ERJ
Engineering Research Journal
Faculty of Engineering
Menoufiya University

HYBRID SHUFFLE FROG LEAPING ALGORITHM AND GENETIC ALGORITHM FOR ECONOMIC LOAD DISPATCH WITH WIND POWER UNITS

Ehab E. Elattar

Department of Electrical Engineering, Faculty of Engineering,
Menoufiya University, Shebin El-Kom, Egypt

ABSTRACT

Economic load dispatch (ELD) problem considering valve point effect and wind power is a large scale mixed-integer nonlinear problem. Solving this problem is one of the critical tasks in electric power system operation. So, proposing an accurate method to solve this problem is of great interest. Shuffle frog leaping algorithm (SFLA) is recently proposed to solve many optimization problems but it has some problems. It is often being trapped in local optima. In this paper, a hybrid shuffle frog leaping algorithm and genetic algorithm (HSFLAGA) is proposed to solve this problem. The proposed algorithm can be derived by combining the SFLA and genetic crossover and mutation operators so that the drawbacks of original SFLA algorithm can be treated before employing it to solve the ELD problem with valve point effect and wind power. To show the effectiveness of the HSFLAGA algorithm, several test systems with different numbers of generating units are used. The results of the proposed HSFLAGA algorithm are compared with those obtained by other published methods employing same test systems. The obtained results prove the efficiency and the superiority of HSFLAGA algorithm over other published methods.

ان مشكلة التوزيع الاقتصادى للقدرة الفعالة فى حالة وجود تأثير لنقاط الصمام وطاقة الرياح من المشكلات المعقدة والغير خطية. لذلك فإن حل هذه المشكلة من الامور الحرجة فى تشغيل نظم القوى الكهربية.وبالتالى فإن اقتراح طريقة دقيقة لحل هذه المشكلة من الامور ذات الأهمية الكبيرة. على الرغم من استخدام خوارزمية قفز الضفدع لحل الكثير من المشاكل، لكنها تعانى من عيب الوقوع فى نقاط مثلى محلية. مشكلة التوزيع الاقتصادى للقدرة الفعالة وخاصة فى وجود طاقة الرياح هى مشكلة معقدة لذلك يجب حل عيب خوارزمية قفز الضفدع قبل استخدامها فى حل هذه المشكلة. لحل هذا العيب فى هذا البحث تم اقتراح طريقة هجينة من خوارزمية قفز الضفدع والخوارزمية الجينية . فى الطريقة المقترحة يتم استخدام عامل الانتقال الجينى وعامل الطفرات فى الخوارزمية الجينية لحل العيب الموجود فى خوارزمية قفز الضفدع التقليدية قبل استخدامها لحل مشكلة التوزيع الاقتصادى للقدرة الفعالة وخاصة فى وجود طاقة الرياح. يتم تقييم أداء الخوارزم الهجين المقترح باستخدام بعض الانظمة القياسية و مقارنة النتائج مع بعض الطرق المنشورة. أظهرت النتائج كفاءة الطريقة الهجين المقترحة على الطرق الأخرى المستخدمة فى المقارنة.

Keywords: Wind power penetration, economic load dispatch, valve effect, shuffle frog leaping algorithm and genetic algorithm.

1. INTRODUCTION

The main target of economic load dispatch (ELD) is to minimize the operating cost of the whole system. This can be achieved by determining the power output of each generating unit to meet the constraint condition of the system load demands while satisfying various system and generator constraints [1].

The ELD problem which considered as a large-scale highly constrained nonlinear optimization problem is highly addressed in the literature employing different kinds of constraints, mathematical programming and many optimization techniques [2]–[5]. Various deterministic and heuristic optimization methods have been applied to solve the ELD problem [4], [6]–[9].

In classic ELD problem, the optimization problem is represented by a smooth quadratic or piecewise quadratic objective function. But if the valve point effects are taken into account, the optimization problem becomes even more complicated which results in a non-convex, non-smooth fuel cost function [10], [11].

In literature, there are two different approaches can be used to represent the valve-point effects [12]. In the first approach, these effects can be formulated as inequality constraints which characterize them as prohibited operating zones [13], [14]. While in the second approach a sinusoidal term is considered in the original objective function to model these effects [10], [15]. The second approach is considered in this work.

Recently, the fuel prices and environmental concerns are increased. Therefore, the research on renewable energy applications has been appointed in many countries. This encourages many researchers to make further researches to solve ELD problem incorporating wind power [16]. This leads to modify the fitness function of fuel cost.

Wind power is a promising renewable energy source among the various renewable energy sources. Integrating wind power into power system networks regarded as problematic for power system operation and planning. This is due to the fact that wind power is neither easily predictable nor dispatchable.

Because of the unpredictable nature of wind power, its penetration into power system will lead to some effects such as security concerns. So, a reasonable tradeoff between system risk and operational cost is desired in this case. As a result, further researches are needed to study ELD problem with the integration of wind power [16]. To solve the ELD problem considering wind power, some algorithms such as genetic algorithm (GA) [17], particle swarm optimization (PSO) [18] and bacteria foraging (BF) algorithm [19] are employed in last few years. Hybrid methods which found to be more effective in solving complex optimization problems are also employed in [20] to solve the ELD problem considering wind power.

