

HOSTED BY



ELSEVIER

Contents lists available at ScienceDirect

Engineering Science and Technology, an International Journal

journal homepage: www.elsevier.com/locate/jestch

Full Length Article

Modified social spider algorithm for solving the economic dispatch problem

W.T. Elsayed^{a,*}, Y.G. Hegazy^b, F.M. Bendary^a, M.S. El-bages^a^aElectrical Engineering Department, Faculty of Engineering at Shoubra, Benha University, Cairo, Egypt^bFaculty of Information Engineering and Technology, German University in Cairo (GUC), Cairo, Egypt

ARTICLE INFO

Article history:

Received 7 June 2016

Revised 1 September 2016

Accepted 1 September 2016

Available online 16 September 2016

Keywords:

Economic load dispatch

Metaheuristic techniques

Non-convex optimization

Social spider algorithm

Transmission losses

ABSTRACT

The economic load dispatch problem is a non-convex and non-linear optimization problem due to the inclusion of practical features such as valve point effects, prohibited operating zones, ramp rate limits, and transmission losses. For solving the non-convex economic load dispatch problem, a social spider algorithm has been proposed recently. This paper proposes a modified version of the social spider algorithm and studies the application of this version for solving the non-convex economic load dispatch problem. The proposed modification significantly improves the performance of the social spider algorithm. Four benchmark test systems having 6 units, 40 units, 80 units, and 140 units are considered to demonstrate the efficacy of the proposed algorithm. The results obtained from the modified social spider algorithm surpass the results obtained by the original social spider algorithm and significantly compete with the best results presented in previous literature.

© 2016 Karabuk University. Publishing services by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

The worldwide extensive and continuous energy consumption leads to a reduction in the fossil fuel resources. As a result, optimal exploitation of the available fossil fuel resources is a critical issue. Distributing the power system load to the generation units in the most economic manner implies the best exploitation of the fossil fuel resources in the power generation process. Accomplishing the previous task while satisfying the system constraints is known as the economic load dispatch (ELD) problem. There are several variants of the ELD problem. These variants can be distinguished based on aspects considered in the mathematical formulation of the problem. Considering the emission of pollutants from fossil fuel based generation leads to a combined economic emission dispatch problem which is a multi-objective optimization problem [1]. Dynamic models are required for solving the dynamic ELD problem over a time horizon of 24 h with the consideration of ramp rate limits [2]. Stochastic or probabilistic formulation is necessary for studying the incorporation of renewable energy resources into the ELD problem [3]. Increasing the energy conversion efficiency can be achieved using a combined heat and power economic dispatch. Formulating and solving the combined heat

and power economic dispatch problem has been discussed recently in references such as [4,5]. Considering practical features of the generation units such as valve point effects, prohibited operating zones, ramp limits, and multiple fuel options entails a non-convex formulation of the ELD problem [6,7]. Traditionally, quadratic cost function is used to model the fuel consumption cost of generation units; however, some steam generators have a rectified sinusoid imposed on the quadratic cost function due to steam valves opening processes [6]. On the other hand, due to physical limitations in some power plant components, generation units may fail to operate within certain power ranges. This issue creates disjoint prohibited operating zones. The previous practical issues render the solution space of the ELD problem non-convex and non-smooth. As a result, the practical ELD problem is considered as an NP-hard optimization problem [7].

Several methods have been proposed for solving the economic dispatch problem. These methods can be divided into conventional optimization methods and metaheuristic based optimization methods. The conventional methods such as linear programming, interior point method, and lambda iteration method work effectively to solve the ELD problem with convex cost functions only. They may fail in finding the global optimal solution of the non-convex ELD problem. On the other hand, the metaheuristic techniques are capable to hit the global optimal solution of the non-convex ELD problems. Based on that, in the previous two decades, many metaheuristic techniques have been proposed for

* Corresponding author.

E-mail address: wael.alsayed@feng.bu.edu.eg (W.T. Elsayed).

Peer review under responsibility of Karabuk University.

solving the non-convex ELD problem such as genetic algorithm [8], particle swarm optimization [9], differential evolution [10], seeker optimization algorithm [11], differential harmony search algorithm [12], artificial bee colony optimization [13], group search optimizer [14], immune algorithm [15], backtracking search algorithm [16], and hybrid harmony search with arithmetic crossover operation [17]. The genetic algorithms require large computational time. The hybrid harmony search with arithmetic crossover operation, the immune algorithm, and the differential evolution suffer from premature convergence. Artificial bee colony has a poor exploitation capability. The remaining of these algorithms suffer with the trapping in local optimal solutions. Another limitation exists in many of the above algorithms is the need for tuning several parameters based on the considered optimization problem. The previous limitations require developing more efficient techniques for solving the non-convex ELD problem.

