ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645
Indian Journal of Science and Technology, *Vol 11(41), DOI: 10.17485/ijst/2018/v11i41/108579, November 2018*

Quasi-oppositional Grey Wolf Optimizer Algorithm for Economic Dispatch

Diptanu Das*, Aniruddha Bhattacharya and Rup Narayan Ray

Department of Electrical Engineering, National Institute of Technology Agartala, Purba Barjala, Jirania – 799046, Tripura, India;
diptanuonline@yahoo.co.in, bhatta.aniruddha@gmail.com, rupnarayan_r@yahoo.co.in

Abstract

Objectives: To minimize the fuel price of generator while satisfying different constraints. Valve point effects and multiple fuel option are also considered in some cases. **Methods**: Quasi-Oppositional Grey Wolf Optimizer algorithm is applied here for solving different economic dispatch problems. Grey Wolf Optimizer is a meta-heuristic method, motivated by social behaviour of grey wolves. Quasi-Oppositional learning is implemented in the present work within Grey Wolf Optimizer for improving the quality of solution in minimum time. Quasi-opposite numbers are used within the algorithm in place of normal random numbers for improving the convergence speed. **Findings**: The proposed technique is applied to six different systems to test the efficiency of the algorithm. Simulation results obtained by this method are compared with those obtained by some well-known optimization methods to show the robustness and superiority of this technique. Simulation results also show that the computational efficiency of Quasi-Oppositional Grey Wolf optimizer is better as compared to several previously developed optimization methods. **Improvement**: It is found that the convergence speed, success rate, efficiency and solution quality of the proposed algorithm is improved.

Keywords: Economic Load Dispatch, Oppositional Based Learning, Quasi-oppositional Grey Wolf Optimizer, Valve Point Effects

1. Introduction

Minimization of power generation cost of fossil fuel based plants is a big challenge for the power engineers. Therefore, in recent decades, researchers have given much attention for minimizing the cost of power production. The objective of Economic Load Dispatch (ELD) is to minimize the cost of power generation while satisfying various equality and inequality constraints. Many techniques have been developed to solve the economic dispatch problems. Erstwhile, many classical optimization methods like Linear programming[1], Dynamic Programming[2], Lagrangian method[3] etc. have been developed to solve ELD problems but they suffer from various limitations when deals with non-smooth cost functions.

A practical ELD problem considers the effect of prohibited zone, ramp rate limit and valve point. So the problem becomes a complex optimization problem which is very difficult to solve by traditional methods. In recent years, various artificial intelligence methods like Genetic Algorithm (GA)[4], Simulated annealing[5], Cuckoo search algorithm[6], Particle Swarm Optimization (PSO)[7], Modified Artificial Bee Colony Optimization (MABCA)[8], Differential Evolution (DE)[9], Fuzzy Adaptive Chaotic Ant Swarm Optimization (CASO)[10], Real Coded Chemical

**Author for correspondence*

Reaction Optimization (RCCRO)[11], Oppositional Real Coded Chemical Reaction Optimization (ORCCRO)[12], Biogeography-based Optimization (BBO)[13], Teaching Learning based Optimization (TLBO)[14], Exchange Market Algorithm (EMA)[15], One Rank Cuckoo Search Algorithm (ORCSA)[16], Real Coded Genetic Algorithm (RCGA)[17], Krill Herd Algorithm (KHA)[18], Grey Wolf Optimizer (GWO)[19], Ant Lion Optimization (ALO)[20], Kinetic Gas Molecule Optimization (KGMO)[21] have been proposed by the researchers for solving non-convex ELD problems due to their ability of searching global optimum solution for non-smooth cost function.

The techniques mentioned above may be fruitful for solving non-convex economic load dispatch problems. In any case, these algorithms do not ensure that the solution is the global best solutions. Therefore, various hybridisations and modifications of DE, PSO, GA, EP, BBO have been made. Some important hybrid methods like Improved fast evolutionary programming[22], DE with generator of chaos sequences and sequential quadratic programming[23], Directional Search GA (DSGA)[24], New PSO with local random search (NPSO_LRS)[25], Variable scaling hybrid differential evolution[26], Bacterial foraging with Nelder–Mead algorithm[27], Self-organising hierarchical PSO[28], Improved coordinated aggregation-based PSO[29], Improved PSO[30], Hybrid differential evolution with BBO (DE/ BBO)[31], Oppositional Invasive Weed Optimization (OIWO)[32], Hybrid chemical reaction optimization with DE (HCRO-DE)[33], Modified Differential Evaluation (MDE)[34], Hybrid ACO-ABC-HS[35] etc, have been developed for solving various types of economic load dispatch problems.

In 2014, have developed a new meta-heuristic optimization method called Grey Wolf Optimizer (GWO)[36]. This method is inspired by the hunting behaviour of grey wolves. This optimization technique has been tested on 29 well-known benchmark functions[36] and it was found that GWO algorithm gives better result as compared to other well-known optimization techniques.

Oppositional Based Learning (OBL)[37] has been proposed by Tizhoosh in order to improve back propagation in neural networks. In order to approach the solution, OBL exploits the opposite numbers. By contrasting a number compared to the opposite number, a compact search space is required to obtain the correct solution. It has been demonstrated that a quasi-opposite number[38] is likely to be nearer to the solution as compared to an arbitrary number. It has additionally been demonstrated that a quasi-opposite number is typically nearer to the solution compared to an opposite number. As quasi-opposition based learning is proved to have improved computational efficiency, the present authors have adopted this methodology in GWO (QOGWO) for accelerating the speed of convergence of GWO to a greater extent. In this paper, the Quasi Oppositional Grey Wolf Optimizer (QOGWO) algorithm is used for solving various ELD problems and results obtained by QOGWO method are compared to other optimization techniques. The details of this proposed technique have been discussed in section 3.

Section 2 gives the problem formulation and brief description of various economic load dispatch problems. Section 3 describes a short description of GWO algorithm. Short description about QOGWO algorithm is explained in Section 4. Simulation results are discussed in Section 5. The conclusion is described in Section 6.

2. Problem Formulation

2.1 Objective Function

2.1.1 Quadratic Cost Function of ELD

The objective function[12] of economic dispatch problem for this case can be written as

$$C_T = \min \sum_{i=1}^{N} C_i(\mathrm{W}_i) = \min \sum_{i=1}^{N} a_{1i} + b_{1i} W_i + c_{1i} W^2{}_i \tag{1}$$

where, $C_i(W_i)$ is the cost function of i^{th} generator; a_{1i}, b_{1i}, c_{1i} are the coefficients of fuel cost of i^{th} unit; N represents number of units; W_i is the power output of i^{th} unit.

2.1.2 Economic Dispatch with Valve Point Loading Effect

The overall objective function C_T of ELD with valve point[12] can be expressed as follows

$$C_T = \left(\sum_{i=1}^{N} C_i(W_i)\right) = \left(\sum_{i=1}^{N} a_{1i} + b_{1i}W_i + c_{1i}W_i^2 + \left|E_{1i} \times \sin\left\{F_{1i} \times (W_i^{\min} - W_i)\right\}\right|\right) \tag{2}$$

where, E_{1i} and F_{1i} represents the coefficients of unit i reflecting the effect of valve point.

2.1.3 Fuel Price Function Considering the Effect of Valve –point and Multiple Fuel

In a network, if N represents number of generator and n_F is the fuel option of individual unit, then the generator fuel price function considering the effect of valve point and multiple fuel can be represented by

$$C_{ip}(W_i) = \sum_{i=1}^{N}\left(a_{1ip} + b_{1ip}W_i + c_{1ip}W_i^2 + \left|E_{1ip} \times \sin\{F_{1ip} \times (W_{ip}^{\min} - W_i)\}\right|\right) \tag{3}$$

if

$$W_{ip}^{\min} \le W_i \le W_{ip}^{\max} \quad p = 1, 2, ..., n_F \tag{4}$$

where, $W_{ip}^{\min}$ *and* $W_{ip}^{\max}$ are the lower and upper limit of thermal power generation with fuel option p; a_{1ip}, b_{1ip}, c_{1ip}, E_{1ip}, F_{1ip} represent the coefficients of fuel price of i^{th} generator with fuel option p.

2.2 Constraints of ELD problem

2.2.1 Real Power Constraint or Demand Constraint

The total generation must be equal to transmission loss and system demands. This can be represented as

$$\sum_{i=1}^{N} W_i - (W_D + W_L) = 0 \tag{5}$$

where, W_L, W_D represents the total transmission loss and total system demand respectively. The transmission loss W_L can be calculated as

$$W_L = \sum_{i=1}^{N}\sum_{j=1}^{N} W_i B_{ij} W_j + \sum_{i=1}^{N} B_{0i} W_i + B_{00} \tag{6}$$

2.2.2 Generator Operating Limits Constraint

The power generated by individual generator must vary within it's maximum and minimum limit. Therefore, mathematically this may be written as

$$W_i^{\min} \le W_i \le W_i^{max} \quad i = 1, 2, 3 .... N \tag{7}$$

where, $^{\min}$ and W_i^{max} are the minimum and maximum real power output of the i^{th} generator.

2.2.3 Ramp Rate Limit Constraints

In practical circumstances, the working range of each online unit may be confined by the ramp rate limit[12]. Depending on up (UR_i) and down (DR_i) ramp rate limits, the generation can be increased or decreased.

If generation increases

$$W_i - W_{i0} \le UR_i \tag{8}$$

If generation decreases

$$W_{i0} - W_i \le DR_i \tag{9}$$

where, W_{i0} represents power generation of i^{th} unit at earlier hour.