Recently, Eusuff et.al. [21] introduced memetic metaheuristic algorithm called shuffle frog leaping algorithm (SFLA). The SFLA is based on the behavior of frogs search for the location that has the maximum amount of available food. SFLA is started with an initial population of individuals (frogs) which are generated randomly. Then, these frogs are grouped into memeplexes. Within every memeplex, memetic evolution step (local search) is carried out and a shuffling is completed among the memeplexes. This process is repeated till a stopping criterion is met [22]. This algorithm has been successfully applied for several engineering optimization problems and proves to be a very efficient algorithm [23]–[26].

The conventional SFLA has some advantages. It has fast calculation speed where it can get global optima with a small population size. Also, the generation process of population and updating them is simple. Nevertheless, the conventional SLFA has a problem. It is often being trapped in local optima [27]. Hence, this drawback should be treated first before using it to solve the complex and high dimensioned search space problem.

Because of ELD problem with valve point effect and wind power is a large-scale mixed-integer nonlinear problem; it still cannot be considered a well solved problem for all practical sizes and operating environments. Therefore, in this paper, a hybrid shuffle frog leaping algorithm and genetic algorithm (HSFLAGA) is proposed to solve the ELD problem with valve point effect and wind power. The HSFLAGA algorithm uses the crossover and mutation operators of GA to overcome the problem of conventional SFLA. The proposed algorithm is evaluated using different test systems and compared with some published methods employing the same data. The contributions of this paper are: to propose a hybridized algorithm using crossover and mutation operators of GA and SFLA algorithm, to apply the proposed algorithm to solve the ELD problem with valve point effect and incorporating wind power and to improve the ELD problem solution in comparison with the results obtained with other published methods.

The paper is organized as follows: Section 2 reviews the mathematical formulation of economic load dispatch problem with valve point effect and wind power. Section 3 gives a brief overview of shuffle frog leaping algorithm. The HSFLAGA algorithm is described in Section 4. Simulation results and comparisons with other methods are presented in Section 5. Finally, Section 6 concludes the work.

2. PROBLEM FORMULATION

This paper deals with ELD problem with valve point effect and wind power with equality and inequality constraints. This problem will now be mathematically described as discussed in the literature [1], [10], [12].

2.1 Objective Function

The main idea of ELD problem is to minimize the total fuel cost of generator units while satisfying the total required demand. The total fuel cost of generator units can be mathematically stated as a second order quadratic function as follows [1]:

$$C=\sum_{i=1}^{n}C_i(P_{Gi})=\sum_{i=1}^{n}(a_i+b_iP_{Gi}+c_iP_{Gi}^2)\quad \$/hr \tag{1}$$

where, C is the total fuel cost, C_i is the fuel cost of generator i, a_i; b_i; c_i are the fuel cost coefficients of generator i, P_{Gi}, is the power generated by generator i and n is the number of generators.

By considering valve-point effects, the fuel cost of the generation unit can be defined by adding sinusoidal term as following [1], [10]:

$$C(P_{Gi})=a_i+b_iP_{Gi}+c_iP_{Gi}^2+\left|e_i\times\sin(h_i(P_i^{\min}-P_{Gi}))\right| \tag{2}$$

where, e_i, h_i are the valve-point coefficients of generator i and $P_i^{\min}$ is the minimum capacity limit of generator i.

2.2 Constraints

The economic load dispatch problem is subjects to the following constraints [3], [18]:

1) Power balance constraint: The total power generated must supply the total load demand and the transmission losses.

$$\sum_{i=1}^{n} P_{Gi} - P_D - P_{Loss} = 0 \tag{3}$$

Where, P_D is the total load demand and P_{Loss} is the transmission losses. In this paper, system loss as a function of units' power production is calculated using Kron's loss formula known as B-matrix coefficients [28] as following.

$$P_{Loss} = \sum_{i=1}^{n}\sum_{j=1}^{n} P_{Gi}B_{ij}P_{Gj} + \sum_{i=1}^{n} B_{i0}P_{Gi} + B_{00} \tag{4}$$

where B_{ij} is the loss coefficient relating the productions of units i and j, B_{i0} is the loss coefficient associated with the production of unit i and B_{00} is the loss coefficient parameter.

2) Maximum and minimum limits of power generation:

$$P_{Gi_{\min}} \le P_{Gi} \le P_{Gi_{\max}} \tag{5}$$

where P_{Gimin} and PG_{imax} are the minimum and maximum power generated, respectively.

2.3 Wind Power

1) Power balance including wind power: By considering the wind power (P_w), Eq. 3 can be modified as following [10]:

$$\sum_{i=1}^{n} P_{Gi} + P_w - P_D - P_{Loss} = 0 \tag{6}$$

2) Availability of wind power: The wind power, P_w, in 6 is limited by the available amount from the wind park (P_{av}) [10]:

$$P_{Loss} + P_D - \sum_{i=1}^{n} P_{Gi} \le P_{av} \tag{7}$$

3. SHUFFLED FROG LEAPING ALGORITHM

Shuffled frog leaping algorithm (SFLA) can be considered as a combination of two approaches. The first one is deterministic approach which lets the algorithm to effectively use response surface information to guide the heuristic search, while the second one is random approach which ensures the flexibility and robustness of the search pattern [21]. SFLA can combine the benefits of genetic based memetic algorithm and social behavior based particle swarm optimization algorithm.

