW)	117	150	200	250	283.4	300	350	400	435
P_{G1} (MW)	50	75.96083	116.3169	155.6807	181.6329	200	200	200	200
P_{G2} (MW)	20.09735	27.23823	35.96627	44.4934	50.12272	46.33002	78.99963	80	80
P_{G3} (MW)	15	15	16.14298	18.56025	20.15867	20.26121	24.39775	21.89781	50
P_{G4} (MW)	10	10	10	10	10	12.44659	10.00169	29.49016	35
P_{G5} (MW)	10	10	10	10	10.46971	10.06413	25.93428	30	30
P_{G6} (MW)	12	12	12	12	12	12	12	40	40
Fuel cost ($/ MW hr)	292.6102	378.5813	521.9338	680.5186	794.9129	828.6273	1027.465	1229.463	2805.379
Total P_G (MW)	117.0974	150.1991	200.4261	250.7343	284.384	301.102	351.3334	401.388	435
P_L (MW)	0.090852	0.189186	0.41102	0.714077	0.960433	1.101951	1.332519	1.387965	1.400663
CPU Time (s)	1.54441	1.57561	1.51321	1.669211	1.466409	1.825212	1.638011	1.57561	1.56001

Table 6 Comparison of results for IEEE 30 bus system

Heuristic Algorithms	Output Power (MW)						Fuel cost ($/hr)	Total power P_G (MW)	Power loss (MW)	CPU time (s)
	P_{G1}	P_{G2}	P_{G3}	P_{G4}	P_{G5}	P_{G6}				
DE-OBL	181.6329	50.12272	20.15867	10	10.46971	12	**794.9129**	284.384	**9.30433**	1.466409
LIM	174.3403	56.89421	29.66026	10	10	12	808.9491	292.8948	9.48889	25.9063
HGA	176.2358	49.0093	21.5023	21.8115	12.3387	12.0129	802.465	292.9105	9.5105	NA
EP	176.1522	48.8391	21.5144	22.1299	12.2435	12	802.404	292.8791	9.4791	NA
FGA	189.613	47.745	19.5761	13.8752	10	12	799.823	292.8093	9.6897	0.125
PS	175.727	48.6812	21.4282	22.8313	12.0667	12	802.015	292.7344	9.3349	NA
GA	179.367	44.24	24.61	19.9	10.71	14.09	803.699	292.917	9.5177	315
GA-PS	175.6627	48.6413	21.4222	22.6219	12.3806	12	802.0138	292.7287	9.3286	NA
ACO	177.863	43.8366	20.893	23.1231	14.0255	13.1199	803.123	292.8611	9.4616	20
DE	177.3	49.18	12.24	11.19	21.23	21.74	802.23	292.88	NA	NA
SADE_ALM	176.1522	48.8391	21.5144	22.1299	12.2435	12	802.404	292.8791	9.4791	NA
WIPSO	177.1567	48.6905	21.3013	20.9714	11.9314	12.0078	799.1665	292.0591	8.66	15.453
ABC	176.88	49.54	21.69	2l.71	10.92	12.15	801.881	271.18	NA	8.94

*NA – Data Not available in reported literature

In this paper, an extensive study was carried out for selecting the most suitable DE-OBL parameter set for the chosen problem. In order to select the most suitable {F, Cr} pair, P was fixed to 80, with a load demand of 283.4 MW, and experimented by varying $F \in [0,1]$ and $Cr \in [0.1,1]$ with a step size of 0.2 and 0.1 for F and C_r respectively. To assure convergence maximum generations (MAXGEN=500) was allowed in every experimental run. The results of the influence of Cr and F are shown in Table 4. The results suggest that for most of the Cr and F settings, DE is capable of exhibiting better performance. However, the best settings are F=0.8 and C_r=0.8 corresponding to the minimum cost of 794.9129 $/hr.