2.2.4 *Prohibited Zone Constraint*

Each generator might have some zone of operation where operation is limited because of fault in the machines, steam valve operation, boilers, vibration in shafts etc[12]. Thus, a discontinuous cost curve is produced corresponding to prohibited operating zone. Prohibited operating zone may be formulated as

$$\begin{aligned} &W_i^{\min} \le W_i \le W_{i,1}^l \\ &or \\ &W_{i,k-1}^u \le W_i \le W_{i,k}^l \qquad k = 2,3,\ldots\ldots..n_i \\ &or \\ &W_{i,n_i}^u \le W_i \le W_i^{\max} \end{aligned} \tag{10}$$

where, k represents total prohibited zone numbers of i^{th} unit. $W_{i,k}^u$ and $W_{i,k}^l$ represents higher and lower limit of the k^{th} prohibited zone; n_i represents total prohibited zones number of i^{th} unit. During optimization, if $W_{i,k}^u \ge W_i \ge (\mathrm{W}_{i,k}^u + W_{i,k}^l)/2$ then W_i will be fixed to $W_{i,k}^u$. Mathematically this can be expressed as

$$\begin{aligned} &W_i = W_{i,k}^u \\ &\text{if } W_{i,k}^u \ge W_i \ge (\mathrm{W}_{i,k}^u + W_{i,k}^l)/2 \quad k = 2,3,\ldots\ldots..n_i \end{aligned} \tag{11}$$

If $W_{i,k}^l < W_i < (\mathrm{W}_{i,k}^u + W_{i,k}^l)/2$ then W_i will be fixed to $W_{i,k}^l$. Mathematically this can be expressed as

$$\begin{aligned} &W_i = W_{i,k}^l \\ &\text{if } W_{i,k}^l < W_i < (\mathrm{W}_{i,k}^u + W_{i,k}^l)/2 \quad k = 2,3,\ldots\ldots..n_i \end{aligned} \tag{12}$$

2.3 Calculation of Slack Generator Power Output

2.3.1 *Without Transmission Loss*

$$W_N = W_D - \sum_{i=1}^{(N-1)} W_i \tag{13}$$

2.3.2 *With Transmission Loss*

$$W_N = W_D + W_L - \sum_{i=1}^{N-1} W_i \tag{14}$$

Using equation (6) and (14) the modified equation may be written as

$$B_{NN}W_N^2 + W_N\left(2\sum_{i=1}^{N-1} B_{Ni}W_i + B_{ON} - 1\right) + \left(W_D + \sum_{i=1}^{N-1}\sum_{j=1}^{N-1} W_i B_{ij} W_j + \sum_{i=1}^{N-1} B_{0i}W_i - \sum_{i=1}^{N-1} W_i + B_{00}\right) = 0 \tag{15}$$

W_N is the same as mentioned in[31]

3. Grey Wolf Optimizer

This section of the paper presents a newly developed optimization method called grey wolf optimizer, which has been proposed in[36].

Grey wolves usually live in a group of 5-12 members. Alpha is the group leader and their main their main tasks are decisions making, hunting etc. The second member in the hierarchy is Beta and it helps Alpha in order to make the decision. Delta and Omega are the lowest ranking grey wolves. Omega wolves are not an important individual, but entire group facing the fighting problem if they lose any omega. Deltas are responsible for watching the boundary of territory, carrying weak grey wolves, and help the Alpha at the time of hunting.

3.1 Encircling

Hunting behaviour of the grey wolf is started by encircling the prey. For mathematically model of this behaviour following equations have been developed.

$$\vec{D} = \left|\vec{S}\,\vec{G}_P(t) - \vec{G}(t)\right| \tag{16}$$

$$\vec{G}(t+1) = \vec{G}_P(t) - \vec{P}\vec{Q} \tag{17}$$

Here, t, $\vec{P}$ and $\vec{S}$ are the current iteration and coefficients vectors respectively. $\vec{G}_P$ represents prey's position and $\vec{G}$ is the grey wolf's position. $\vec{P}$ and $\vec{S}$ can be formulated as follows

$$\vec{P} = 2\vec{s}_1.\vec{y}\,^{-}\vec{s} \tag{18}$$

$$\vec{S} = 2.\vec{y}_2 \tag{19}$$

Here the value of $\vec{s}$ are decreased in the iterative process from 2 to 0. $\vec{y}_1$ and $\vec{y}_2$ are the random vectors in [0 1].

3.2 Hunting

Alpha wolves are always guiding the other grey wolves at the time of hunting. But in an abstract place, it is very difficult to guess the prey's optimal location. According to the best search agent position, other search agents' positions are updated. This behaviour is mathematically modelled using the following equations

$$\vec{D}_\alpha = \left|\vec{S}_1.\vec{G}_\alpha - \vec{G}\right| \tag{20}$$

$$\vec{D}_\beta = \left|\vec{S}_2.\vec{G}_\beta - \vec{G}\right| \tag{21}$$

$$\vec{D}_\delta = \left|\vec{S}_3.\vec{G}_\delta - \vec{G}\right| \tag{22}$$

$$\vec{G}_1 = \vec{G}_\alpha - \vec{P}_1.(\vec{Q}_\alpha) \tag{23}$$

$$\vec{G}_2 = \vec{G}_\beta - \vec{P}_2.(\vec{Q}_\beta) \tag{24}$$

$$\vec{G}_3 = \vec{G}_\delta - \vec{P}_3.(\vec{Q}_\delta) \tag{25}$$

$$\vec{G}(t+1) = \frac{\vec{G}_1 + \vec{G}_2 + \vec{G}_3}{3} \tag{26}$$

3.3 Exploitation

The hunting process of grey wolves has been stopped after attacking the prey. In order to mathematically model this, the vector a ($\vec{s}$) should be decreased. $\vec{P}$ represents the random number in the interval [-2s,2s] and its value is reduced from 2 to 0. If random value $\vec{P}$ is in position [-1,1], then the next search agent may occupy in any location between the current location of prey and the current location of search agent. When $|\vec{P}|<1$, the grey wolves assault the prey.

3.4 Exploration

Grey Wolves wander from one place to another in order to search a prey and they unite again to assault the prey. To model this behaviour, $\vec{P}$ is used having random values less than -1 or more than 1 or in order to separate the search agents from the prey.

3.5 GWO Algorithm

- Initialize the population matrix according to the number of search agent and dimension of the problem.
- Initialize $\vec{s}$, $\vec{P}$ and $\vec{S}$
- Calculate the fitness value. The position representing best fitness function is denoted by $\vec{G}_\alpha$ and the positions of second and third best fitness function are denoted by $\vec{G}_\beta$ and $\vec{G}_\delta$ respectively.
- Update search agents' position using (20-25).

- Update $\vec{s}, \vec{P}$ and $\vec{S}$
- Calculate the fitness value of all modified sets.
- Update the values of $\vec{G}_\alpha$, $\vec{G}_\beta$ and $\vec{G}_\delta$
- Repeat the steps (iv)-(vii) until the termination criteria are fulfilled otherwise stop the process.

4. Oppositional based Learning

Tizhoosh has proposed Oppositional based learning technique to enhance computational speed and quicken the rate of convergence of various optimization algorithms. For randomly generated population number, it is not possible to guess about the value optimal solution and that is why time required to reach optimal solution is large. In OBL, opposite numbers are introduced along with population numbers. It is found that OBL has the capability to reach optimal solution in minimum time due to introduction of these opposite numbers. Oppositional based learning deals with opposite numbers, quasi-opposite numbers and Quasi-reflected numbers.

If y is the real number between [pq , pr] then the opposite number of y may be defined as

$$y_0 = pq + pr - y \tag{27}$$

Here y_0 is the opposite number of y

If y is the real number between [pq, pr], then the quasi point, y_{q0} can be expressed as

$$y_{q0} = rand(pc, \mathrm{y}_0) \tag{28}$$

Here, pc represents the midpoint of the interval [pq , pr] ; rand (pc, y_0) is the random number which is distributed uniformly between pc and y_0. A Similar principle can be applied for reflecting the quasi-opposite point y_{q0}

If y is the real number between [pc, pr], then the value of y_{qr} can be defined as

$$y_{qr} = rand(pc, \mathrm{y}) \tag{29}$$

Here y_{qr} is the quasi-reflected point.

The above-mentioned definitions can without much of a stretch may be reached out to larger dimensions.

4.1 Implementation of OBL in GWO Algorithm

Oppositional based learning technique is implemented in GWO algorithm in order to accelerating the convergence speed of GWO.QOGWO algorithm starts by choosing GWO variables such as search agent number, maximum iteration number. The population matrix U is generated according to the number of search agent and dimension of the problems. It is necessary to check the various constraints limits. The quasi-reflected sets are generated. After generating quasi-reflected set, the fitness function is calculated for initial population set and quasi-reflected set. On the basis of fitness, the best U sets are sorted out and an updated matrix is formed. After that the minimum value of fitness function is calculated and the corresponding position of search agent is regarded as $\vec{G}_\alpha$. The position representing second and third minimum values of fitness function are considered as $\vec{G}_\beta$ and $\vec{G}_\delta$ respectively. Then $\vec{s}, \vec{P}$ and $\vec{S}$ is calculated using equation (18) and (19). The position of each search agent is updated using equation (20)-(25). Again random number is generated. If the random number is less than jumping rate, quasi reflected sets are generated using the updated set. The constraints limits are checked again. The fitness value for each updated search agent is calculated if all constraints limits are satisfied otherwise quasi-reflected sets are again generated. On the basis of fitness, the best U sets are sorted out between updated set and quasi reflected set. The value of $\vec{G}_\alpha, \vec{G}_\beta, \vec{G}_\delta$ and $\vec{s}, \vec{P}$ and $\vec{S}$ are updated. The process is terminated if maximum number of iteration is reached.

4.2 Algorithms for Quasi Reflected based Initialization

The steps are given below:

Randomly generate initial population U in between maximum and minimum limits of decision variables and in between [0 1] generate a reflection weight μ.

Generate Quasi-Reflected Sets (QRS) for each initially generated population set U, using following procedure

For e=1:A (A= pop set)

For f=1:B (B=decision variable)

If Ue,f < Median

QRSe,f = Ue,f + (Median- Ue,f)*μe %%%% (Median= Ue,f =(pq+pr)/2)

Else

QRSe,f =Median+(Ue,f –Median)*μe

end

end

end

Evaluate fitness value for QRS and total population.

Sort out best A individuals on the basis of their fitness.

Store the best population sets.