In general, as the research in metaheuristic techniques progresses, more efficient and more capable metaheuristic techniques to find the global optimal solution are developed. Recently, many new metaheuristic techniques have achieved remarkable success in solving complex and non-convex optimization problems in different research fields. These new promising metaheuristic techniques are bat algorithm [18–20], firefly algorithm [21–24], flower pollination algorithm [25,26], teaching learning based optimization algorithm [27,28], symbiotic organisms search algorithm [29], real coded genetic algorithm [30], non-dominated sorting hybrid cuckoo search algorithm [31], hybrid harmony-gravitational search algorithm [32], genetically encoded mutable smart bee algorithm [33], and social spider algorithm (SSA) [34]. Among these promising techniques, the SSA is the most promising one. The SSA mimics the foraging behavior of certain type of spiders, which work in groups and known as social spiders. There are two variants of the SSA proposed recently; one is proposed in [35], and the second is proposed in [36]. The latter is considered in this paper for modification and improvement. Applying the SSA for solving the non-convex economic dispatch problem has been studied in [34]. In [34], the SSA is applied in its original form with a simple modification for solving the non-convex economic dispatch problem. This simple modification is to perform a random walk with chaotic sequence based memory factor. The results presented in [34] show that the SSA is capable to provide superior results compared to many previously proposed metaheuristic techniques. In this paper, the SSA is modified through replacing the binary mask based random walk, which exists in the original SSA with a new mutation process followed by a selection process. The proposed modification utilizes the natural evolution concepts to guide the spiders towards better solutions or positions on the spider web.

The advantages of the proposed modified social spider algorithm (MSSA) are as follows:

- The MSSA provides superior convergence characteristics compared to that of the original SSA. The minimum cost, the mean cost, and the maximum cost values obtained by the MSSA are lower than those obtained by the original SSA.
- The results obtained by the MSSA have lower standard deviation compared to that of the original SSA.
- The proposed MSSA has lower number of parameters which require tuning compared to that of the original SSA. Moreover, the parameters of the proposed MSSA can be assumed constant at certain values with satisfactory results obtained over wide range of systems.
- The results obtained by the MSSA do not only outperform that obtained by the original SSA, but also outperform the results obtained by many recently proposed metaheuristic techniques for solving the non-convex ELD problem as will be demonstrated in Section 4 of this paper.

The remaining of the paper is organized as follows. Section 2 presents the non-convex ELD problem formulation. Section 3 reviews the original SSA algorithm and introduces the proposed MSSA algorithm. The simulation results which demonstrate the efficiency of the proposed algorithm are displayed in Section 4, and finally the conclusion is presented in Section 5.

2. Mathematical formulation of the problem

The ELD problem in its simplest form can be expressed as follows: Minimize:

$$F_T = \sum_{i=1}^n F_i(P_i) \quad (1)$$

Subject to:

$$P_i^{\min} \leq P_i \leq P_i^{\max} \quad (2)$$

$$\sum_{i=1}^n P_i = P_L + P_{Loss} \quad (3)$$

where $F_i(P_i)$ is the operation cost associated with unit i , when the output power of this unit is P_i , n is the total number of units, P_L refers to the system load, and P_{Loss} represents the total transmission losses of the system. P_i^{\min} and P_i^{\max} are the minimum and maximum output power given by unit i , respectively. The cost of the generated power can be calculated in terms of the output power using the following quadratic formula:

$$F_i(P_i) = a_i + b_i P_i + c_i P_i^2 \quad (4)$$

where a_i , b_i and c_i represent the fuel cost coefficients of unit i . Using the loss coefficient matrices B , B_0 , and B_{00} , the Kron's loss formula is used to calculate the total transmission losses as follows:

$$P_{Loss} = \sum_{i=1}^n \sum_{j=1}^n P_i B_{ij} P_j + \sum_{i=1}^n B_{0i} P_i + B_{00} \quad (5)$$

In steam power plants, as the steam is admitted to the units through the valves, ripples appear on the cost function of the units. These ripples can be modeled as a recurring rectified sinusoid imposed on the quadratic cost function, and the cost function that incorporates the valve point effects can be expressed as follows [6] and [7]:

$$F_i(P_i) = a_i + b_i P_i + c_i P_i^2 + |e_i \sin(f_i \times (P_i^{\min} - P_i))| \quad (6)$$

where a_i , b_i , c_i , e_i , and f_i are fuel cost coefficients associated with unit i .

Thermal and hydro power units may fail to operate safely within certain power ranges due to some physical limitations such as the occurrence of amplified vibrations in shaft bearings [7]. In order to model the prohibited operating zones, the problem formulation is augmented with the following constraints:

$$\begin{aligned} P_i^{\min} &\leq P_i \leq P_{i,1}^l & (i = 1, 2, \dots, n) \\ P_{i,j-1}^u &\leq P_i \leq P_{i,j}^l & (j = 2, 3, \dots, nj) \quad (i = 1, 2, \dots, n) \\ P_{i,nj}^u &\leq P_i \leq P_i^{\max} & (i = 1, 2, \dots, n) \end{aligned} \quad (7)$$

where $P_{i,j}^l$ represents the lower limit of the prohibited zone j of unit i , $P_{i,j}^u$ is the upper limit of the prohibited zone j of unit i , and nj is the total number of prohibited zones associated with unit i .