Simulation Results of Test System I

With the best values of P = 80, F = 0.8 and Cr = 0.8 obtained from Tables 3, and 4, the DE-OBL algorithm was run for different values of demand ranging between 117 MW and 435 MW. For each demand, 50 independent trials with 500 iterations per trial have been performed. The individual generator powers, minimum fuel cost, total power generated, power loss and the computational time required to obtain the simulation results are shown in Table 5.

Comparative Analysis

The results of the proposed DE-OBL for IEEE 30 bus system are compared with other reported approaches such as Hybrid GA (HGA) [39], Evolutionary Programming (EP) [40], Fast GA (FGA) [38], Pattern Search (PS) [41], GA [42], GA-PS [41], Ant Colony Optimization (ACO) [43], DE [47], Self-Adaptive Differential Evolution with Augmented Lagrange Multiplier method (SADE_ALM) [46], Weight Imrpoved PSO (WIPSO) [44], and Artificial Bee Colony (ABC) [45]. The economic dispatch obtained through the Lambda iteration method (LIM) was also used for comparison and all the results are shown in Table 6. The minimum cost for the demand of 283.4 MW reported so far in the literature was 799.1665 $/hr [44], compared to all others, while the proposed DE-OBL produced a cost of 794.9129 $/hr, promisingly optimal and consistent. The power loss during the optimal dispatch was 9.30433 MW relatively less than all other meta-heuristic algorithms.

5.2 Test System II – 20 Unit System

In order to demonstrate the effectiveness of the DE-OBL algorithm, the ELD benchmark consisting of twenty generator units [12] is selected. The details of fuel cost coefficients and generating limits for each unit are given in Table 7. The maximum and minimum power generating limits of the system are 3865 MW and 1010 MW, respectively.

The Transmission Loss Coefficient Matrix for calculating power loss of 20 Unit test system can be obtained from [12]. The various DE-OBL parameters used to implement ELD problem for 20 unit generating system is similar to that of the six unit test system except for the dimension which is varied based on the size of the problem. Here D=19 for 20 unit system and the population is usually set based on 10 times the D value. Notations of the parameters and the range of values are given in the Table 2.

Table 7 Fuel cost coefficients and power limits for twenty unit test system

Unit no.	a_i ($/hr)	b_i ($/MW hr)	c_i ($/MW^2 hr)	P_{Gimax} (MW)	P_{Gimin} (MW)
1	0.00068	18.19	1000	600	150
2	0.00071	19.26	970	200	50
3	0.00650	19.80	600	200	50
4	0.00500	19.10	700	200	50
5	0.00738	18.10	420	160	50
6	0.00612	19.26	360	100	20
7	0.0079	17.14	490	125	25
8	0.00813	18.92	660	150	50
9	0.00522	18.27	765	200	50
10	0.00573	18.92	770	150	30
11	0.00480	16.69	800	300	100
12	0.00310	16.76	970	500	150
13	0.00850	17.36	900	160	40
14	0.00511	18.70	700	130	20
15	0.00398	18.70	450	185	25
16	0.00712	14.26	370	80	20
17	0.0089	19.14	480	85	30
18	0.00713	18.92	680	120	30
19	0.00622	18.47	700	120	40
20	0.00773	19.79	850	100	30

Effect of population size

To determine the best choice of population size for the twenty unit system with a demand of 2500 MW, the DE-OBL algorithm was run with different values for 30 independent trials. The minimum, maximum and the mean cost were determined along with the standard deviation and simulation time. The results are shown in Table 8 and the best value of population size was 40 resulting in minimum mean cost during 28 hits out of 30 trials.

Effect of F and Cr

For a population size of 40, the crossover probability C_r is increased from 0.1 to 0.9 in steps of 0.1. The scale factor is increased from 0 to 1 in steps of 0.2 and the results are tabulated in Table 9. The best values of C*r* and F were found to be 0.6 and 0.8 respectively at a minimum generation cost 518276.4353 $/hr.