4.3 Effect of Jumping Rate

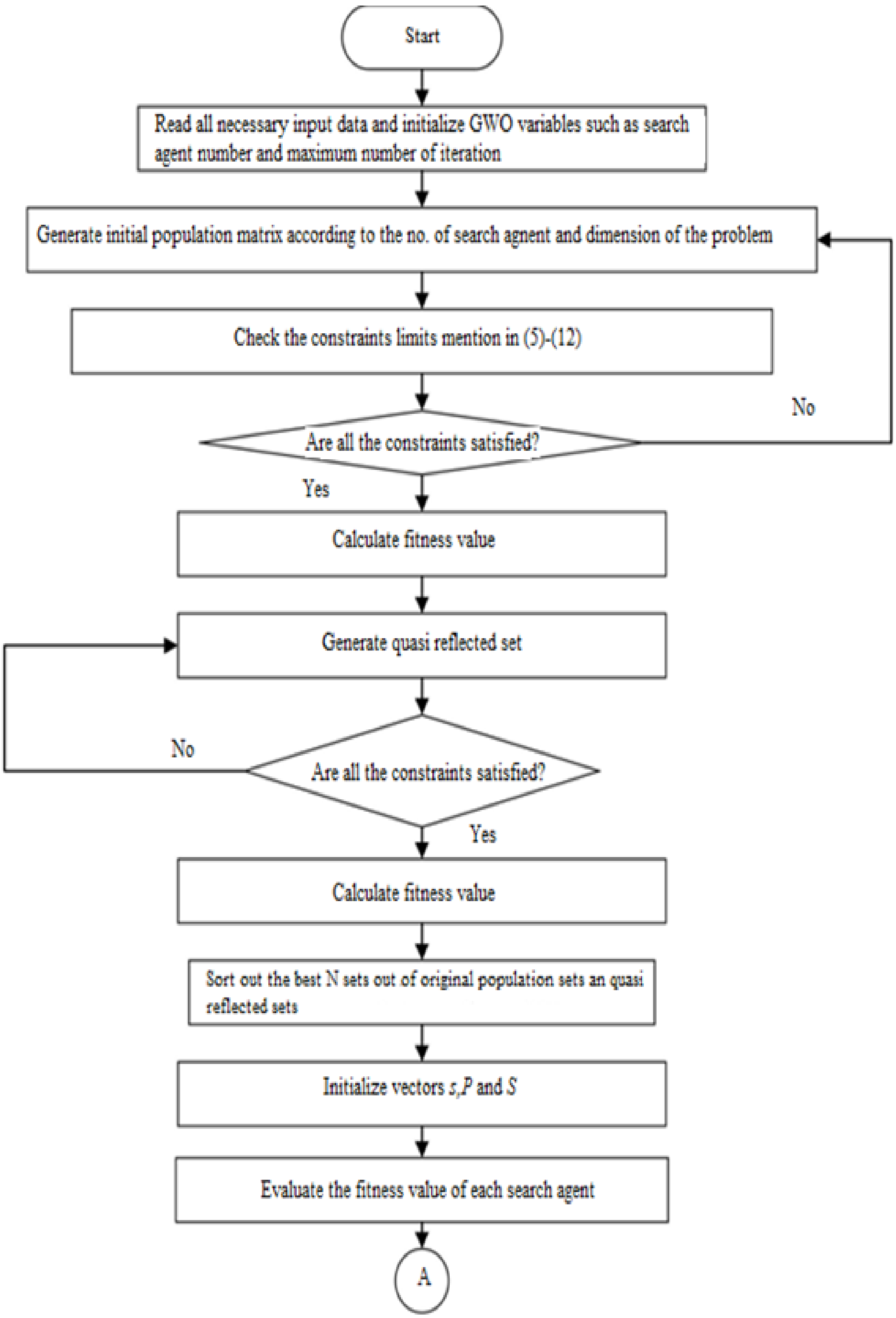
Quasi reflected sets have been used in GWO algorithm for accelerating the convergence speed. But it has been found that if quasi reflected sets are generated in every steps, it may increase the simulation time. Therefore, in order to optimize the computational time, a control parameter called jumping rate[38] has been used. It is a control parameter whose value is set by the user in order to skip the creation of quasi reflected set at certain generation. The effect of this parameter in QOGWO algorithm has been explained in the flow chart of section 4.4.

The pseudo code is given below

1. for e= 1:A (A=population size)
2. if ($rand_e < J_r$)
3. In between [0,1] generate reflection weight μ which is useful for determining the reflection amount of each fitness.
4. For f=1:B (B=desicion variable)
5. If $U_{e,f}$ < Median
6. $QRS_{e,f} = U_{e,f} + (Median- U_{e,f})*\mu_e$ %%%% (Median= $U_{e,f} =(pq+pr)/2$)
7. Else
8. $QRS_{e,f}$ =Median+($U_{e,f}$ –Median)*μ_e
9. end
10. end
11. end
12. Evaluate fitness value for QRS, if selected by J_r
13. End

4.4. Application of Quasi-oppositional Grey Wolf Optimizer Algorithm in ELD

The flow chart of QOGWO algorithm is described in Figure 1 which shows the application of QOGWO algorithm in ELD problems.


Start
Read all necessary input data and initialize GWO variables such as search agent number and maximum number of iteration
Generate initial population matrix according to the no. of search agent and dimension of the problem
Check the constraints limits mention in (5)-(12)
Are all the constraints satisfied?
No
Yes
Calculate fitness value
Generate quasi reflected set
No
Are all the constraints satisfied?
Yes
Calculate fitness value
Sort out the best N sets out of original population sets an quasi reflected sets
Initialize vectors s,P and S
Evaluate the fitness value of each search agent
A

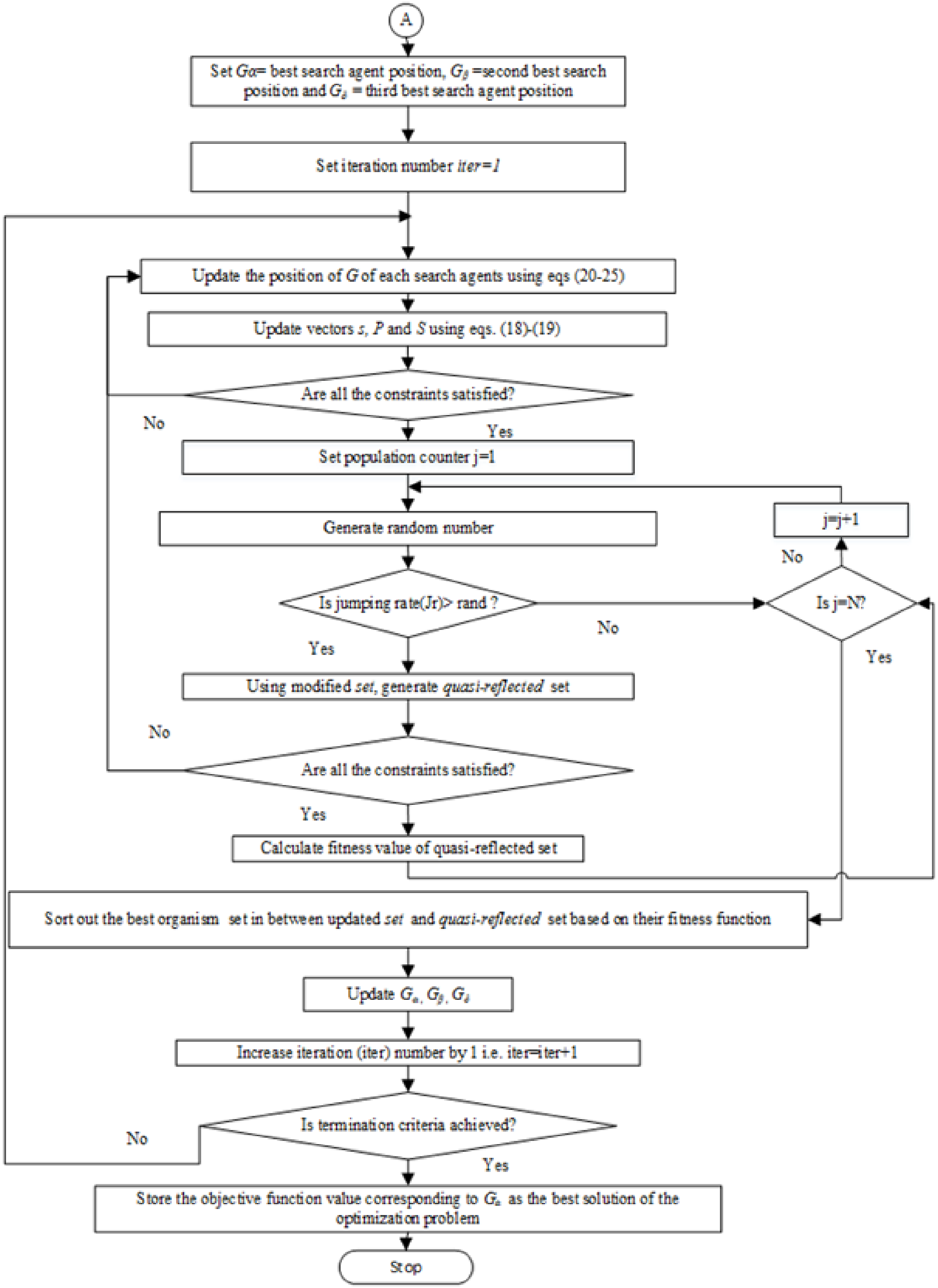


Figure 1. Flow chart of QOGWO algorithm applied to ELD problems.

5. Simulation Results

The QOGWO algorithm has been applied to six different systems of ELD problem and the performance of this method is compared with different soft computing techniques like GA[4], PSO[7], RCCO[11], ORCCRO[12], TLBO[14], OIWO[32], DE/BBO[31], EMA[15], GWO[20] and so on. This algorithm has been coded in Matlab 9 and the program has been executed on a 2.40 GHz core i3 computer with 2 GB RAM.

5.1 Description of Systems

5.1.1 System 1

A 3 generator[20] system having load demand of 850 MW and 1050 MW is considered in this case. The transmission loss has not been considered here. The best results obtained by QGWO, GWO[19], GA[20], PSO[20], ABC[20] and ALO[20] are presented in Table1 and Table 2. The cost convergence characteristics for 3 generators system are

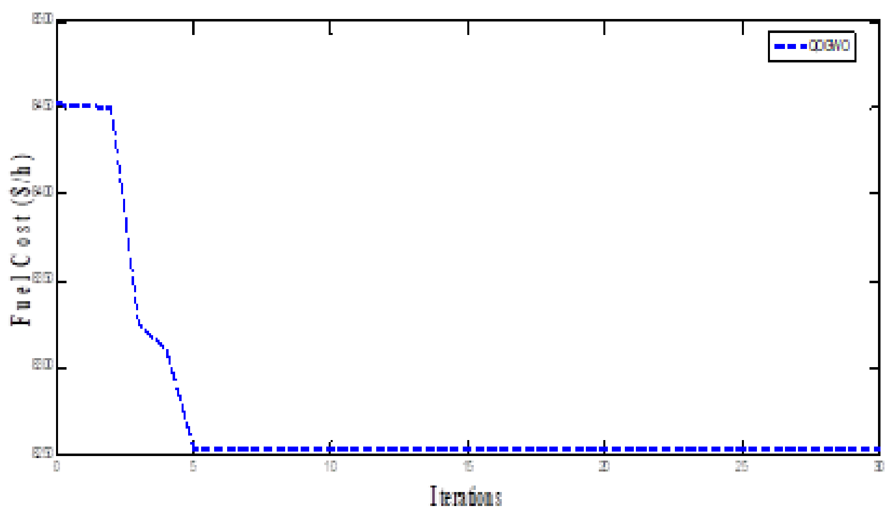


Figure 2. Convergence characteristics of system I (Load demand=850MW).