The thermal generating units have a limitation on changing their output power from one time interval to the next one. When the unit increases its output power, this limitation can be modeled using (8). When the unit decreases its output power, this limitation can be modeled using Eq. (9).

$$P_i - P_{i0} \leq UR_i \quad (8)$$

$$P_{i0} - P_i \leq LR_i \quad (9)$$

Using (8), (9), and (2), the actual limits of unit i can be expressed as follows:

$$\max(P_i^{\min}, P_{i0} - LR_i) \leq P_i \leq \min(P_i^{\max}, P_{i0} + UR_i) \quad (10)$$

where P_{i0} is the previous interval output power of unit i , UR_i is the upper ramp rate limit of unit i , and LR_i is the lower ramp rate limit of unit i .

3. The proposed approach

This section starts by reviewing the original SSA algorithm in the first subsection. After that, the proposed MSSA is introduced in the second subsection.

3.1. The original SSA

In SSA [36], the spider web represents the search space of the optimization problem. It is assumed that all feasible solutions of the optimization problem exist on the spider web. The spider position on the web represents a possible solution. When a spider moves to a new position, the spider creates a vibration carried by the web to the other spiders on that web. The vibration created by each spider contains two pieces of information; the position of the spider created the vibration and the fitness value corresponding to this position. The SSA algorithm starts by initializing a population of spiders distributed over the web, where the position of each spider represents a possible solution to the optimization problem. Each spider is associated with some pieces of information, which are updated during each iteration of the algorithm. These pieces of information are the spider position on the web, the fitness value corresponding to the spider position, the target vibration produced by the spider during the previous iteration, the number of iterations during which the spider did not change its target value, the movement of the spider in the previous iteration, and the dimension mask used by the spider in the previous iteration. The last four pieces of information are used to guide the movement of the spider from the current position to a new one. When a spider moves from one position to another, it produces a vibration. The intensity of this vibration is calculated using the following formula:

$$I_i = \log\left(\frac{1}{\bar{F}_i(P_i) - c} + 1\right) \quad (11)$$

where c is a small confidently constant. The vibration produced by the spider is transferred over the web and is sensed by all the spiders on the web including the spider which sends this vibration. Hence, all the spiders sense N vibrations, where N is the total number of spiders or the population size. Each spider senses its produced vibration as it was produced by Eq. (11). Each spider also senses the attenuated vibrations produced by other spiders on the web according to the following formula:

$$I_{ij} = I_i \times \exp\left(-\frac{D_{ij}}{\bar{\sigma} \times r_a}\right) \quad (12)$$

where I_{ij} is the vibration intensity sensed by spider j while the source of vibration is spider i . D_{ij} is the Euclidean distance between spider i and spider j . $\bar{\sigma}$ is the mean of the standard deviation of all spider positions along each dimension. r_a is a user controlled parameter. Small values of r_a means strong attenuation imposed on the vibration and vice versa. In each iteration, each spider will have N vibrations. After that, each spider finds the highest vibration

intensity among N vibrations and compares this highest vibration intensity with that in its memory. If this highest vibration is higher than the intensity kept in the spider memory, the spider will update the intensity of the vibration in its memory and will replace this intensity with the new higher one. Also, the spider will keep in its memory the position corresponding to this highest vibration as a target position. If this update process accomplished by a spider at a certain iteration, the spider will reset a counter d_i to zero. If the spider found that the highest vibration among N vibrations is lower than the one kept in its memory from the previous iteration, it will not change both the target position and the target intensity. After the spider determines its target position in each iteration, the spider starts a random walk towards this target position using a binary mask vector. This mask vector has a length equal to the problem dimension D . In each iteration, each spider changes its mask with a probability of $1 - P_c^{d_i}$, where $P_c \in (0, 1)$ is a variable defined by the user. Each element in the mask vector has a probability P_m to be assigned a value of 1 and $1 - P_m$ to be assigned a value of zero. After the binary mask vector is generated for each spider, a random vector is also generated which has a length equal to the problem dimension. The random vector is combined with the target vector through the mask in order to generate a new vector as follows:

$$P_{ij}^{\text{new}} = \begin{cases} P_{ij}^{\text{tar}} & \text{if } m_{ij} = 0 \\ P_{ij}^r & \text{if } m_{ij} = 1 \end{cases} \quad (13)$$

where P_{ij}^{tar} is the j -th dimension of the target position associated with spider i . P_{ij}^r is the j -th dimension of the random vector associated with spider i . r is a random integer value between $(1, N)$. N is the population size. m_{ij} is the j -th dimension of the binary mask vector accompanying spider i . P_{ij}^{new} is the j -th dimension of the new generated position for spider i . With this new position, the random walk for spider i is completed using the following formula:

$$P_i^{(t+1)} = P_i + (P_i - P_i^{(t-1)}) \times r_i + (P_i^{\text{new}} - P_i) \otimes R \quad (14)$$

where $P_i^{(t+1)}$, P_i , and $P_i^{(t-1)}$ are the position of spider i in the next iteration, the current iteration, and the previous iteration, respectively. r_i is a random number generated for spider i . This random number is uniformly distributed between zero and one. R is a vector of floating-point numbers uniformly distributed between zero and one and has a length equal to the problem dimensions. \otimes denotes element-wise multiplication.