Simulation Results for Test System II

The power demands are varied between [1010,3865] for the 20-unit system. For each value of P_D, 30 trials are performed with 500 iterations per trial, the results are shown in Table 10.

Table 8 Effect of population size on 20 unit system

Population size	Min Cost ($/hr)	Max cost ($/hr)	Mean cost ($/hr)	SD	CPU Time (s)	No. of hits for min cost
20	518276.4	549391.4	519245.5	4695.932	1.6068	22
40	**518276.4**	**521859.2**	**518369.1**	**517.3652**	**3.12**	**28**
60	518276.4	539991.1	518815	3104.772	4.5708	24
80	518276.4	537961.5	518843.8	2857.256	5.8344	26
100	518276.4	521805	518376.6	519.0833	7.0512	27

Table 9 Influence of F and C_r on 20 unit system

C_r	F					
	0	**0.2**	**0.4**	**0.6**	**0.8**	**1**
0.1	577970.6004	518276.453	518276.5673	518277.5870	518276.4668	518276.5902
0.2	630381.2086	518277.1892	518276.5567	518276.5673	518276.4556	518276.4527
0.3	659122.3723	580981.5683	518276.5378	518276.5433	518276.4553	518276.4658
0.4	652899.8416	562270.4641	518276.5189	518276.4980	518276.4521	518495.8300
0.5	650428.2807	577368.9461	518276.4890	518647.0872	518276.4478	518890.2527
0.6	659347.8982	626230.2365	518411.8924	518276.4753	**518276.4353**	518276.4736
0.7	745288.9464	703185.9761	526067.2981	518276.4389	518276.4390	518647.0872
0.8	773636.0369	737310.0134	583023.8938	518890.2527	518276.4412	518647.0874
0.9	785706.711	732704.1988	678034.9648	518647.0872	518276.4418	518925.9563
1	851667.4802	800142.605	651293.8501	562128.0087	518276.4365	522158.4855

Table 10 Results using DE for twenty generator test system

P_D (MW)	1010	1500	2000	2500	3000	3500	3865
P_{G1} (MW)	150.0005	261.5139	439.0248	600	600	600	600
P_{G2} (MW)	50	50	198.9982	200	200	200	200
P_{G3} (MW)	117.8433	200	200	200	200	200	200
P_{G4} (MW)	50	84.40388	200	200	200	200	200
P_{G5} (MW)	50	50	160	160	160	160	160
P_{G6} (MW)	20	47.36409	100	100	100	100	100
P_{G7} (MW)	25	25	25	83.94886	83.94887	83.94885	83.94887
P_{G8} (MW)	50	150	150	150	150	150	150
P_{G9} (MW)	50	50	200	200	200	200	200
P_{G10} (MW)	126.7899	150	150	150	150	150	150
P_{G11} (MW)	100	100	100	100	100	100	100
P_{G12} (MW)	150	150	150	150	150	150	150
P_{G13} (MW)	40	40	55.85962	132.3652	132.3652	132.3652	132.3652
P_{G14} (MW)	20	20	20	20	20	20	20
P_{G15} (MW)	25	25	104.6126	185	185	185	185
P_{G16} (MW)	20	62.68257	80	80	80	80	80
P_{G17} (MW)	30	46.44614	85	85	85	85	85
P_{G18} (MW)	30	120	120	120	120	120	120
P_{G19} (MW)	40	40	120	120	120	120	120
P_{G20} (MW)	30	100	100	100	100	100	100
Fuel cost ($/hr)	35511.5	47331.11	66494.72	518276.4	1018276	1518276	1883276
Total power	1026.065	1541.207	2065.542	2592.214	3124.486	3683.671	3865
Power loss (MW)	16.0651	41.20678	65.54228	92.21373	124.4856	183.6713	214.3426
CPU Time (s)	1.62241	1.856412	1.747211	1.981213	1.981213	2.012413	1.918812