The first step of SFLA is the creation of an initial population of P frogs (every frog represents a feasible solution) randomly. For K-dimensional problems (K variables), a frog i can be represented as $X_i = (x_{i1}; x_{i2,\dots,}x_{iK})$. Once a population is created, each frog is evaluated using the fitness function then the frogs are sorted in descending manner according their fitness score. After that, the population is divided into q groups. Each group called memeplex and contains z frogs ($P = q\ x\ z$). The frogs is divided according to the following process: the first frog goes to the first memeplex, the second one goes to the second memeplex, frog q goes to the q^{th} memeplex and the frog $q + 1$ goes back to the first memeplex, and so on [29].

Inside each memeplex, the frog with the best fitness score is known as X_b, while the frog with the worst fitness score is known as X_w. In addition, the frog which has the global best fitness score in all memeplexes is recognized as X_g. Then, the update operation (similar to PSO) is just done only to the frog which has the worst fitness score (X_w) in each cycle. Correspondingly, the location of the frog which has the worst fitness score can be regulated as follows [29]:

$$D_i = Rand() \times (X_b - X_w), \quad (-D_{\max} \le D_i \le D_{\max}) \tag{8}$$

where D_i is the change in i^{th} frog position.

The updated location is given by:

$$X_w(new) = X_w + D_i \tag{9}$$

where Rand( ) is a random number ϵ [0,1] and D_{max} is the maximum allowed change in a frog's position.

If the fitness score of X_w(new) is better than X_w, X_w(new) will replace the worst frog. If it is not improved, a new frog is randomly generated and replaces the worst frog. The calculations then continue for a specific number of iterations [29]. More details of the SFLA can be found in [21], [29], [30].

4. HYBRID SHUFFLE FROG LEAPING ALGORITHM AND GENETIC ALGORITHM (HSFLAGA)

However, the conventional SFLA has been applied to solve several real-world optimization problems, it has a problem. The problem of original SFLA is the possibility of being trapped in the local optima [29], [30]. This problem should be treated first before applying SFLA to solve the non-convex optimization problem. Therefore, to enhance the SFLA and avoid the premature convergence, the HSFLAGA algorithm is proposed in this paper. The proposed algorithm can be derived by combining the SFLA with the crossover and mutation operators of GA.

In the proposed algorithm, GA crossover operator is applied in each memeplex to generate new individual (offspring) which have better fitness to share with other memeplexes. In addition, the crossover operation is employed between the best global frog (the best frog in all memeplexes) and the worst global frog (the worst frog in all memeplexes) to minimize the range of the fitness value. By mutation, individuals are randomly altered by small value called mutation probability. Normally, offspring are mutated after being created by crossover. This leads to prevent premature convergence and loss of genetic diversity. In the proposed algorithm, after shuffling the memeplexes, the genetic mutation operator is employed to modify the population with a small mutation rate p_m.

To employ the proposed algorithm to solve the economic load dispatch problem with valve-point effect and wind power, the main steps of the proposed approach and further procedures especially the modifications based GA crossover and mutation operations are explained here under.

4.1 Representation

The first step is the representation of the frogs. Many solutions to a specific problem are randomly generated initially as memotype (Frog) to form an initial population. Each memotype contains a series of memes similar to genes in GA. In the problem presented in this paper, each meme in a memotype (frog) represents the value of each variable. The length of memotype equals K while the population size equals P (number of frogs). A frog i can be represented as:

$$X_i = (x_{i1}, x_{i2}, ..., x_{iK}), \quad i = 1,2,...,P \qquad (10)$$

4.2 Fitness Function

A fitness function is a function that assigns a quality measure to the memotypes (Frogs). Our objective function which denoted as fitness function is to minimize the total fuel cost considering valve point effect and wind power with respect to equality and inequality constraints.

4.3 GA Crossover Operator

In the proposed algorithm, GA crossover operator is used locally and globally. Locally means that the crossover is used between the best frog memeblex (X_b) and the worst frog (X_w) inside each memeplex to generate offspring which have better ideas to share with other memeplexes.

By applying this operator between X_b and X_w, two new frogs are generated. Then the better two frogs according their fitness score among the four frogs replace the X_b and X_w. The idea behind the crossover operator is to combine useful segments of different parents to form an offspring that benefits from advantageous bit combinations of both parents. Therefore by crossing over X_b and X_w, the two offspring will produce obviously better fitness values than X_w.

In addition, using the crossover operator globally means that the operator is used between the best global frog (X_{bg}) and the worst global frog (X_{wg}). The frogs in all memeplexes are sorted according to their fitness score. Then the best global frog (X_{bg}) and the worst global frog (X_{wg}) are selected. GA crossover operator is executed between X_{bg} and X_{wg} to generate two new offspring with better fitness scores. After that, the two best frogs from the four frogs are chosen. This leads to replace X_{wg} with its child frog which has better fitness value. In this paper, two points crossover is used [31].