3.2. Modified SSA

The proposed MSSA is the same as the original SSA except that the binary mask based random walk used to generate new solutions for the next iteration has been replaced with the following mutation process, after which a selection process is applied.

$$P_i^{\text{trial}} = P_i^{\text{tar}} + r_i \times K_i \otimes (P_{r_1}^{\text{tar}} - P_{r_2}^{\text{tar}}) \quad (15)$$

where P_i^{trial} is a trial vector generated for spider i . P_i^{tar} is the target vector associated with spider i . $P_{r_1}^{\text{tar}}$ and $P_{r_2}^{\text{tar}}$ are two target vectors associated with spiders r_1 and r_2 , where r_1 and r_2 are two integer random numbers between 1 and N , and $r_1 \neq r_2$. r_i is a random number between zero and one. This number is drawn from a uniform distribution. \otimes denotes element-wise multiplication. K_i is a binary vector for spider i . The elements of this binary vector are generated as follows:

$$K_{ij} = \begin{cases} 1 & \text{if } \text{rand}_{ij} < C \\ 0 & \text{Otherwise} \end{cases} \quad (16)$$

where C is a constant, $C \in (0, 1)$. $Rand_{i,j}$ is the j -th element of a random vector i . This element is generated randomly between zero and one and obeys the uniform distribution. After generating the trail vector i , the position of spider i in the next iteration is calculated using the following greedy selection process:

$$P_i^{(t+1)} = \begin{cases} P_i^{trial} & \text{if } f(P_i^{trial}) < f(P_i) \\ P_i & \text{Otherwise} \end{cases} \quad (17)$$

The above proposed modification has some similarity in its framework with the differential evolution algorithm; however, the proposed modification process is different than the differential evolution algorithm. The variables r_i and K_i in Eq. (15) have the effect of adapting the size of the change added to each target vector in the population. At the first iterations, the change added to each target vector in Eq. (15) is relatively large. This supports the exploration capability of the algorithm. When the algorithm starts to