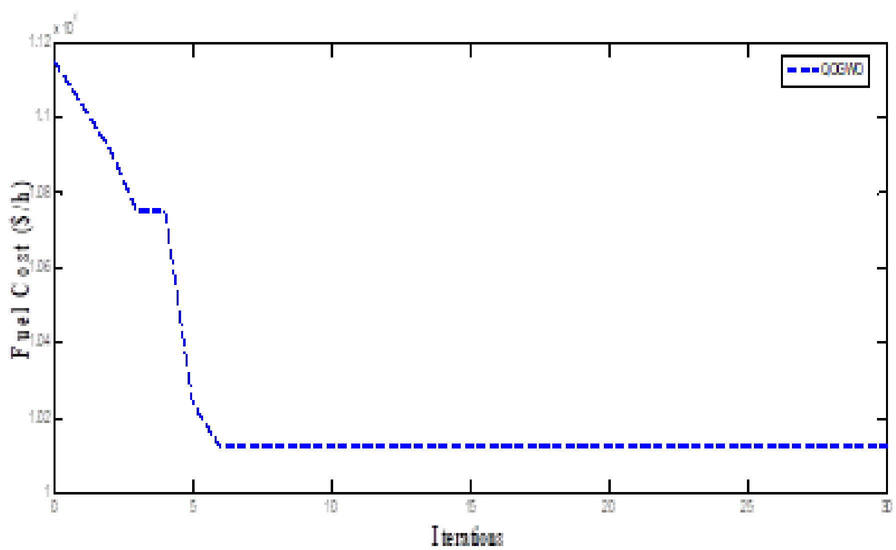


Figure 3. Convergence characteristics of system I (Load demand=1050 MW).

Table 1. Power output for system I against minimum fuel price (Load demand=850 MW)

Units	Lambda iteration[20]	GA[20]	PSO[20]	ABC[20]	GWO[19]	ALO[20]	QOGWO
P_1 MW	382.258	382.2552	394.5243	300.266	300.5116	300.2673	300.2646
P_2 MW	127.419	127.4184	200	149.733	149.8107	149.733	149.7331
P_3 MW	340.323	340.3202	255.4756	400	399.6777	399.9997	400
Fuel Cost ($/h)	8575.68	8575.64	8280.81	8253.1	8253.1053	8253.1052	8253.0629

Table 2. Performance analysis of different methods for system I taken after 50trails (Load demand=850MW)

Methods	Best cost ($/h)	Mean cost($/h)	Worst cost ($/h)	Time/iteration (s)	No of hits to best Solution
QOGWO	8253.0629	8253.0865	8253.3579	12	46
Lambda iteration[20]	NA	NA	NA	NA	NA
GA[20]	NA	NA	NA	NA	NA
PSO[20]	NA	NA	NA	NA	NA
ABC[20]	NA	NA	NA	NA	NA
GWO[19]	8253.1053	8253.10558	8253.1061	NA	NA
ALO[20]	8253.1052	NA	NA	NA	NA

Table 3. Power output for system I against minimum fuel price (Load demand= 1050MW)

Units	Lambda iteration[20]	GA[20]	PSO[20]	ABC[20]	GWO[19]	ALO[20]	QOGWO
P_1 MW	487.5	487.498	492.699	492.6991	492.8465	492.6994	492.6991
P_2 MW	162.5	162.499	157.3	157.301	157.3927	158.1015	158.1015
P_3 MW	400	400	400	400	399.7609	399.1991	399.1993
Fuel Cost ($/h)	10212.459	10212.44	10123.73	10123.73	10123.7196	10123.6949	10123.6931

Table 4. Performance analysis of different methods for system I taken after 50trails (Load=1050MW)

Methods	Best cost ($/h)	Mean cost($/h)	Worst cost ($/h)	Time/iteration (s)	No of hits to best Solution
QOGWO	10123.6931	10123.69325	10123.6931	14	45
Lambda iteration[20]	NA	NA	NA	NA	NA
GA[20]	NA	NA	NA	NA	NA
PSO[20]	NA	NA	NA	NA	NA
ABC[20]	NA	NA	NA	NA	NA
GWO[19]	10123.72	10123.7347	10123.7392	NA	NA
ALO[20]	10123.6949	10123.71481	10123.7347	NA	NA

shown in Figure 2 and Figure 3. Best, worst and average fuel price obtained by QOGWO, Lambda iteration[20], GA[20], PSO[20], ABC[20], GWO[19] and ALO[20] are shown in Table 3 and Table 4.

5.1.2 System II

In this system, a 5 generator system is considered. The Fuel price characteristic is quadratic in nature. The system demand is 730MW. The data required for this case is taken from a paper written by Kamboj et al.[20] The results obtained by QOGWO, Lambda iteration[20], GA[20], PSO[20], APSO[20], EP[20], ABC[20], GWO[20] and ALO[20] are presented in Table 5. Minimum, Worst and Mean value obtained by Lambda iteration[20], GA[20], PSO[20], APSO[20], EP[20], ABC[20], GWO[20] and ALO[20] are displayed in Table 6. Figure 4 shows the fuel price convergence curve.

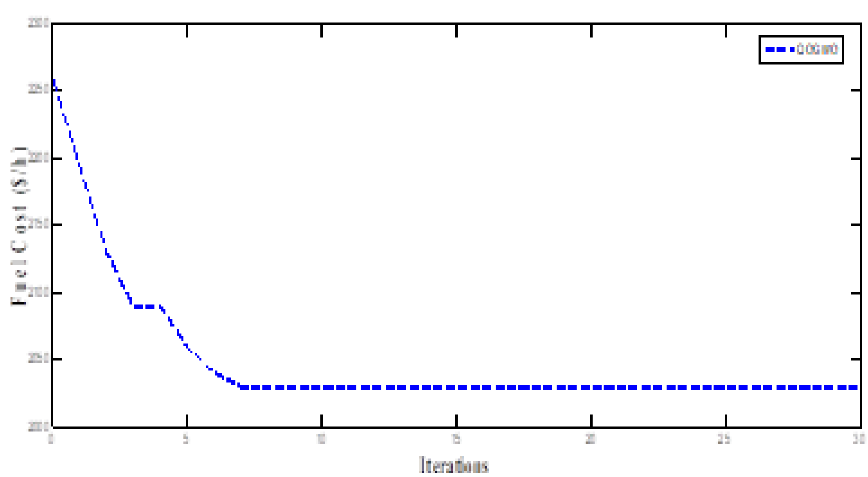


Figure 4. Convergence characteristics of system II.

Table 5. Power output for system II against minimum fuel price (Load demand=730 MW)

Methods	Best cost ($/h)	Mean cost($/h)	Worst cost ($/h)	Time/iteration (s)	No of hits to best Solution
QOGWO	2029.6653	2029.6760	2029.9331	12.2	48
Lambda iteration[20]	NA	NA	NA	NA	NA
GA[20]	NA	NA	NA	NA	NA
PSO[20]	NA	NA	NA	NA	NA
APSO[20]	NA	NA	NA	NA	NA
EP[20]	NA	NA	NA	NA	NA
ABC[20]	NA	NA	NA	NA	NA
GWO[20]	2030.0713	2084.4342	2161.4967	NA	NA
ALO[20]	2029.6669	2055.1717	2089.3825	NA	NA

Table 6. Performance analysis of different methods for system II taken after 50trails (Load demand=730 MW)

Units	GA[33]	PSO[33]	IPSO[33]	SOH-PSO[33]	DE[33]	BFO[33]	CRO[33]	HCRO-DE[33]	GWO	QOGWO
P_1 (MW)	462.0444	447.5823	440.5711	438.2100	448.5181	449.4600	447.9314	447.4021	447.5544	447.3992
P_2 (MW)	189.4456	172.8387	179.8365	172.5800	173.3686	172.8800	173.5548	173.2407	173.1587	173.2409
P_3 (MW)	254.8535	261.3300	261.3798	257.4200	263.0535	263.4100	262.9452	263.3812	263.4457	263.3816
P_4 (MW)	127.4296	138.6812	131.9134	141.0900	138.6733	143.4900	138.8521	138.9774	138.8615	138.9797

Table 6 Continued

P_5 (MW)	151.5388	169.6781	170.9823	179.3700	165.4849	164.9100	165.3046	165.3897	165.4779	165.3918
P_6 (MW)	90.7150	74.8963	90.8241	86.8800	86.3543	81.2520	86.8575	87.0538	86.9499	87.0516
Total Power (MW)	NA	NA	NA	NA	NA	NA	NA	NA	1275.4400	1275.4400
Power loss (MW)	13.0260	13.0066	12.5480	12.5500	12.4527	12.4020	12.4456	12.4449	12.4400	12.4400
Fuel cost ($/h)	15457.960	15450.140	15444.000	15446.0200	15443.105	15443.8497	15443.080	15443.075	15443.0743	15443.0738

5.1.3 System III

A 6 generators system having the system demand of 1263 MW is considered in this case. The prohibited zones, ramp rate limit and transmission loss are also considered here. Necessary data for this system is taken from a paper written by Roy et al.[33] The results obtained by QOGWO, GWO, GA[33], PSO[33], IPSO[33], SOH-PSO[33], DE[33], BFO[33], CRO[33], and HCRO-DE [33] are displayed in Table 7. Figure 5 represents the cost convergence curve for 6-generator system. Minimum, worst and average fuel cost obtained with QOGWO, GWO, HCRO-DE[33], CRO[33], DE[33], BFO[33], SOH-PSO[33], IPSO[33], NAPSO[33], IPSO-TVAC[33], and PSO[33] are shown in Table 8.