4.4 GA Mutation Operator

The crossover step is followed by the mutation step in which some of individuals (Frogs) are randomly changed according to small rate called mutation rate p_m to simulate any sudden change in the environment. This step is important to prevent premature convergence.

Fig. 1 presents the computation procedure of the proposed algorithm where Itloc is the number of local iteration and Itglo is the number of global (algorithm) iteration.

5. SIMULATION RESULTS

In this section, numerical results from different test systems are presented to show the effectiveness of the proposed method. Because of the stochastic nature of proposed method, 30 independent runs were conducted for each system with random initial solution for each run and results (minimum, average, maximum) were calculated. To implement the proposed method, there are some parameters should be tuned for optimal search process. These parameters are population size, number of memeplexes, number of local iteration in each memeplex, number of global (algorithm) iteration, GA crossover rate and GA mutation rate. The best values of these parameters for each system were selected from empirical tests by running the algorithm several times with different parameters combinations. The proposed method is implemented in Pentium 4 personal computer with 2.8 GHz clock frequency and 2 GB of random access memory using MATLAB R2012a.

To seek a fair comparison with other published methods, wind power is considered in all test systems used in this paper where the wind power is assumed to be 10% of the total load demand in each of the test systems.

5.1 Case 1: IEEE 30-Bus System with 6 Generators

To investigate the proposed algorithm performance in comparison with other methods, the HSFLAGA algorithm was tested using IEEE 30-bus system with 6 generators. To seek a fair comparison, the total load demand is 1800 MW. Also, the system losses are considered. The data of this system can be found in [32], [33].

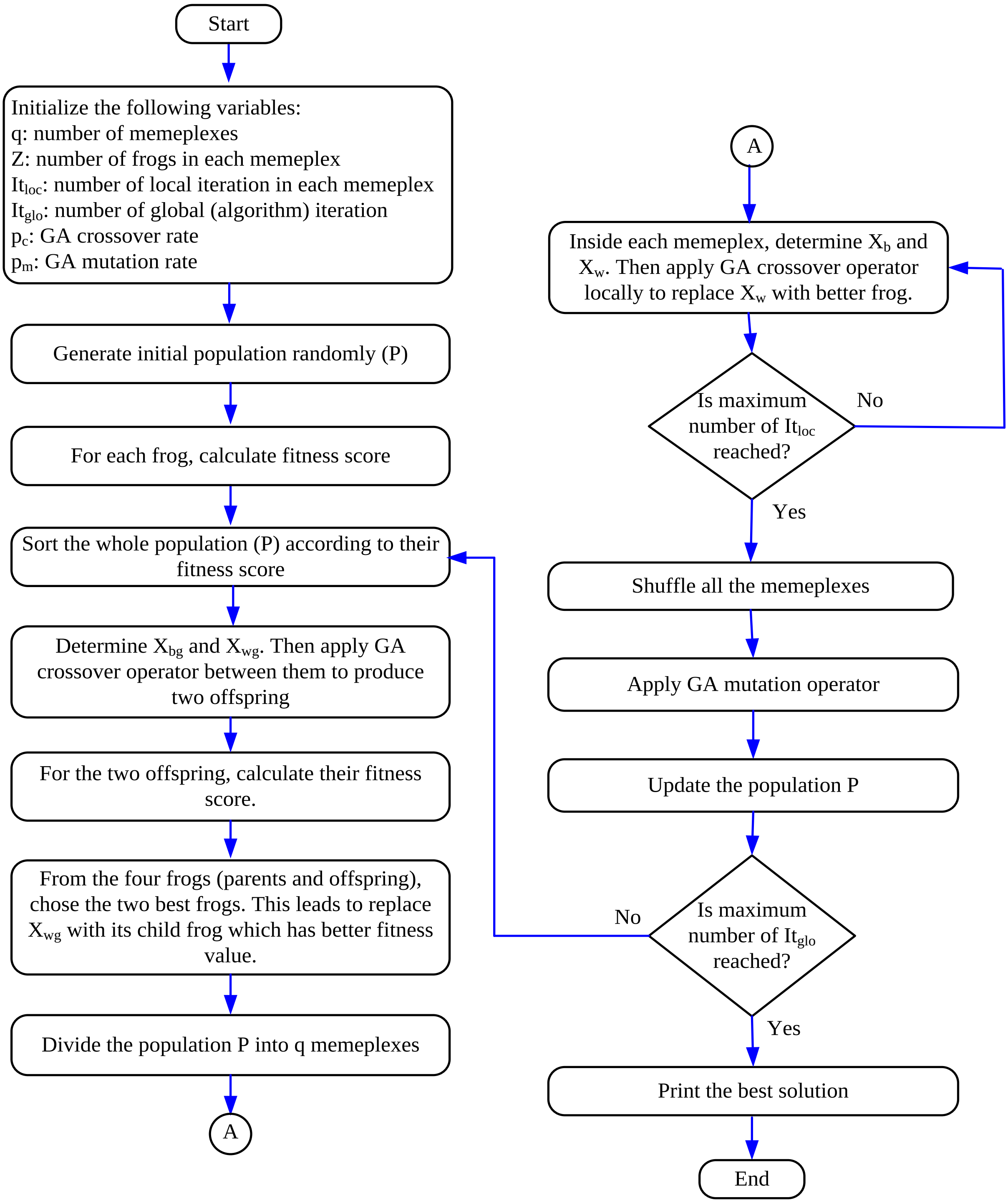
Start
Initialize the following variables:
q: number of memeplexes
Z: number of frogs in each memeplex
It_{loc}: number of local iteration in each memeplex
It_{glo}: number of global (algorithm) iteration
p_c: GA crossover rate
p_m: GA mutation rate
Generate initial population randomly (P)
For each frog, calculate fitness score
Sort the whole population (P) according to their fitness score
Determine X_{bg} and X_{wg}. Then apply GA crossover operator between them to produce two offspring
For the two offspring, calculate their fitness score.
From the four frogs (parents and offspring), chose the two best frogs. This leads to replace X_{wg} with its child frog which has better fitness value.
Divide the population P into q memeplexes
A
A
Inside each memeplex, determine X_b and X_w. Then apply GA crossover operator locally to replace X_w with better frog.
Is maximum number of It_{loc} reached?
No
Yes
Shuffle all the memeplexes
Apply GA mutation operator
Update the population P
Is maximum number of It_{glo} reached?
No
Yes
Print the best solution
End