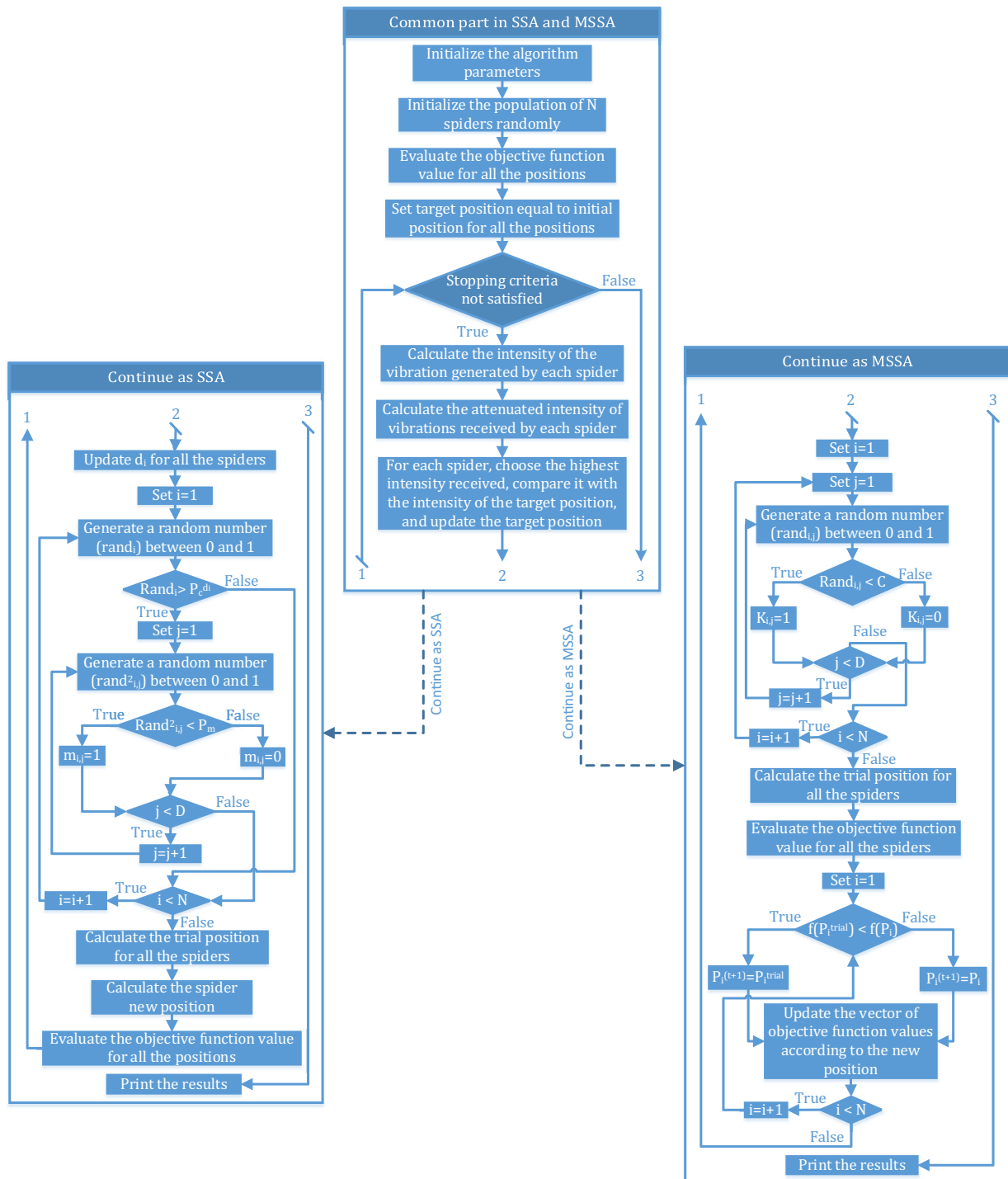


Fig. 1. Flowchart of the MSSA and the original SSA.

converge, the difference between the two target vectors chosen randomly will be reduced. This will make the change added to the target vector P_i^{tar} small, which will undoubtedly enhance the exploitation capability of the algorithm. The proposed MSSA requires only two parameters for tuning: the constant C and r_a . On the other hand, the original SSA algorithm requires the tuning of three parameters: r_a , P_c , and P_m . Fig. 1 shows a flowchart of the MSSA and the original SSA.

4. Results and discussion

In order to demonstrate the efficiency of the proposed algorithm, four benchmark test systems having 6-units [9], 40-units [37], 80-units [38], and 140-units [39] are considered. The parameters settings used for all the systems are the same, excluding from that the maximum number of iterations. Table 1 presents a list of parameters used with each system. A computer with core i7 (2.4 GHz) and 8 GB of RAM is used to run the MATLAB platform for simulating four case studies, one case study for each benchmark system. The total number of objective function evaluations is used to fairly compare the performance of the proposed algorithm with that of previously proposed algorithms. The total

Table 1
Parameters setting used with test systems.

	Number of iterations	Population size	C	r_a
6-unit	1200	10	0.2	1
40-unit	14,400	10	0.2	1
80-unit	16,000	10	0.2	1
140-unit	16,000	10	0.2	1

Table 2
Parameters setting for SSA-I and SSA-II.

	Population size	r_a	P_c	P_m
SSA-I	13	10	0.9	0.1
SSA-II	30	1	0.7	0.1

Table 3
Six generators test system: comparison of results.

Method	Min. (\$/hr)	Mean (\$/hr)	Max. (\$/hr)	TFE	Time (sec.)	Standard deviation
SA [40]	15461.10	15488.98	15545.50	NA	50.36	28.367
TS [40]	15454.89	15472.56	15498.05	NA	20.55	13.719
PSO [40]	15450.14	15465.83	15491.71	100,000	6.82	10.150
GA [9]	15459	15469	15524	20,000	41.58	0.057
PSO [9]	15450	15454	15492	20,000	14.89	0.002
NPSO-LRS[41]	15450	15450.5	15452	20,000	NA	NA
SSA-I	15449.899	15450.122	15462.080	12,000	1.07	1.408
SSA-II	15449.902	15450.256	15457.936	12,000	1.02	1.097
MSSA	15449.899	15449.937	15453.545	12,000	1.16	0.3647
RDPSO [42]*	15442.757	15445.024	15455.293	20,000	NA	2.2828
SSA [34]*	15419.803	NA	NA	100,000	0.338	NA

NA: Data are Not Available.

* The computed total losses with the Kron's loss formula are higher than the values reported in these references.

Table 4
Output power for six generators test system using the MSSA (100 trials).

P_1	P_2	P_3	P_4	P_5	P_6
447.5029	173.3186	263.4630	139.0656	165.4730	87.13490
Total power (MW)		1275.958	Total loss (MW)		12.958
Power Mismatch		-14×10^{-12}	Total Cost (\$/hr)		15449.8995

number of objective function evaluations (TFE) per one run of the algorithm is calculated as follows:

$$TFE = \lambda \times \text{Population size} \times \text{Total number of iterations} \quad (18)$$

where λ is the number of objective function evaluations executed by an algorithm when the population size is equal to 1 and the number of iterations is also equal to 1. The proposed MSSA algorithm has $\lambda = 1$.

In order to compare the performance of the proposed algorithm with that of the original social spider algorithm, the original social spider algorithm has been simulated in MATLAB with two optimized sets of parameters. The first optimized set of parameters is the set proposed in [34], and the SSA algorithm that uses this set of parameters is denoted by SSA-I. The second optimized set of parameters is the set reported in [36], and the SSA algorithm that uses this set of parameters is denoted by SSA-II. Table 2 lists the parameters of the SSA-I and SSA-II.

4.1. Case study 1

This case study utilizes a benchmark test system having 26 buses, 46 transmission lines, and 6 thermal units. The total system load is 1263 MW. In addition to the power balance and the generator limits constraints; ramp rate limits, prohibited operating zones, and transmission losses are considered. The cost coefficients, loss coefficients, ramp rate limits, and prohibited operating zones data are given in [9]. Table 3 presents a comparison between the results obtained by the MSSA, SSA-I, SSA-II, and other algorithms in the previous literature. Table 4 displays the output power computed for each unit using the MSSA. The statistical results of the MSSA, SSA-I, and SSA-II are obtained with 100 trails.