Table 11 Comparative Analysis for 20 unit test system

Parameters	DE-OBL	LIM	CLIM	SHN	BBO	PSO	IWD
P_{G1} (MW)	600	470.6366	512.7805	512.7804	513.09	563.3155	563.32
P_{G2} (MW)	200	50	169.1033	169.1035	173.35	106.5639	106.56
P_{G3} (MW)	200	151.1845	126.8898	126.8897	126.92	98.7093	98.71
P_{G4} (MW)	200	97.11856	102.8657	102.8656	103.33	117.3171	117.32
P_{G5} (MW)	160	97.77008	113.6836	113.6836	113.77	67.0781	67.08
P_{G6} (MW)	100	55.68459	73.5710	73.5709	73.07	51.4702	51.47
P_{G7} (MW)	83.94886	125	115.2878	115.2876	114.98	47.7261	47.73
P_{G8} (MW)	150	150	116.3994	116.3994	116.42	82.4271	82.43
P_{G9} (MW)	200	68.82129	100.4062	100.4067	100.69	52.0884	52.09
P_{G10} (MW)	150	150	106.0267	106.0267	100	106.5097	106.51
P_{G11} (MW)	100	194.5108	150.2394	150.2395	148.98	197.9428	197.94
P_{G12} (MW)	150	337.2191	292.7648	292.7647	294.02	488.3315	488.33
P_{G13} (MW)	132.3652	151.1625	119.1154	119.1155	119.58	99.9464	99.95
P_{G14} (MW)	20	20	30.8340	30.8342	30.55	79.8941	79.89
P_{G15} (MW)	185	103.9979	115.8057	115.8056	116.45	101.525	101.53
P_{G16} (MW)	80	80	36.2545	36.2545	36.23	25.8380	25.84
P_{G17} (MW)	85	51.67328	66.8590	66.8590	66.86	70.0153	70.02
P_{G18} (MW)	120	98.43284	87.9720	87.9720	88.55	53.9530	53.95
P_{G19} (MW)	120	98.48716	100.8033	100.8033	100.98	65.4271	65.43
P_{G20} (MW)	100	42.17147	54.3050	54.3050	54.27	36.2552	36.26
Fuel cost ($/hr)	**518276.4**	63295.81	62456.6391	62456.6341	62456.79	59804.05	59799
Total power P_G (MW)	2592.214	2593.871	2537.662	2591.967	2592.11	2512.3343	2512.34
Power loss (MW)	92.21373	93.83006	91.9670	91.967	92.11	92.3343	92.33
CPU time (s)	1.981213	1232.1	33.757	6.355	6.93		3.9

Comparative Analysis

The optimal dispatch of the test case II was computed through the lambda iteration method. The results of the proposed method for 20 unit system are compared against the results obtained in reported heuristic methods like SHN [12], BBO [1], PSO [28], IWD [8] and the classical LIM [12]. For a demand of 2500 MW, the fuel cost computed through the proposed DE-OBL is 518276.4 $/hr, comparatively much lesser than other reported heuristic algorithms as shown in Table 11.

5.3 Summary of Discussions

The results obtained for the 6 unit and the 20 unit systems have proved that DE-OBL is efficient in producing the optimal dispatch when compared with several heuristic methods. The consequences of the output based on the solution quality, generation costs, robustness and efficiency are summarized in this section.

Solution quality **-** Solution quality is justified based on the key optimizing parameter for ELD problems, the total operating cost. The results obtained for both the test systems have showed that the proposed DE-OBL method is suitable for producing the best compromise solution in terms of fuel cost. Table 6 shows that the best competent solutions in terms of fuel cost and power loss for IEEE 30 bus system are obtained by the DE-OBL when compared with the classical DE [47] and other algorithms. Similarly, Table 11 also emphasizes that DE-OBL is more suitable for larger unit power systems generating minimum operational cost. The characteristic features of the DE-OBL like simple, compact structure, and high convergence nature has motivated the algorithm in attaining quality solutions for the ELD problems.