Fig. 1. Flowchart of the proposed algorithm

The proposed HSFLAGA is applied to this system with and without considering wind power generation. The obtained results without wind power is compared with surrogate worth trade-off with Newton Raphson (SWT-NR) approach [33], the sequential quadratic programming (SQP) method [32], the pattern search (PS) optimization [32] and dynamic adaptive bacterial foraging algorithm (DABFA) [10]. While the obtained results with wind power is compared with DABFA [10]. These results are shown in Tables 1- 3 and Fig 2.

These results show the superiority of the proposed algorithm over other published methods. In addition, it can be noticed that the standard deviation which demonstrates the small variation range of the cost value obtained by the proposed algorithm over 30 runs of a system is very small, which depicts the consistency of the method over different runs with different parameter settings. Moreover, it can be observed that, as expected, the inclusion of wind power led to a decrease of the cost of the system.

5.2 Case 2: 13 Generators System

In this case, the proposed algorithm is tested using system consists of 13 generation units. The data for this system can be found in [7]. To seek a fair comparison, the total load demand is 1800 MW and the losses are neglected while the valve-point effects considered. As in case 1, we have two sub-cases (with and without wind power). The obtained results without wind power is compared with evolutionary programming (EP) [7], particle swarm optimization (PSO) [34], hybrid evolutionary programming with SQP (HEP-SQP) [34], hybrid particle swarm with SQP (HPSO-SQP) [34], chaotic differential evolution with SQP (CDE-SQP) [35] and DABFA [10]. While the obtained results with wind power is compared with DABFA [10]. These results are shown in Tables 4- 5 and Fig. 3

These results show the superiority and effectiveness of the proposed algorithm over other published methods in both sub-cases. The proposed algorithm gives production costs less than those from other methods with or without wind power with small value of standard deviation.

5.3 Case 3: 40 Generators System

In this case, the performance of proposed algorithm is evaluated using system consists of 40 generation units. The data for this system can be found in [7], [36]. To seek a fair comparison, the total load demand is 10500 MW. As in case 2, the valve point effect is considered while the losses are neglected. Also, there are two sub-cases (with and without wind power units). The obtained results without wind power are compared with EP [7], PSO [34], modified PSO (MPSO) method [36], HEP-SQP [34], HPSO-SQP [34], CDE-SQP [35] and DABFA [10]. While the obtained results with wind power is compared with DABFA [10]. These results are shown in Tables 6- 7 and Fig. 4.

Again, the results prove the superiority of the proposed algorithm over other methods in both sub-cases (with and without wind power).

5.3 Discussion

To investigate the effectiveness of the proposed algorithm, its performance is compared with some published methods. The results of these published methods have been directly quoted from their corresponding references. From the above results, one can be notice that the minimum cost obtained from the proposed algorithm is better than other published methods in all cases with or without wind power units. The proposed HSFLAGA significantly outperformed all methods used in this paper. Without wind power, it achieved a minimum cost of 18670.21 \$/hr, 17555.02 \$/hr and 119012.21 \$/hr for cases 1, 2 and 3 respectively which is a yearly saving of about 448000 \$, 2547408 \$ and 7761246 \$ compared to the lowest cost obtained by other methods in cases 1, 2 and 3, respectively. When wind power is considered, it also achieved a minimum cost of 16631.41 \$/hr, 16897.11 \$/hr and 116866.72 \$/hr for cases 1, 2 and 3 respectively which is a yearly saving of about 1945508 \$, 3200115 \$ and 4817737 \$ compared to the cost obtained by DABFA [10] in cases 1, 2 and 3, respectively.

In addition, the difference between minimum and maximum cost of the proposed HSFLAGA algorithm are small, which show the stability of the results obtained by the HSFLAGA algorithm. Moreover, the efficiency of the HSFLAGA algorithm is proved in the standard IEEE test system (case 1) and a large non-convex type problem (case 3). The good performance of proposed HSFLAGA algorithm on the ELD problems with valve-point effect and wind power gives some evidence that the proposed algorithm can be successfully applied to various practical power system optimization problems in the near future.