In Table 3, only the RDPSO [42] and the SSA [34] provided minimum total cost lower than the minimum total cost obtained by the MSSA; however, the computed total losses reported in references [42] and [34] do not agree with that computed using Kron's loss formula. In addition, according to [6], the total cost of the global optimal solution for 6-unit system considered in this case study should not be lower than 15447.72 \$/hr, since the latter is the optimal solution of the system without the ramp rate limits and the prohibited operating zones. From Table 3, it can be observed that

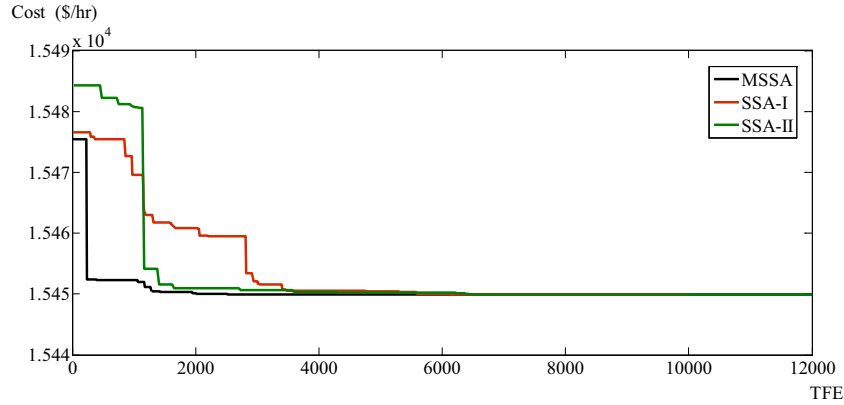


Fig. 2. Convergence characteristics of case study 1.

Table 5

40 generators test system: comparison of results.

Method	Min. (\$/hr)	Mean (\$/hr)	Max. (\$/hr)	TFE	Time (sec.)	Standard deviation
TLBO [43]	129960	NA	NA	NA	NA	NA
GSO [14]	124265.4	124609.18	125204.47	120,000	14.636	NA
IA_EDP [15]	121436.97	122492.70	121648.44	24,000	NA	182.527
BSA [16]	121415.61	121474.88	121524.95	600,000	13.147	0.295
TSARGA [44]	121,463.07	122,928.31	NA	NA	NA	50.4751
ACHS [17]	121414.85	121510.5	121655.66	60,000	2.18	54.28
CPSO [45]	121,865.23	122,100.87	NA	NA	114.65	NA
FCASO-SQP [46]	121,456.98	122,026.21	NA	NA	133.54	NA
CTLBO [47]	121553.83	121790.23	122116.18	NA	NA	150
SSA [34]	121412.55*	NA	NA	100,000	NA	NA
SSA-I	121448.32	121541.39	121648.32	144,001	5.504	38.3401
SSA-II	121507.25	121581.12	121643.00	144,000	4.137	30.2426
MSSA	121413.46	121466.61	121521.73	144,000	5.762	28.6932
CSA [48]	121412.53	121520.41	121810.25	160,000	3.03	81.5705

NA: Data are Not Available.

* Exact fuel cost obtained from the reported power output in [32] should be 121,414.95 \$/h instead of 121412.55 \$/h.

Table 6

Best solution for forty generators test system using the MSSA (100 trials).

P ₁	110.80	P ₂	110.83	P ₃	97.40	P ₄	179.73
P ₅	87.81	P ₆	140.00	P ₇	259.60	P ₈	284.60
P ₉	284.60	P ₁₀	130.00	P ₁₁	94.00	P ₁₂	94.00
P ₁₃	214.76	P ₁₄	394.28	P ₁₅	394.28	P ₁₆	394.28
P ₁₇	489.28	P ₁₈	489.28	P ₁₉	511.28	P ₂₀	511.28
P ₂₁	523.28	P ₂₂	523.28	P ₂₃	523.28	P ₂₄	523.28
P ₂₅	523.28	P ₂₆	523.28	P ₂₇	10.00	P ₂₈	10.00
P ₂₉	10.00	P ₃₀	87.93	P ₃₁	190.00	P ₃₂	190.00
P ₃₃	190.00	P ₃₄	164.80	P ₃₅	194.22	P ₃₆	200.00
P ₃₇	110.00	P ₃₈	110.00	P ₃₉	110.00	P ₄₀	511.28
Total Power (MW)		10500		Total cost (\$/hr)		121413.4686	