Testing of robustness **-** The performance of any heuristic search based optimization algorithm is best judged through repetitive runs in order to compare the robustness and consistency of the algorithm. For this specific goal, the frequency of convergence to the minimum cost at different ranges of generation cost with fixed load demand is to be recorded. Experimental results show that the frequency of convergence, for a 6 unit system, using DE-OBL, towards the optimal fuel cost was 49 out of 50 trial runs for all power demands. Similarly, for the 20 unit system, 30 trials were repeated and it was observed that the convergence rate of DE towards the optimal cost was 28 out of 30.

Computational efficiency - Apart from yielding the optimal solution, it may also be noted that DE-OBL yields the minimum cost at a comparatively lesser time of execution. It may be observed from Table 6 and 11, that the average computational time of DE-OBL in test systems I and II is much less than the compared heuristics optimization techniques. Hence the proposed DE-OBL is computationally more efficient in terms of speed of convergence.

6. Conclusion

The DE-OBL algorithm had been implemented to solve the ELD problems. The main motivation of the current work is to use the notion of opposition to accelerate the SDE. It has been observed from the results of test systems I and II, that DE-OBL is capable in achieving optimal quality solutions with speedy convergence characteristics. With high dimension problems such as test case II, the solution quality, and computational efficiency of DE-OBL outperforms other method. It is clear from the results obtained through several trials, that the implementation of DE-OBL overcomes the effect of premature convergence, exhibited by other heuristic optimization techniques. The idea of proposing the DE-OBL is to introduce a new version of opposition optimization through meta-heuristic algorithms like SDE. Possible directions for future work include proposing OBL concepts into mutation in SDE and other heuristics like GA, PSO and ACO.

References:

1. Lakshmi Devi A., Vamsi Krishna O. "Combined economic and emission dispatch using Evolutionary algorithms-a case study." *ARPN Journal of Engineering and Applied Sciences* 3, no. 6, 28-35, 2008.
2. C.L., Wadhwa. *Electrical Power Systems*. New Delhi: New Age International (p) Limited Publishers, 2000.
3. Lowery, P. G. "Generation unit commitment by dynamic programming." *IEEE Trans. Power App.Syst.*, vol. PAS-102, 1218–1225, 1983.
4. Wilson, J. A. Muckstadt and R. C. "An application of mixed-integer programming duality to scheduling thermal generating systems." *IEEE Trans. Power App. Syst* vol. PAS-87, no. 12, 1968-1978, 1968.
5. Yoshimura, A. I. Cohen and M. "A branch-and-bound algorithm for unit commitment." *IEEE Trans. Power App. Syst.* vol. PAS-102, no. 2, 444–451, 1983.
6. J. B. Park, K. S. Lee, J. R. Shin and K. Y. Lee. "A particle swarm optimization for economic dispatch with non smooth cost functions." *IEEE Trans. on Power Systems* vol. 8, no. 3, 1325-1332, August 1993.
7. Wood J., Wollenberg B. F. *Power generation operation and control.* John Wiley & Sons, 1984.
8. Rayapudi, S. Rao. "An Intelligent Water Drop Algorithm for Solving Economic Load Dispatch Problem." International Journal of Electrical and Electronics Engineering vol. 5, no. 1, 43-49, 2011.
9. H. T. Yang, P. C. Yang and C. L. Huang. "Evolutionary Programming Based Economic Dispatch For Units With Non-smooth Fuel Cost Functions." *IEEE Transactions on Power Systems* vol. 11, no. 1, 112-118, 1996.
10. T. Jayabarathi, G. Sadasivam and V. Ramachandran. "Evolutionary programming based economic dispatch of generators with prohibited operating zones." *Electric Power Systems Research* vol. 52, no. 3, 261-266, 1999.
11. Glover, Z. X. Liang and J. D. "A zoom feature for a dynamic programming solution to economic dispatch including transmission losses." *IEEE Trans. on Power Systems* vol. 7, no. 2, 544-550, May 1992.
12. Lin, C.-T. Su and C.-T. "New approach with a Hopfield modeling framework to economic dispatch." IEEE Trans. Power Syst. vol.15, no. 2, 541, May 2000.
13. G. Damousis, A. G. Bakirtzis, and P. S. Dokopoulos. "Network-constrained economic dispatch using real-coded genetic algorithm." IEEE Trans. Power Syst. vol. 18, no. 1, 198–205, Feb 2003.
14. W.-M. Lin, F.-S. Cheng, and M.-T. Tsay. "An improved tabu search for economic dispatch with multiple minima." *IEEE Trans. Power Syst* vol. 17, 108 – 112, February 2002.
15. Wong, K. P. Wong and Y. W. "Genetic and genetic/simulated-annealing approaches to economic dispatch." *Proc. Inst. Elect. Eng. Gen. Trans. Distrib.* vol. 141, 507-513, September 1994.
16. Gaing, Z.-L. "Particle swarm optimization to solving the economic dispatch considering the generator constraints,”." IEEE Trans. Power Syst. vol.18, no. 3, 1187–1195, Aug 2003.
17. Y. H. Hou, Y. W. Wu, L. J. Lu, and X. Y. Xiong. "Generalized ant colony optimization for economic dispatch of power systems." Proc.Int.