Table 8 shows the comparison between the proposed algorithm and other methods regarding to the computational burden without using wind power for case 2 and case 3. The results of Tables 1, 2, 4 and 8 prove that the HSFLAGA algorithm has high-speed convergence. Its computational burden is less than other methods in all cases except CDE-SQP [35] and HPSO-SQP [34] in case 2 only. The real life ELD problem is solved off line and solution time of several minutes is acceptable. This makes it possible to use the HSFLAGA algorithm to solve the real life ELD problem.

Table (1), Solutions Obtained From 30 Independent Runs for Each Method without Wind Power for Case 1

Method	Minimum Cost ($/hr)	Mean Cost ($/hr)	Maximum Cost ($/hr)	Standard deviation	Average time (S)
DABFA [10]	18721.40	18721.83	18727.42	1.10	0.56
HSFLAGA	18670.21	18670.99	18675.89	1.02	0.49

Table (2), Solutions Obtained From 30 Independent Runs for Each Method with Wind Power for Case 1

Method	Minimum Cost ($/hr)	Mean Cost ($/hr)	Maximum Cost ($/hr)	Standard deviation	Average time (S)
DABFA [10]	16853.50	16862.60	16866.57	3.50	0.61
HSFLAGA	16631.41	16634.96	16642.64	3.12	0.58

Table (3), Comparison and Results for Case 1.

	Without wind power					With wind power	
	SWT-NR [33]	SQP [32]	PS [32]	DABFA [10]	HSFLAGA	DABFA [10]	HSFLAGA
P_{G1} (MW)	251.70	251.70	252.24	252.31	252.95	269.32	269.51
P_{G2} (MW)	303.79	303.79	306.70	303.32	303.02	270.44	270.32
P_{G3} (MW)	503.48	503.48	505.38	503.09	503.12	435.25	435.29
P_{G4} (MW)	372.32	372.32	365.13	372.74	372.52	329.071	329.23
P_{G5} (MW)	301.47	301.47	302.32	301.33	301.01	261.037	260.56
P_{G6} (MW)	197.40	197.40	198.53	197.32	197.31	158.57	158.55
P_{loss} (MW)	130.15	130.15	130.31	130.12	129.92	103.69	103.46
Cost ($/h)	18721.39	18721.39	18721.5	18721.39	18670.21	16853.4982	16631.41

Table (4), Solutions Obtained From 30 Independent Runs for Each Method without Wind Power for Case 2

Method	Minimum Cost ($/hr)	Mean Cost ($/hr)	Maximum Cost ($/hr)	Standard deviation	Average time (S)
DABFA [10]	17845.82	17865.43	17901.02	13.99	0.64
HSFLAGA	17555.02	17572.31	17603.77	11.86	0.52

Table (5), Solutions Obtained From 30 Independent Runs for Each Method with Wind Power for Case 2

Method	Minimum Cost ($/hr)	Mean Cost ($/hr)	Maximum Cost ($/hr)	Standard deviation	Average time (S)
DABFA [10]	17262.42	17297.41	17361.30	18.65	0.53
HSFLAGA	16897.11	16927.11	16987.09	13.92	0.46

Table (6), Solutions Obtained From 30 Independent Runs for Each Method without Wind Power for Case 3

Method	Minimum Cost ($/hr)	Mean Cost ($/hr)	Maximum Cost ($/hr)	Standard deviation	Average time (S)
DABFA [10]	119898.20	120294.37	121368.13	580.16	6.42
HSFLAGA	119012.21	119508.2	120141.44	115.34	5.02

Table (7), Solutions Obtained From 30 Independent Runs for Each Method with Wind Power for Case 3

Method	Minimum Cost ($/hr)	Mean Cost ($/hr)	Maximum Cost ($/hr)	Standard deviation	Average time (S)
DABFA [10]	117416.67	117658.30	117949.12	135.99	7.81
HSFLAGA	116866.72	117016.61	117263.07	80.01	7.80