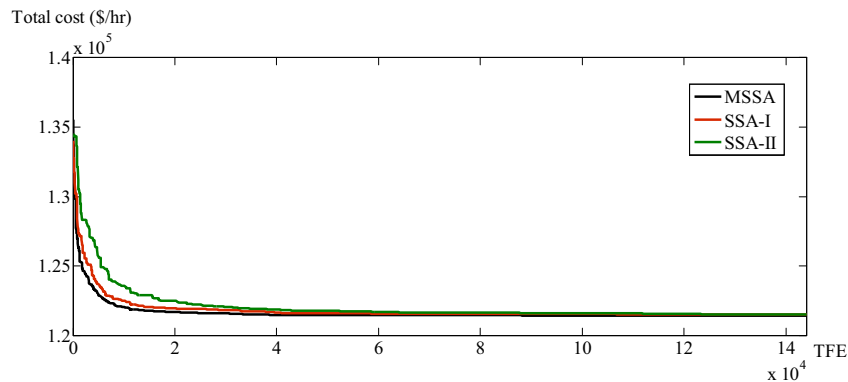


Fig. 3. Convergence characteristics of case study 2.

the results provided by the MSSA are better than those provided by the SSA-I and SSA-II. Fig. 2 shows the convergence characteristics of the studied algorithms.

4.2. Case study 2

Forty units system is used in this case study. All the units have valve point effects, and hence the cost functions are non-convex. The cost coefficients of this system are available in [37]. The system has a total load equal to 10500 MW. In this case study, the performance of the MSSA algorithm has been compared with that of the SSA-I, SSA-II, TLBO [43], GSO [14], IA_EDP [15], BSA [16],

TSARGA [44], ACHS [17], CPSSO [45], FCASO-SQP [46], CTLBO [47], SSA [34], and CSA [48]. Table 5 summarizes the results of this comparison. Again, the MSSA provided superior results than the SSA-I and SSA-II.

In Table 5, only CSA provided lower minimum cost than the MSSA; however, the mean value is much higher for the CSA than that of the MSSA. The best solution for the 40 units system is shown in Table 6. 100 independent trails are used to develop the statistical results of the studied algorithms. Fig. 3 displays the convergence characteristics of the MSSA, SSA-I, and SSA-II for case study 2. In Table 5, IA_EDP and ACHS provide their results at low TFE values; however, if the TFE value of the MSSA has been reduced

Table 7
80 generators test system: comparison of results.

Method	Min. (\$/hr)	Mean (\$/hr)	Max. (\$/hr)	TFE	Time (sec.)	Standard deviation
THS (t = 8) [44]	243,192.69	243,457.36	NA	NA	NA	120.99
FAPSO [49]	244,273.54	NA	NA	NA	NA	NA
CSO [38]	243,195.38	243,546.63	244,038.735	NA	27.76	NA
PSO [38]	244,188.35	246,375.87	248,483.592	NA	20.13	NA
SCA [38]	250,864.05	254,579.79	261,099.642	NA	848.34	NA
SSA-I	243,173.59	243,306.105	243,490.808	160,004	8.451	72.663
SSA-II	243,398.17	243,575.35	243,760.885	160,020	6.699	68.983
MSSA	242,909.25	243,037.25	243,229.947	160,000	8.668	53.762

NA: Data are Not Available.

Table 8
Best solution for eighty generators test system using the MSSA (100 trials).

P ₁	112.41	P ₂	113.60	P ₃	97.43	P ₄	179.74
P ₅	95.03	P ₆	140.00	P ₇	259.70	P ₈	284.63
P ₉	285.15	P ₁₀	130.01	P ₁₁	94.00	P ₁₂	168.84
P ₁₃	214.77	P ₁₄	394.28	P ₁₅	394.25	P ₁₆	394.33
P ₁₇	489.51	P ₁₈	489.31	P ₁₉	511.35	P ₂₀	511.31
P ₂₁	523.30	P ₂₂	523.36	P ₂₃	523.28	P ₂₄	523.34
P ₂₅	523.31	P ₂₆	523.33	P ₂₇	10.00	P ₂₈	10.02
P ₂₉	10.00	P ₃₀	88.13	P ₃₁	190.00	P ₃₂	190.00
P ₃₃	190.00	P ₃₄	165.28	P ₃₅	200.00	P ₃₆	200.00
P ₃₇	110.00	P ₃₈	110.00	P ₃₉	110.00	P ₄₀	511.34
P ₄₁	114	P ₄₂	110.91	P ₄₃	97.50	P ₄₄	179.88
P ₄₅	88.25	P ₄₆	140.00	P ₄₇	260.29	P ₄₈	285.11
P ₄₉	284.62	P ₅₀	130.01	P ₅₁	168.85	P ₅₂	94.00
P ₅₃	125.00	P ₅₄	304.58	P ₅₅	394.29	P ₅₆	394.29
P ₅₇	489.45	P ₅₈	489.24	P ₅₉	511.43	P ₆₀	511.28
P ₆₁	523.29	P ₆₂	523.30	P ₆₃	523.36	P ₆₄	523.31
P ₆₅	523.42	P ₆₆	523.29	P ₆₇	10.04	P ₆₈	10.00
P ₆₉	10.01	P ₇₀	90.08	P ₇₁	189.99	P ₇₂	190.00
P ₇₃	190.00	P ₇₄	165.27	P ₇₅	196.05	P ₇₆	200.00
P ₇₇	109.99	P ₇₈	110.00	P ₇₉	110.00	P ₈₀	511.31
Total Power (MW)		21,000		Total cost (\$/hr)		242,909.25	