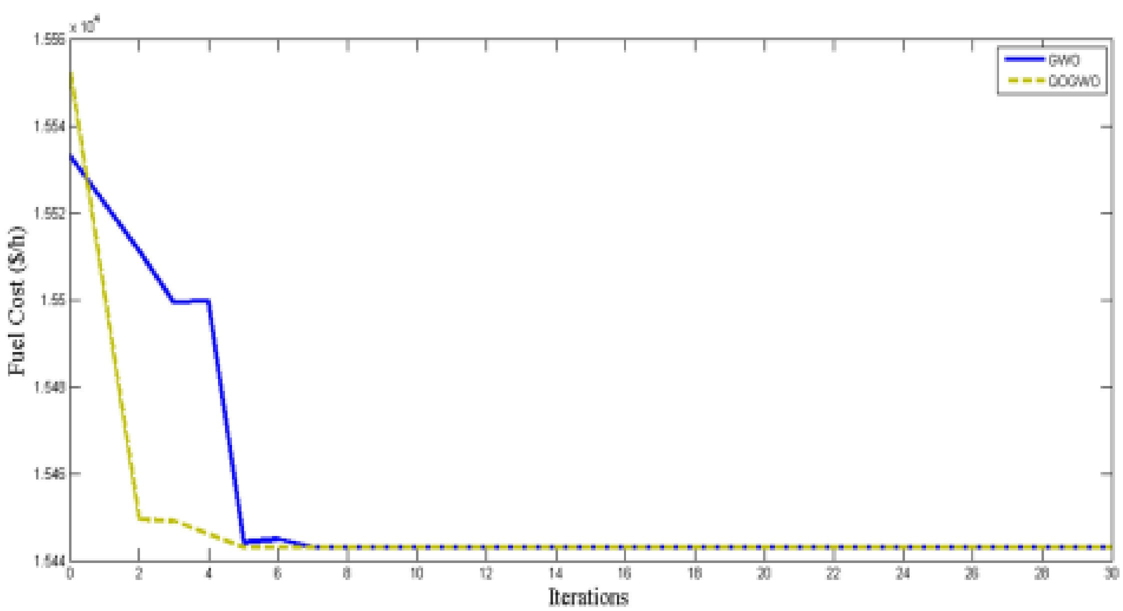


Figure 5. Convergence characteristics of system III.

Table 7. Power output for system III with considering transmission loss (PD=1263 MW)

Units	Lambda iteration[20]	GA[20]	PSO[20]	APSO[20]	EP[20]	ABC[20]	GWO[20]	ALO[20]	QOGWO
P_1 MW	218.028	218.0184	229.5195	225.3845	229.803	229.5247	229.5534	29.5196	229.5196
P_2 MW	109.014	109.0092	125	113.02	101.5736	102.0669	102.3639	102.988	102.9911
P_3 MW	147.535	147.5227	175	109.4146	113.7999	113.4005	113.2209	112.6765	112.6735
P_4 MW	28.38	28.37844	75	73.11176	75	75	74.9183	75	75
P_5 MW	272.042	227.0275	125.4804	209.0692	209.8235	210.0079	209.9434	209.8159	209.8158
Fuel Cost ($/h)	2412.709	2412.538	2252.572	2140.97	2030.673	2030.259	2030.0713	2029.6669	2029.6653

Table 8. Performance analysis of different methods for system III taken after 50 trails (Load demand=1263 MW)

Methods	Generation cost($/h)			Time (s)	Success rate (%)
	Max. Cost ($/h)	Min. ($/h)	Average ($/h)		
QOGWO	15443.0738	15443.0738	15443.0738	3.8	100
GWO	15443.0857	15443.0743	15443.0747	4	96
HCRO-DE[33]	15443.916	15443.075	15443.182	4.17	97
CRO[33]	15446.753	15443.080	15444.135	4.96	82
DE[33]	15448.361	15443.105	15445.287	4.43	76
BFO[33]	NA	15443.8497	15446.9538	NA	NA
SOH-PSO[33]	15609.64	15446.02	15497.35	6.33	NA
IPSO[33]	NA	15444.00	15446.30	NA	NA
NAPSO[33]	15443.7657	15443.7656	15443.7657	NA	NA
IPSO-TVAC[33]	15445.114	15443.063	15443.582	NA	NA
PSO[33]	15492.0	15450.14	15454.00	6	NA

5.1.4 System IV

A 38 generators unit,[13] with quadratic fuel cost function is considered in this case. The system demand is 6000MW. The transmission loss has not been considered for this case. The best result obtained by QOGWO method is shown in Table 9. The average, worst and best fuel price obtained by QOGWO, GWO, BBO[13], DE/BBO[26] and RCCRO[11] is shown in Table 10. Figure 6 presents the convergence characteristic for 38 generators system.

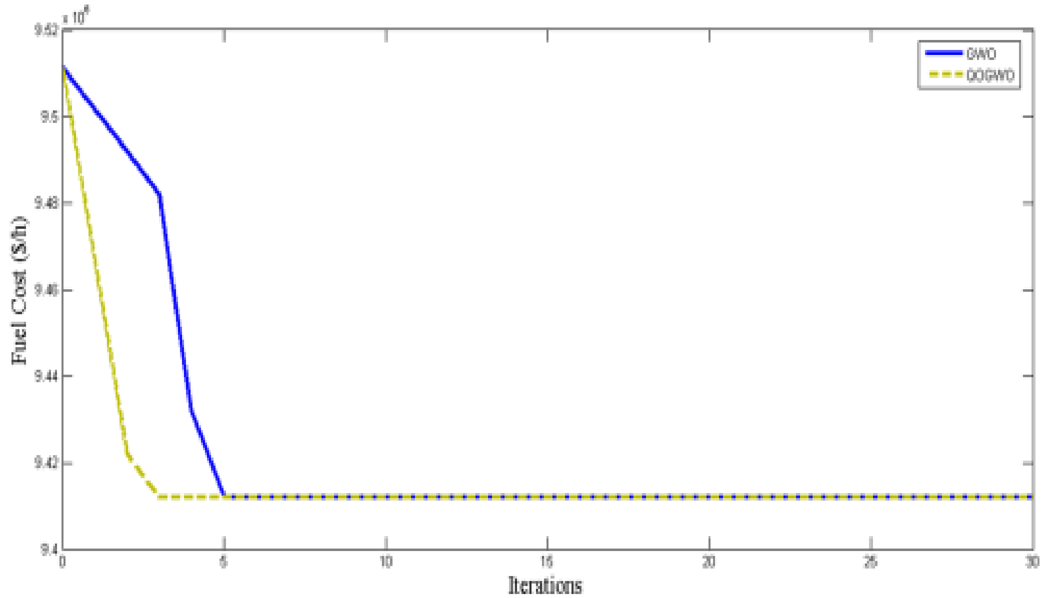


Figure 6. Fuel price convergence characteristics of system IV.

Table 9. Power output for system IV against minimum fuel price (PD=6000 MW)

GWO				QOGWO			
Units	Power outputs (MW)	Units	Power outputs (MW)	Units	Power outputs (MW)	Units	Power outputs (MW)
1	426.283482	21	271.999760	1	426.606400	21	272.000000
2	428.325554	22	259.999546	2	426.714400	22	260.000000
3	428.279991	23	130.237235	3	429.643300	23	130.655800
4	428.837848	24	10.001852	4	429.707600	24	10.001140
5	428.628224	25	113.893154	5	429.631000	25	113.246300
6	429.016029	26	88.381782	6	429.652300	26	88.027270

Table 9 Continued

7	429.160972	27	37.490037	7	429.588400	27	37.486970
8	430.380091	28	20.000012	8	429.749100	28	20.000000
9	114.006472	29	20.000028	9	114.000000	29	20.000000
10	114.000321	30	20.000089	10	114.000000	30	20.000000
11	120.666869	31	20.000000	11	119.796600	31	20.000000
12	127.474273	32	20.000019	12	127.073900	32	20.000000
13	110.000107	33	25.000039	13	110.000000	33	25.000000
14	90.000014	34	18.000010	14	90.000000	34	18.000000
15	82.000055	35	8.000032	15	82.000000	35	8.000002
16	120.000142	36	25.000045	16	120.000000	36	25.000000
17	159.969453	37	21.811972	17	159.574400	37	21.773290
18	65.000006	38	21.154546	18	65.000000	38	21.071580
19	65.000035	**Cost($/h) = 9411940.8691023886**		19	65.000000	**Cost($/h)=9411935.7244**	
20	271.999905			20	272.0000		

Table 10. Performance analysis of different methods for system IV taken after 50trails (Load demand=6000 MW)

Methods	**Generation cost($/h)**			**Time/ iteration, (s)**	**No of hits to minimum Solution**
	Max. cost($/h)	**Min. cost ($/h)**	**Average cost ($/h)**		
QOGWO	9411935.7244	9411935.7244	9411935.7244	0.49	50
GWO	9411940.8691023	9411940.8691023	9411940.8691023	0.56	50
BBO[13]	9417658.7520243911	9417633.6376443729	9417638.15823277617	12.12	41
DE/BBO[31]	9417250.83217432	9417235.786391673	9417237.2909699377	17.5	45
RCCRO[11]	9412404.2774250172	9412404.2774250172	9412404.27 74250172	0.65	50

5.1.5 System V

An 110 unit[32] system with quadratic fuel price has been considered here. The system demand is 15000 MW. The best fuel price obtained by QGWO and GWO techniques is displayed in Table 11. Figure 7 represents the convergence curve of 110 generator system. Average, maximum and best fuel cost acquired by QOGWO, GWO, OIWO[32], ORCCRO[32], SAB[32], SAF[32], BBO[13], DE/BBO[31] and SA[32] for 50 trails are shown in Table 12.