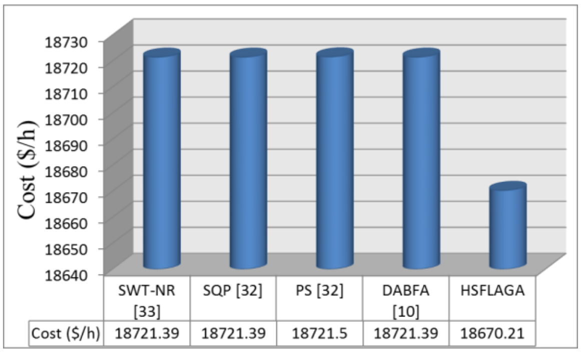

Fig. 2. Comparison and results for Test Case 1 without using wind power

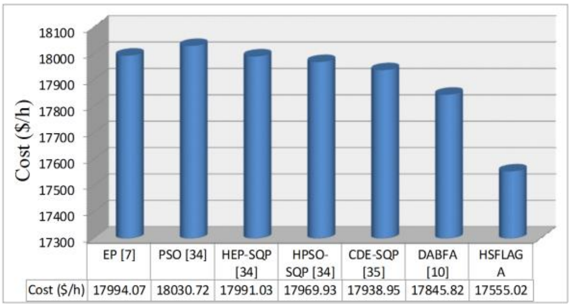

Fig. 3. Comparison and results for Test Case 2 without using wind power

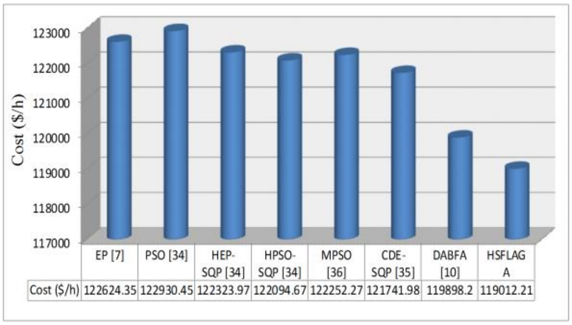

Fig. 3. Comparison and results for Test Case 3 without using wind power

The wind power is included in the formulation of the ELD problem in the power balance equation. Also, the wind power is considered as 10% of the total load demand in all systems used in this paper. From the results in Tables 2, 3, 5 and 6, we can notice that the inclusion of wind power decreases of the cost of the system. This is because the wind power does not consume any fuel.

6. CONCLUSION

In this paper, an optimization method called HSFLAGA was employed to solve the ELD problem considering valve point effect and wind power units. The proposed algorithm can be derived by combining SFLA with the genetic algorithm crossover and mutation operators. This leads to overcome the drawbacks of original SFLA. The feasibility and efficiency of the proposed algorithm have been demonstrated using the commonly used test systems. The numerical results have been compared with the recently reported approaches. These results revealed that the solution of ELD problem (with or without wind power) obtained by the HSFLAGA algorithm led to a smaller operating cost than those found by other methods. This proves the capability of the algorithm to determine the global or near global solutions. In addition, the results show the economic benefits of integrating wind power into power system grids.

7. REFERENCES

[1] J. Zhu, “Optimization of power system operation”, John Wiley inc; 2009.

[2] Y. Xu, W. Zhang and W. Liu, “Distributed dynamic programming-based approach for economic dispatch in smart grids”, IEEE Transactions on Industrial Informatics, Vol. 11, No. 1, pp. 166–175, 2015.

[3] Z. Li Z, W. Wu, B. Zhang and B. Wang, “Decentralized multi-area dynamic economic dispatch using modified generalized benders decomposition”, IEEE Transactions on Power Systems, Vol. pp, No. 99, 2015 (appeared online with DOI 10.1109/TPWRS.2015.2399474).

[4] Z. Li, W. Wu, B. Zhang, H. Sun and Q. Guo, “Dynamic economic dispatch using Lagrangian relaxation with multiplier updates based on a quasi-newton method”, IEEE Transactions on Power Systems, Vol. 28, No. 4, pp. 4516–4527, 2013.

[5] E. Elanchezhian, S. Subramanian and S. Ganesan, “Economic power dispatch with cubic cost models using teaching learning algorithm”, IET Generation, Transmission and Distribution, Vol 8, No. 7, pp. 1187–1202, 2014.

[6] F. Benhamida, I. Ziane, S. Souag, A. Graa and B. Dehiba, “Solving dynamic economic load dispatch with ramp rate limit using quadratic programming”, In: North American Power Symposium (NAPS), Manhattan, KS; p. 1–5, 2013.

[7] N Sinha, R. Chakrabarti and P. Chattopadhyay, “Evolutionary programming techniques for economic load dispatch”, IEEE Transactions on Evolutionary Computation, Vol. 7, No. 1, pp. 83–94, 2003.

[8] S. Agrawal, T. Bakshi and D. Majumdar, “Economic load dispatch of generating units with multiple fuel options using PSO”, International Journal of Control and Automation, Vol. 5, No. 4, pp. 79–92, 2012.

[9] B. Goswami and V. Agrawal, “Security constrained economic dispatch using a hybrid fuzzy genetic algorithm approach”, In: International Conference on Computer Communication and Informatics (ICCCI); p. 1–6, 2013.

[10] I. Farhat and M. El-Hawary, “Dynamic adaptive bacterial foraging algorithm for optimum economic dispatch with valve-point effects and wind power”, IET Generation, Transmission and Distribution, Vol. 4, No. 9, pp. 989–999, 2010.

[11] D. Vo, P. Schegner and W. Ongsakul, “Cuckoo search algorithm for non-convex economic dispatch”, IET Generation, Transmission and Distribution, Vol. 7, No. 6, pp. 645–654, 2013.

[12] T. Victoire and A. Jeyakumar, “Reserve constrained dynamic dispatch of units with valve-point effects”, IEEE Transactions on Power Systems, Vol. 20, No. 3, pp. 1273–1282, 2005.

[13] F. Lee and A. Breipohl, “Reserve constrained economic dispatch with prohibited operating zones”, IEEE Transactions on Power Systems, Vol. 8, No. 1, pp. 246–254, 1993.

[14] Z. Gaing, “Particle swarm optimization to solving the economic dispatch considering the generator constraints”, IEEE Transactions on Power Systems, Vol. 18, No. 3, pp. 1187–1195, 2003.