Conf. Power System Technology, Power-Con., 225–229, Oct. 2002.
18. K. Panigrahi, S. R. Yadav, S. Agrawal. "A clonal algorithm to solve economic load dispatch." Elect. Power Syst. Res. vol.77, no. 10, 1381–1389, 2007.
19. Iba, N. Nomana and H. "Differential evolution for economic load dispatch problems." Elect. Power Syst. Res. vol.78, no. 3, 1322–1331, 2008.
20. Pandi, B. K. Panigrahi and V. R. "Bacterial foraging optimisation: Nelder-Mead hybrid algorithm for economic load dispatch." IET Gen.,Transm., Distrib., vol.2, no. 4, 556-565, 2008.
21. Chattopadhyay, A. Bhattacharya and P. K. "Biogeography-based optimization for different economic load dispatch problems." IEEE Trans. Power Syst. vol. 25, no. 2, 1064 – 1077, May 2010.
22. Aniruddha Bhattacharya, P.K. Chattopadhyay, “Hybrid Differential Evolution with Biogeography-Based Optimization for Solution of Economic Load Dispatch." IEEE Trans. Power Syst. vol. 25. No. 4, 1955-1964, Nov 2010.
23. Kuo, C. C. "A novel coding scheme for practical economic dispatch by." IEEE Trans. Power Syst. vol. 23, no. 4, 1825–1835, Nov. 2008.
24. Mariani, L. D. S. Coelho and V. C. "Particle swarm approach based on quantum mechanics and harmonic oscillator potential well for economic load dispatch with valve-point effects." Energy Convers. Manage. vol. 49, no. 11, 3080-3085, 2008.
25. K. T. Chaturvedi, M. Pandit, and L. Srivastava. "Self-organizing hierarchical particle swarm optimization for nonconvex economic dispatch." IEEE Trans. Power Syst. vol.23, no. 3, 1079-1087, Aug 2008.
26. K. Panigrahi, V. R. Pandi, and S. Das. "Adaptive particle swarm optimization approach for static and dynamic economic load dispatch." Energy Convers. Manage, vol. 49, no. 6, 1407-1415, 2008.
27. D.-K. He, F.-L. Wang, and Z.-Z. Mao. "Hybrid genetic algorithm for economic dispatch with valve-point effect." Elect. Power Syst. Res., vol. 78, no. 4, 626-633, 2008.
28. Lee, L. D. S. Coelho and C.-S. "Solving economic load dispatch problems in power systems using chaotic and Gaussian particle swarm optimization approaches." Int. J. Elect. Power Energy Syst., vol.30, no. 4, 297–307, 2008.
29. S.-K. Wang, J.-P. Chiou, and C.-W. Liu. "Non-smooth/non-convex economic dispatch by a novel hybrid differential evolution algorithm." IET Gen., Transm., Distrib., vol. 1, no. 5, 793–803, 2007.
30. Chiou, J.-P. "Variable scaling hybrid differential evolution for large scale economic dispatch problems." Elect. Power Syst. Res., vol. 77, no. 1, 212–218, 2007.
31. Chiang, C.-L. "Genetic-based algorithm for power economic load dispatch." IET Gen., Transm., Distrib., vol. 1, no. 2, 261–269, 2007.
32. Mariani, L. D. S. Coelho and V. C. "Correction to “combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect." IEEE Trans. Power Syst. vol.21, no. 3, 1465–1465, Aug 2006.
33. N. Sinha, R. Chakrabarti, and P. K. Chattopadhyay. "Evolutionary programming techniques for economic load dispatch." IEEE Trans. Evol. vol.7, no. 1, 83–94, Feb 2003.
34. R. Storn, K.V. Price. "Differential evolution a simple and efficient heuristic for global optimization over continuous spaces." J. Global Optimization vol.11, no. 4, 341–359, 1997.
35. K.V. Price, R.M. Storn, J.A. Lampinen. Differential Evolution: A Practical Approach to Global Optimization. Berlin, Heidelberg: Springer, 2005.
36. Jiriwibhakorn, S. Khamsawang and S. "Solving the Economic Dispatch Problem by Using Differential Evolution." International Journal of Electrical and Electronics Engineering vol.3, no. 10, 641-645, 2009.
37. Shahryar Rahnamayan, H.R. Tizhoosh, M.M.A.Salama. "Opposition based differential evolution." IEEE Congress on evolutionary computation, 2010 – 2017, 2006.
38. Sailaja Kumari M., Sydulu M. "A Fast Computational Genetic Algorithm for Economic Load Dispatch." International Journal of Recent Trends in Engineering vol.1, no. 1, 349-356, May 2009.
39. Mary, N. Thenmozhi and D. "Economic emission load dispatch using hybrid Genetic Algorithm." Chiang Mai, Thailand, 476-479, 2004.
40. J. Yuryevich, K. P. Wong. "Evolutionary Programming Based Optimal Power Flow Algorithm." IEEE Transaction on power systems vol.14, no. 4, 1245 – 1250, Nov 1999.
41. Y. Labbi, D. Ben Attous. "A hybrid GA–PS method to solve the economic load dispatch problem", Journal of Theoretical and Applied