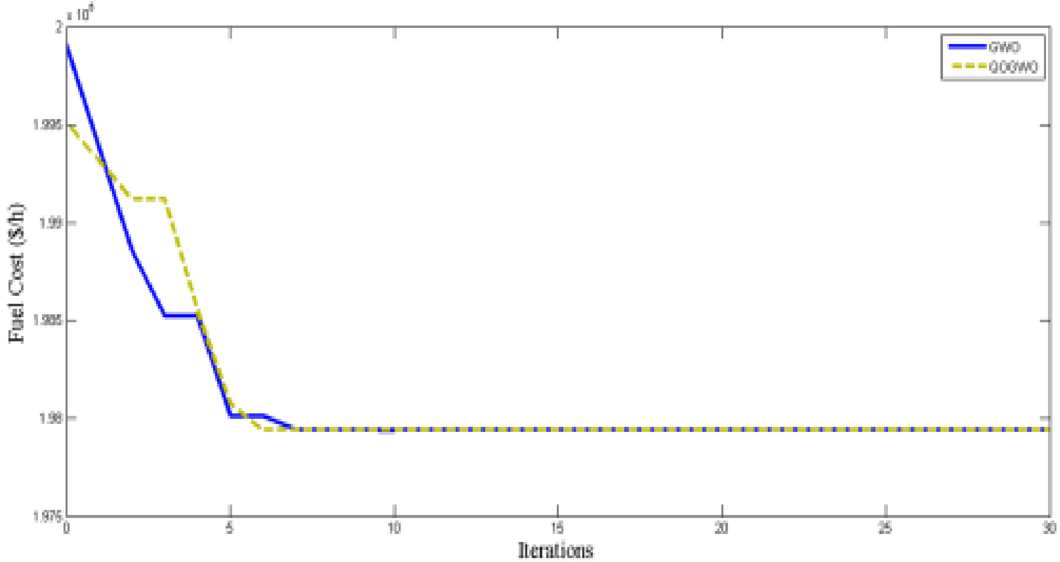


Figure 7. Cost Convergence characteristics for system V.

Table 12. Performance analysis of different methods for system V taken after 50 trails

GWO						QGWO					
Units	Power outputs (MW)	Units	Power outputs (MW)	Units	Power outputs (MW)	Units	Power outputs (MW)	Units	Power outputs (MW)	Units	Power outputs (MW)
1	2.405370	41	146.289200	81	10.001510	1	2.402484	41	154.473600	81	10.000120
2	2.4000000	42	219.999400	82	12.684820	2	2.407023	42	219.980400	82	12.014320
3	2.4000000	43	440.000000	83	20.252710	3	2.400128	43	439.985700	83	20.000590
4	2.404406	44	559.997400	84	199.997400	4	2.400003	44	559.999400	84	199.991600
5	2.400186	45	659.996800	85	325.000000	5	2.400000	45	659.997200	85	324.657300
6	4.002318	46	613.522500	86	439.997800	6	4.000008	46	608.893500	86	439.999800
7	4.002490	47	5.4000560	87	12.468040	7	4.000211	47	5.407616	87	34.991320
8	4.042332	48	5.400046	88	20.044090	8	4.000002	48	5.400769	88	20.046580

Table 11 Continued

9	4.000069	49	8.402872	89	59.389670	9	4.000001	49	8.400001	89	73.331440
10	63.848770	50	8.400017	90	92.689530	10	75.843070	50	8.400000	90	85.298620
11	64.459060	51	8.400010	91	62.302940	11	76.000000	51	8.408124	91	55.543110
12	40.363620	52	12.002930	92	99.946830	12	34.222160	52	12.005930	92	94.715960
13	50.198360	53	12.000000	93	439.997200	13	55.623220	53	12.000020	93	439.999800
14	25.001110	54	12.000000	94	499.943800	14	25.000700	54	12.000030	94	499.959000
15	25.000110	55	12.000020	95	599.999900	15	25.013820	55	12.000940	95	599.962000
16	25.000060	56	25.201300	96	472.196400	16	25.000390	56	25.238840	96	486.265200
17	154.998400	57	25.317330	97	3.603280	17	154.999700	57	25.367110	97	3.600004
18	155.000000	58	35.000000	98	3.600005	18	154.998800	58	35.006200	98	3.600001
19	155.000000	59	35.007430	99	4.401353	19	154.999200	59	35.036350	99	4.400013
20	154.998600	60	45.000000	100	4.400014	20	154.928500	60	45.000420	100	4.400000
21	68.900010	61	45.016410	101	10.092860	21	68.900000	61	45.000920	101	10.000290
22	68.900060	62	45.003140	102	10.000290	22	68.900100	62	45.000010	102	10.028160
23	68.907460	63	184.892100	103	20.000010	23	68.900030	63	184.980800	103	20.003130
24	349.987400	64	184.875900	104	20.000160	24	349.993900	64	185.000000	104	20.000070
25	399.990800	65	184.981900	105	40.001650	25	399.996100	65	184.956800	105	40.000030
26	399.999500	66	184.944100	106	40.000020	26	399.999200	66	184.810900	106	40.000000
27	499.987400	67	70.000730	107	50.000540	27	499.979500	67	70.000470	107	50.000010
28	499.958900	68	70.000040	108	30.015350	28	499.989800	68	70.003380	108	30.018690
29	199.999000	69	70.004050	109	40.000200	29	199.999900	69	70.022430	109	40.000250
30	99.998570	70	359.945600	110	20.027320	30	100.000000	70	359.940300	110	20.000710

Table 11 Continued

31	10.000000	71	399.999200	**Cost($/h) = 197938.7729**	31	10.000510	71	399.999700	**Cost($/h) = 197938.2294**
32	19.799020	72	399.699200		32	19.998330	72	399.999900	
33	79.632870	73	102.031000		33	79.999930	73	100.613000	
34	249.814200	74	194.395900		34	249.998300	74	187.970900	
35	360.000000	75	89.999910		35	359.982800	75	89.986560	
36	399.998200	76	49.989840		36	399.999800	76	49.789200	
37	39.990400	77	160.000600		37	40.000000	77	49.789200	
38	69.999040	78	304.321500		38	69.996980	78	290.147800	
39	99.954810	79	182.017000		39	99.976990	79	170.920500	
40	119.093400	80	118.653400		40	119.968700	80	93.251880	

Table 12. Performance analysis of different methods for system V taken after 50 trails

Methods	**Generation cost($/h)**			**Time (s)**	**No of hits to minimum Solution**
	Max. cost ($/h)	**Min. cost ($/h)**	**Average cost ($/h)**		
QOGWO	197938.2294	197938.2294	197938.2294	21	50
GWO	197938.7729	197938.7729	197938.7729	22	50
OIWO[32]	197989.93	197989.14	197989.14	31	46
ORCCRO[12]	198016.89	198016.29	198016.32	45	48
SAB[32]	NA	206912.9057	207764.73	NA	NA
SAF[32]	NA	207380.5164	207813.37	NA	NA
SA[32]	NA	198352.6413	201595.19	NA	NA
BBO[13]	199102.59	198241.166	198413.45	115	41
DE/BBO[31]	198828.57	198231.06	198326.66	132	43

5.1.6. System VI

In this system, a 160 generators having multiple fuel options is considered. The load demand for this case is 43200 MW. The transmission loss is not considered here. The input data for this system is taken from a paper written by A.K. Barisal and R.C. Prusty[32]. The results of QOGWO and GWO techniques is shown in Table 13. Table 14 describes the minimum, mean and worst cost of fuel obtained by various methods like ORCCRO[12], BBO[13], DE/BBO[31], ED-DE[32], IGA-MU[32], CGA-MU[32] OIWO[32], GWO and QOGWO method. The fuel price convergence curve of this system is shown in Figure 8.

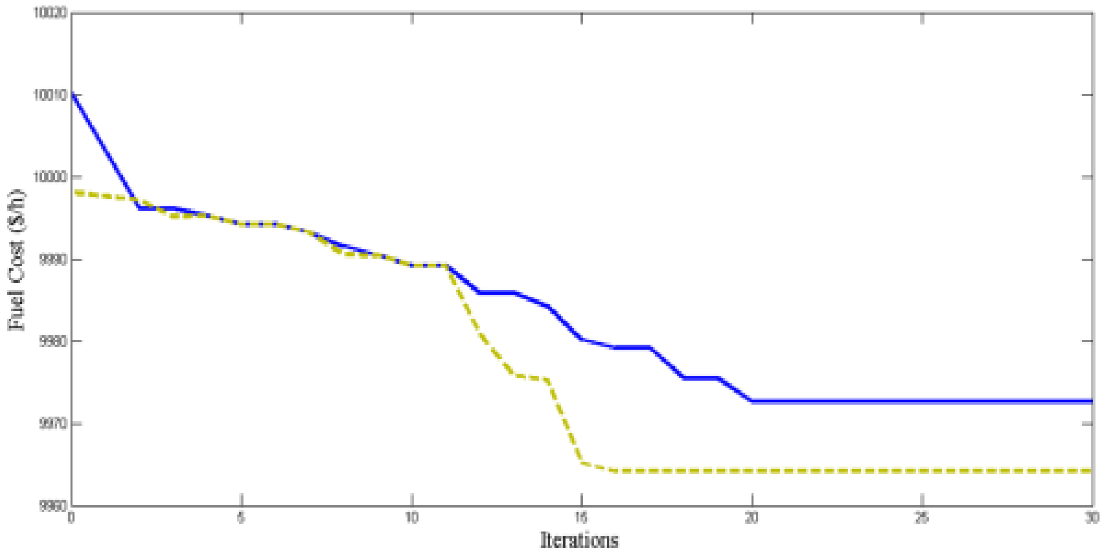


Figure 8. Convergence characteristics of system VI.