[15] P. Attaviriyanupap, H. Kita, E. Tanaka and J. Hasegawa, “A hybrid EP and SQP for dynamic economic dispatch with nonsmooth fuel cost function”, IEEE Transactions on Power Systems, Vol. 22, No. 2, pp. 411–416, 2002.

[16] V. Miranda and P. S. Hang, “Economic dispatch model with fuzzy wind constraints and attitudes of dispatchers”, IEEE Transactions on Power Systems, Vol. 20, No. 4, pp. 2143–2145, 2005.

[17] L. Yong and S. Tao, “Economic dispatch of power system incorporating wind power plant”, In: International Power Engineering Conference, (IPEC 2007), pp. 159–162, 2007.

[18] M. Dozein, J. Ansari and M. Kalantar, “Economic dispatch incorporating wind power plant using particle swarm optimization”, In: Second Iranian Conference on Renewable Energy and Distributed Generation, pp. 178–182, 2012.

[19] M. Tripathy and A. Barisal, “Bacteria foraging algorithm based economic load dispatch with wind energy”, In: World Congress on Nature and Biologically Inspired Computing, (NaBIC 2009), pp. 1327–1332, 2009.

[20] C. Peng, H. Sun, J. Guo and G. Liu, “Dynamic economic dispatch for wind-thermal power system using a novel bi-population chaotic differential evolution algorithm”, International Journal of Electrical Power and Energy Systems, Vol. 42, No. 4, pp. 119–126, 2012.

[21] M. Eusuff, K. Lansey and F. Pasha, “Shuffled Frog leaping: A memetic meta-heuristic for discrete optimization”, Engineering Optimization, Vol. 38, No. 2, pp. 129–154, 2006.

[22] J. Anita and I. Raglend, “Solution of unit commitment problem using shuffled frog leaping algorithm”, In: International Conference on Computing, Electronics and Electrical Technologies [ICCEET], pp. 109–115, 2012.

[23] M. Barati and M. Farsangi, “Solving unit commitment problem by a binary shuffled frog leaping algorithm”, IET Generation, Transmission and Distribution, Vol. 8, No. 9, pp. 1050–1060, 2014.

[24] X. Cui, X. Shi, H. Li and C. Lee, “A shuffle frog-leaping algorithm for test scheduling of 2D/3D SoC”, In: IEEE 11th International Conference on Solid-State and Integrated Circuit Technology (ICSICT), pp. 1–3, 2012.

[25] H. Hasanien, “Shuffled frog leaping algorithm for photovoltaic model identification”, IEEE Transactions on Sustainable Energy, Vol. 6, No. 2, pp. 509-515, 2015.

[26] I. Sardou, I. Shahrood, M. Banejad, R. Hooshmand and A. Dastfan, “Modified shuffled frog leaping algorithm for optimal switch placement in distribution automation system using a multi-objective fuzzy approach”, IET Generation, Transmission and Distribution, Vol. 6, No. 6, pp. 493–502, 2012.

[27] T. Niknam, M. Narimani, M. Jabbari and A. Malekpour, “Modified shuffle frog leaping algorithm for multi-objective optimal power flow, “Energy, pp. 36, pp. 6420–6432, 2015.

[28] A. Abdelaziz, M. Kamh, S. Mekhamer and M. A. Badr, “A hybrid HNN-QP approach for dynamic economic dispatch problem”, Elect. Power Syst. Res. Vol. 78, No. 10, pp. 1784–1788, 2008.

[29] K. Bhattacharjee and S. Sarmah, “A modified shuffled frog leaping algorithm with genetic mutation for combinatorial optimization”, Lecture Notes in Computer Science, Vol. 7654, pp. 513–522, 2012.

[30] P. Roy, P. Roy and A. Chakrabartic, “Modified shuffled frog leaping algorithm with genetic algorithm crossover for solving economic load dispatch problem with valve-point effect”, Applied Soft Computing, Vol. 13, pp. 4244–4252, 2013.

[31] Z. Michalewicz. Genetic algorithms + data structures = Evolving programs. Springer-Verlag, Berlin Heidelberg; 1996.

[32] M. Alhajri and M. EL-Hawary, “Pattern search optimization applied to convex and non-convex economic dispatch”, In: IEEE Int. Conf. on Systems, Man and Cybernetics (ISIC2007), pp. 2674–2678, 2007.

[33] J. Dhillon and D. Kothari, “The surrogate worth trade-off approach for multiobjective thermal power dispatch problem”, Elect. Power Syst. Res., Vol. 56, No. 2, pp. 103–110, 2000.

[34] T. Victoire and A. Jeyakumar, “Hybrid PSOSQP for economic dispatch with valve-point effect”, Elec. Power Syst. Res., Vol. 71, No. 1, pp. 51–59, 2004.

[35] L. Coelho and V. Mariani, “Combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect”, IEEE Trans. Power Syst., Vol. 21, No. 2, pp. 989–996, 2006.

[36] J. Park, K. Lee, J. Shin and K. Lee, “A particle swarm optimization for economic dispatch with nonsmooth cost functions”, IEEE Trans. Power Syst., Vol. 20, No. 1, pp. 34–42, 2005.