Information Technology vol.15, no. 1, 61-68, 2010.
42. Tarek Bouktir, Linda Slimani, M. Belkacemi. "A Genetic Algorithm for Solving the Optimal Power Flow Problem." Leonardo Journal of Sciences, no. 4, 44-58, June 2004.
43. Boumediène Allaoua, Abdellah Laoufi. "Optimal Power Flow Solution Using Ant Manners for Electrical Network." Advances in Electrical and Computer Engineering vol.9, no. 1, 34-40, 2009.
44. PhanTu Vu, DinhLuong Le, NgocDieu Vo, Tlusty Josef. "A novel weight-improved particle swarm optimization algorithm for optimal power flow and economic load dispatch problem", 1 – 7, 2010.
45. Sumpavakup, I. Srikun, and S. Chusanapiputt. "A solution to the Optimal Power Flow using Artificial Bee Colony algorithm." IEEE. Proc. Int. Conf. Power System Technology . Hangzhou, Oct 2010. 1-5.
46. Thitithamrongchai, B. Eua-arporn. "Self-adaptive Differential Evolution Based Optimal Power Flow for Units with Non-smooth Fuel Cost Functions." Journal of Electrical Systems vol.3, no. 2, 88-99, 2007.
47. K.Vaisakh, L.R.Srinivas. "Differential Evolution Approach for Optimal Power Flow Solutions." Journal of Theoretical and Applied Information Technology vol.4, no. 4, 261-268, 2008.