Table 13. Performance analysis of different methods for system VI taken after 50trails (PD=43200MW)

GWO						QOGWO					
Units	Power outputs (MW)	Units	Power outputs (MW)	Units	Power outputs (MW)	Units	Power outputs (MW)	Units	Power outputs (MW)	Units	Power outputs (MW)
1	214.6001	55	266.9351	109	431.318	1	214.4864	55	269.2678	109	438.4047
2	204.7634	56	240.0423	110	277.9399	2	212.4548	56	239.9063	110	275.7719
3	281.6329	57	299.4076	111	222.1518	3	274.6084	57	292.4624	111	213.4571
4	238.6973	58	241.38	112	207.5036	4	241.6550	58	239.7737	112	211.9588
5	277.1996	59	437.0827	113	269.1646	5	273.0997	59	437.7344	113	269.6404
6	241.3864	60	279.4758	114	244.8819	6	241.6521	60	279.0115	114	241.7905

Table 13 Continued

7	285.67	61	219.912	115	267.4034	7	287.7118	61	214.4897	115	272.3082
8	235.7428	62	208.014	116	242.597	8	243.9399	62	211.7117	116	238.6985
9	425.9061	63	280.7645	117	289.9475	9	438.9299	63	271.5832	117	290.2308
10	275.8244	64	238.1614	118	240.4427	10	275.8449	64	242.0581	118	238.0277
11	221.6216	65	279.4439	119	437.2013	11	213.4646	65	269.2202	119	439.0474
12	211.8362	66	237.2238	120	291.3842	12	211.2166	66	243.4007	120	275.8780
13	279.7333	67	302.6804	121	217.5635	13	273.5998	67	292.4699	121	215.5350
14	239.7701	68	241.3819	122	213.4731	14	240.9882	68	240.5802	122	210.4741
15	264.3211	69	436.2868	123	275.9653	15	269.1794	69	435.1228	123	271.5557
16	243.5408	70	275.1905	124	237.7568	16	242.5959	70	279.0103	124	240.4459
17	297.1689	71	216.6148	125	276.8827	17	290.0992	71	214.4859	125	272.9668
18	244.7464	72	213.1637	126	240.4433	18	241.7897	72	209.7313	126	240.7150
19	439.1779	73	274.3264	127	297.2385	19	434.4025	73	273.6007	127	287.7276
20	276.0322	74	243.5391	128	239.7791	20	276.9475	74	243.9399	128	242.5956
21	210.6051	75	263.7286	129	427.7141	21	212.3993	75	271.8932	129	436.4365
22	211.6995	76	238.2939	130	277.4549	22	210.9658	76	239.7719	130	275.8959
23	269.6596	77	291.0201	131	216.1465	23	275.6108	77	290.0982	131	213.4583
24	236.8124	78	240.5769	132	212.2094	24	240.4456	78	241.6563	132	211.7168
25	266.2328	79	431.7919	133	278.5976	25	272.6824	79	436.4151	133	275.6235
26	243.4069	80	276.3289	134	240.0441	26	240.5792	80	275.8721	134	239.6419
27	285.7496	81	217.5026	135	259.9762	27	290.0976	81	214.4867	135	269.1072
28	240.1768	82	208.2832	136	239.9135	28	243.6706	82	212.4560	136	240.8468
29	432.3363	83	267.5013	137	292.1943	29	438.3719	83	273.6853	137	292.5700
30	279.5421	84	238.025	138	240.1774	30	278.5974	84	241.1171	138	240.7143

Table 13 Continued

31	223.7816	85	283.6076	139	439.9826	31	214.5197	85	269.1899	139	438.2133
32	214.9326	86	243.1441	140	273.4101	32	210.7252	86	243.9393	140	279.0129
33	278.5622	87	294.4825	141	215.6783	33	272.5917	87	290.2046	141	215.5135
34	246.6263	88	236.4117	142	209.4936	34	243.6708	88	243.1330	142	213.9497
35	281.8277	89	429.1283	143	276.2568	35	272.7212	89	437.9994	143	274.6892
36	238.8329	90	272.8078	144	240.8676	36	241.2527	90	275.8643	144	242.1923
37	294.6553	91	216.7571	145	266.8746	37	292.4739	91	214.4835	145	268.2304
38	240.0565	92	210.7954	146	238.0274	38	240.3105	92	210.4739	146	242.3242
39	426.5568	93	271.799	147	309.7367	39	436.6100	93	273.6028	147	294.8403
40	291.0992	94	239.9063	148	244.35	40	275.8497	94	241.7901	148	239.9077
41	216.745	95	266.9612	149	440	41	214.4845	95	269.1574	149	436.5000
42	215.6718	96	244.4684	150	286.7391	42	210.4742	96	240.5801	150	276.0904
43	272.0938	97	293.2304	151	215.3467	43	276.6243	97	292.4656	151	214.4825
44	240.5761	98	240.9827	152	208.9302	44	241.2520	98	239.1001	152	210.9607
45	262.0954	99	435.1629	153	271.0514	45	269.9264	99	438.7046	153	273.6007
46	244.8809	100	273.3646	154	240.4477	46	241.2519	100	275.9626	154	240.8488
47	295.8804	101	210.9514	155	268.4716	47	292.4482	101	213.4646	155	271.6976
48	241.5271	102	211.7008	156	237.8934	48	242.3284	102	213.9396	156	242.3269
49	420.8979	103	279.7549	157	290.7041	49	435.4686	103	272.5997	157	292.4563
50	273.8595	104	236.4155	158	241.3857	50	278.8265	104	242.7299	158	241.2518
51	210.9682	105	271.3423	159	429.7962	51	215.5313	105	272.7407	159	438.8034
52	204.5304	106	238.1555	160	273.0223	52	212.4543	106	240.4454	160	275.8399
53	281.448	107	292.5963	**Cost($/h) = 9972.5844**		53	273.5997	107	290.0987	**Cost($/h) = 9964.1677**	
54	246.3391	108	238.0247			54	242.8654	108	242.9967		

Table 14. Performance analysis of different methods for system VI taken after 50trails (Load demand=43200MW)

Methods	Generation cost($/h)			Time/iteration, s	No of hits to minimum Solution
	Max. cost ($/h)	Min. cost($/h)	Average ($/h)		
QOGWO	**9964.1961**	**9964.1697**	**9965.4929**	**0.18**	**49**
GWO	**9974.9912**	**9972.5844**	**9972.6806**	**0.2**	**48**
DE/BBO[31]	10010.26	10007.05	10007.56	0.56	42
BBO[13]	10010.59	10008.71	10009.16	0.62	40
CGA-MU[32]	NA	10143.73	NA	NA	NA
ED-DE[32]	NA	10012.68	NA	NA	NA
ORCCRO[32]	1004.45	10004.20	10004.21	0.019	48
IGA-MU[32]	NA	10042.47	NA	NA	NA
OIWO[32]	9983.998	9981.9834	9982.991	0.028	46

5.2 Parameter Tunning

To check the impact of jumping rate on QOGWO algorithm, six systems have been taken and the program is executed for 50 individual trails for each system. The value of jumping rate has been varied from from 0.1-0.9. The results obtained by QOGWO algorithm is shown in

Table 15. Impact of jumping rate on QOGWO after 50 trials

Jumping rate	Min. Cost ($/h)						
	System I		System II	System III	System IV	System V	System VI
	Load demand = 850MW	Load demand = 1050MW					
0.1	8256.5479	10128.3312	2033.7912	15448.1236	9411940.1239	197942.9963	9971.2547
0.2	8254.2403	10126.8796	2031.8569	15444.6548	9411936.9987	1979345.9613	9975.3398
0.3	8254.8879	10126.8837	2035.9987	15444.1435	9411940.7453	197939.1236	9968.6540
0.4	**8253.0629**	**10123.6931**	**2029.6653**	**15443.0738**	**9411935.7244**	**197938.2294**	**9964.1697**
0.5	8253.9873	10125.4789	2031.4712	15447.3689	9411937.6023	197945.0213	9966.2739

Table 15 Continued

0.6	8254.0214	10125.9856	2032.4596	15448.9951	9411939.0213	197941.3982	9969.8802
0.7	8254.2654	10127.5503	2032.8845	15444.5231	9411936.9541	197942.0147	9971.5014
0.8	8254.4506	10126.4478	2034.1236	15445.3698	9411936.2204	197944.0596	9965.9871
0.9	8255.6523	10125.1235	2032.1478	15446.3366	9411937.3943	197942.3987	9968.1182

Table 15. From this table, it is found that when the value of jumping rate is 0.4 then cost obtained by QOGWO algorithm is minimum for all systems. No changes are found when the value of jumping rate is above or below 0.4.

5.3 Comparative Analysis

5.3.1 Robustness and Solution Quality

The best results obtained by QOGWO algorithm are presented in Tables 1, 3, 5,7,9,11 and 13. It is found that QOGWO method gives better results as compared to other well-known optimization techniques. The best, average and worst values for different optimization techniques are presented in Tables 2, 4, 6, 8, 10, 12 and 14. The performance of QOGWO algorithm is judged after running the program for 50 numbers of trials. Out of 50 trails, QOGWO hits the minimum solution 48 times for system I, 47 times for system II, 50 times for system III, 50 times for system IV, 50 times for system V and 49 times for system VI. Therefore, it is found that the success rate of QOGWO is 92%, 90 %, 96 %, 100%, 100%, 100% and 98% respectively. Therefore, from these simulation results, it is seen that QOGWO algorithm shows better performance in terms of robustness and as well as solution quality when compared with GWO and other previously developed optimization techniques.

5.3.2 Computational Efficiency

The QOGWO algorithm takes less time to reach minimum solution as compared to other meta-heuristic optimization techniques. From tables 2, 4, 6, 8, 10, 12 and 14 it is found that, the computational efficiency of QOGWO is better as compared to recently developed optimization techniques. It is also seen that when oppositional based learning is applied in GWO algorithm then the convergence rate becomes faster as compared to other techniques such as GWO, BBO, DE/BBO, RCCRO, EMA and so on.

5.3.3 Statistical Analysis

In recent years, various statistical methods[39,40] has been used for finding out robustness of different algorithms. In this paper, Friedman test and Quade test are chosen to assess the solution quality of QOGWO algorithm as compared to GWO and other recently developed optimization techniques. Table 16 describes the statistical analysis of QOGWO, GWO, DE/BBO, BBO algorithms. Table 16 shows that F-statistic (Chi-Square) value is 9 and Q-statistic value is 15. It is found that F-statistic value is greater than its corresponding critical chi-square value (7.82) and Q-statistic value is also greater than its critical value (4.76). It is also found that p-values obtained by Friedman test and Quade test are less as compared to p-value at 5% significance level. Therefore, there is major dissimilarity between the algorithms. Depending on the average errors evaluated for different cases, the algorithms are ranked and results are shown in Table 16. Therefore, it is clear that rank achieved by QOGWO algorithm is minimum. The average errors of different algorithms are shown in Table 17. Therefore, it may be concluded that in terms of quality solution the QOGWO algorithm gives

Table 16. Positions achieved by Quade and Friedman tests for system II, III, IV. The F-statistic and p- value for Friedman test and Q-statistic value and p-value for Quade test are also shown

Friedman Test					Quade Test				
Systems	**QOGWO**	**GWO**	**DE/BBO**	**BBO**	**Systems**	**QOGWO**	**GWO**	**DE/BBO**	**BBO**
system II	1	2	3	4	system II	-4.5	-1.5	1.5	4.5
system III	1	2	3	4	system III	-3	-1	1	3
system IV	1	2	3	4	system IV	-1.5	-0.5	0.5	1.5
Statistic 9					Statistic 12				
p-value 0.0293					p-value 0.0060				

Table 17. Average errors obtained in system II, system III and system IV

Systems	QOGWO	GWO	DE/BBO	BBO
system II	0	5.1447023	5301.56657	5702.433833
system III	0	0.5435	388.4306	475.2206
system IV	1.3232	8.5109	43.3903	44.9903

better result in a robust manner as compared to other recently developed optimization techniques.

6. Conclusion

In this paper, Quasi Oppositional Grey Wolf Optimizer technique has been implemented successfully for solving various ELD issues. Six different systems having 3, 5, 6, 38, 110 and 160 units have been used here. The simulation results of QOGWO technique have been compared to other optimization technique such as GWO, BBO, DE/BBO, ORCCRO, TLBO, EMA and so on. Simulation results reveal the supremacy of this algorithm in terms of consistency, solution quality and computational efficiency. The robustness of this algorithm is also judged by some statistical analysis like Friedman and Quade test. From these statistical analysis it is clear that in terms of robustness QOGWO also gives better result as compared to other soft computing methods. Therefore, it may be concluded that QOGWO is a strong optimization tool for solving various complex ELD and other non-convex optimization problems.

7. References

1. Fanshel S, Lynes ES. Economic power generation using linear programming. IEEE Transaction on Power Apparatus System. 1964; 83(4):347–56. https://doi.org/10.1109/TPAS.1964.4766011.
2. Wood J, Wollenberg BF. Power generation operation and control. 2nd edition John Wiley and Sons; 2013. p. 1–9.
3. Bard JF. Short-term scheduling of thermal-electric generators using Lagrangian relaxation. Operation Research. 1988; 36(5):756–66. https://doi.org/10.1287/opre.36.5.756.

4. Walters DC, Sheble GB. Genetic algorithm solution of economic dispatch with valve point loadings. IEEE Transactions on Power System. 1993; 8(3):1325–31. https://doi.org/10.1109/59.260861.
5. Panigrahi CK, Chattopadhyay PK, Chakrabarti RN, Basu M. Simulated annealing technique for dynamic economic dispatch. Electronic Power Components System. 2006; 34(5):577–86. https://doi.org/10.1080/15325000500360843.
6. Dieu VN, Peter S, Ongsakul W. Cuckoo search algorithm for non-convex economic dispatch. IET Generation, Transmission and Distribution. 2013; 7(6):645–54. https://doi.org/10.1049/iet-gtd.2012.0142.
7. Gaing ZL. Particle swarm optimization to solving the economic dispatch considering the generator constraints. IEEE Transaction on Power System. 2003; 18(3):1187–95. https://doi.org/10.1109/TPWRS.2003.814889.
8. Dinu CS. A new modified artificial bee colony algorithm for the economic dispatch problem. Energy Conversion and Management. 2015; 89:43–62. https://doi.org/10.1016/j.enconman.2014.09.034.
9. Nomana N, Iba H. Differential evolution for economic load dispatch problems. Electronics Power System and Research. 2008; 78(3):1322–33. https://doi.org/10.1016/j.epsr.2007.11.007.
10. Cai J, Li Q, Li L, Peng H, Yang Y. A fuzzy adaptive chaotic ant swarm optimization for economic dispatch. International Journal of Electronic Power Energy System. 2012; 341):154–60. https://doi.org/10.1016/j.ijepes.2011.09.020.
11. Bhattacharjee K, Bhattacharya A, Dey SH. Chemical reaction optimisation for different economic dispatch problems. IET General Transmission Distribution. 2014; 8(3):530–41. https://doi.org/10.1049/iet-gtd.2013.0122.
12. Bhattacharjee K, Bhattacharya A, Dey SH. Oppositional real coded chemical reaction optimization for different economic dispatch problems. International Journal of Electronic Power Energy System. 2014; 55:378–91. https://doi.org/10.1016/j.ijepes.2013.09.033.
13. Bhattacharya A, Chattopadhyay PK. Biogeography-based optimization for different economic load dispatch problems. IEEE Transmission Power System. 2010; 25(2):1064–77. https://doi.org/10.1109/TPWRS.2009.2034525.
14. Banerjee S, Maity D, Chanda CK. Teaching learning based optimization for economic load dispatch problem considering valve point loading effect. Electronic Power & Energy System. 2015; 73:456–64. https://doi.org/10.1016/j.ijepes.2015.05.036.
15. Ghorbani N, Babaei E. Exchange market algorithm for economic load dispatch. International Journal of Electronic Power Energy System. 2016; 75:19–27. https://doi.org/10.1016/j.ijepes.2015.08.013.
16. Nguyen TT, Vo DN. The application of one rank cuckoo search algorithm for solving economic load dispatch problems. Applied Soft Computer. 2015; 37:763–73. https://doi.org/10.1016/j.asoc.2015.09.010.
17. Kumar S, Naresh R. Nonconvex economic load dispatch using an efficient real coded genetic algorithm. Applied of Soft Computer. 2015; 37:321–9.
18. Mandal B, Roy PK, Mandal S. Economic load dispatch using krill herd algorithm. International Journal Electronic Power Energy System. 2014; 57:1–10. https://doi.org/10.1016/j.ijepes.2013.11.016.
19. Kamboj VK, Bath SK, Dhillon JS. Solution of non-convex economic load dispatch problem using Grey Wolf Optimizer Neural Comput and Application. 2016; 27(5):1301–16. https://doi.org/10.1007/s00521-015-1934-8.
20. Kamboj VK, Bhadoria A, Bath SK. Solution of non-convex economic load dispatch problem for small scale power systems using ant lion optimizer. Neural Computer and Application; 2016. https://doi.org/10.1007/s00521-015-1934-8.
21. Basu M. Kinetic gas molecule optimization for non-convex economic dispatch problem. International Journal of Electronic Power Energy System. 2016; 80:325–32. https://doi.org/10.1016/j.ijepes.2016.02.005.
22. Sinha N, Chakrabarti R, Chattopadhyay PK. Evolutionary programming techniques for economic load dispatch. IEEE Transmission Evolutionary Computer. 2003; 7(1):83–94. https://doi.org/10.1109/TEVC.2002.806788.
23. Coelho LDS, Mariani VC. Combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect. IEEE Transactions on Power System. 2006; 21(2):989–96. https://doi.org/10.1109/TPWRS.2006.873410.
24. Adhinarayanan T, Sydulu MA. directional search genetic algorithm to the economic dispatch problem with prohibited operating zones. Proceedings of IEEE/PES Transmission and Distribution Conference Exposition; 2008. p. 1–5.
25. Selvakumar I, Thanushkodi K. A new particle swarm optimization solution to nonconvex economic dispatch problems. IEEE Transactions on Power System. 2007; 22(1):42–51. https://doi.org/10.1109/TPWRS.2006.889132.
26. Chiou JP. Variable scaling hybrid differential evolution for large-scale economic dispatch problems. Electronic Power System Research. 2007; 77(3–4):212–18. https://doi.org/10.1016/j.epsr.2006.02.013.
27. Panigrahi BK, Pandi VR. Bacterial foraging optimization Nelder-Mead hybrid algorithm for economic load dispatch. IET Generation Transmission Distribution. 2008; 2(4):556–65. https://doi.org/10.1049/iet-gtd:20070422.

28. Chaturvedi KT, Pandit M, Srivastava L. Self-organizing hierarchical particle swarm optimization for nonconvex economic dispatch. IEEE Transmission Power System. 2008; 23(3):1079–87. https://doi.org/10.1109/TPWRS.2008.926455.
29. Vlachogiannis JK, Lee KY. Economic load dispatch – A comparative study on heuristic optimization techniques with an improved coordinated aggregation-based PSO. IEEE Transmission Power System. 2009; 24(2):991–1001. https://doi.org/10.1109/TPWRS.2009.2016524.
30. Park JB, Jeong YW, Shin JR, Lee KY. An improved particle swarm optimization for nonconvex economic dispatch problems. IEEE Transmission Power System. 2010; 25(1):156–66. https://doi.org/10.1109/TPWRS.2009.2030293.
31. Bhattacharya A, Chattopadhyay PK. Hybrid differential evolution with biogeography-based optimization for solution of economic load dispatch. IEEE Transmission Power System. 2010; 25(4):1955–64. https://doi.org/10.1109/TPWRS.2010.2043270.
32. Barisal AK, Prusty RC. Large scale economic dispatch of power systems using oppositional invasive weed optimization. Applied Soft Computer. 2015; 29:122–37. https://doi.org/10.1016/j.asoc.2014.12.014.
33. Roy PK, Bhui S, Paul C. Solution of economic load dispatch using hybrid chemical reaction optimization approach. Applied Soft Computer. 2014; 24:109–25. https://doi.org/10.1016/j.asoc.2014.07.013.
34. Amjady N, Sharifzadeh H. Solution of non-convex economic dispatch problem considering valve loading effect by a new Modified Differential Evolution algorithm. International Journal of Electronic Power Energy System. 2010; 32:893–903. https://doi.org/10.1016/j.ijepes.2010.01.023.
35. Sen T, Mathur HD. A new approach to solve Economic Dispatch problem using A Hybrid ACO-ABC-HS optimization algorithm. International Journal of Electronic Power Energy System. 2016; 78:735–44. https://doi.org/10.1016/j.ijepes.2015.11.121.
36. Mirjalili S, Mirjalili SM, Lewis A. Grey wolf opimizer. Advance Engineering Software. 2014; 69:46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007.
37. Tizhoosh H. Opposition-based learning: A new scheme for machine intelligence. Proceedings of the International Conference on Computational Intelligence for Modelling Control and Automation; 2005. p. 695–701. https://doi.org/10.1109/CIMCA.2005.1631345.
38. Eegezer M, Simon D, Du D. Oppositional biogeography-based optimization. Proceedings of the IEEE International Conference on Systems Man and cybernetics San Antonio TX, USA; 2009. p. 1009–14.
39. Derac J, Garcia S, Molina D, Herrera F. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evolutionary Computer. 2011; 1:3–18. https://doi.org/10.1016/j.swevo.2011.02.002.
40. Shenkin DJ. Hand book of parametric and no parametric statistical procedures. 4th Edition. Chapman & Hall/CRC; 2006.