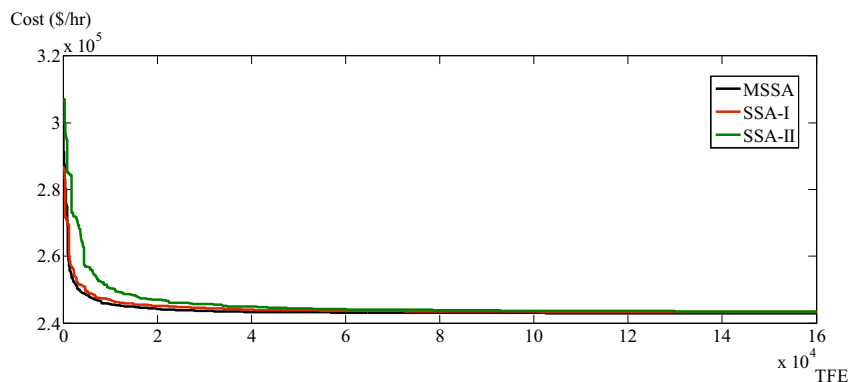


Fig. 4. Convergence characteristics of case study 3.

to the same TFE values of these algorithms, still better mean and standard deviation values can be obtained by the MSSA.

4.3. Case study 3

The 40-unit system data are replicated in this case study in order to simulate a system with 80 units. Since the system in this case study has 80 units with valve point effects, this system is more difficult to solve than the Korean 140-unit system, which will be presented in the next case study. The total

demand of the system is 21000 MW. This system has been studied before using THS (t = 8) [44], FAPSO [49], CSO [38], PSO [38] and SCA [38]. In Table 7, the results reported using these algorithms are compared with those obtained using the SSA-I, SSA-II, and MSSA. Both the MSSA and SSA-I provide superior results with the best results provided by the MSSA. Table 8 presents the best solution for the 80 units system produced by the MSSA. In this case study, the statistical results are obtained with 100 trails. Fig. 4 displays the convergence characteristics for this case study.

Table 9
140 generators test system: comparison of results.

Method	Min. (\$/hr)	Mean (\$/hr)	Max. (\$/hr)	TFE	Time (sec.)	Standard deviation
CCPSO [39]	1,657,962.73	1,657,962.73	1,657,962.73	300,000	150	0
CTPSO [39]	1,657,962.73	1,657,964.06	1,658,002.79	300,000	100	7.315
CSA [48]	1,655,746.14	1,655,904.66	1,661,572.41	200,000	38.90	592.7
MPSO [50]	1,560,436	1,560,445	1,560,462	60,000	18.43	NA
IDE [51]	1,564,648.66	1,564,663.54	1,564,682.73	250,000	27.88	NA
GWO [52]	1,559,953.18	1,560,132.93	1,560,228.40	NA	8.93	1.024
SSA-I	1,559,717.14	1,559,739.78	1,559,905.19	160,004	9.885	25.74
SSA-II	1,559,841.23	1,559,935.92	1,560,066.37	160,020	7.227	50.03
MSSA	1,559,708.7	1,559,708.82	1,559,709.21	160,000	10.49	0.110

NA: Data are Not Available.

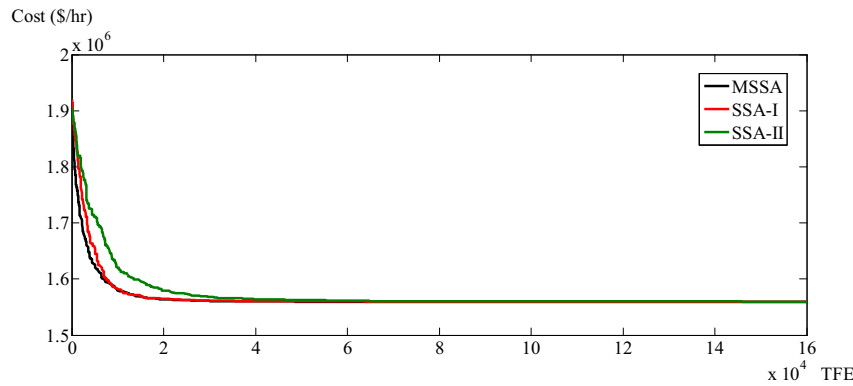


Fig. 5. Convergence characteristics of case study 4.

Table 10
Best solution for 140 generators test system using the MSSA (100 trials).

P ₁	115.1	P ₂₁	505.0	P ₄₁	3.0	P ₆₁	163.0	P ₈₁	398.1	P ₁₀₁	958.0	P ₁₂₁	175.0
P ₂	189.0	P ₂₂	505.0	P ₄₂	3.0	P ₆₂	95.0	P ₈₂	56.0	P ₁₀₂	1007.0	P ₁₂₂	2.0
P ₃	190.0	P ₂₃	505.0	P ₄₃	249.1	P ₆₃	160.0	P ₈₃	115.0	P ₁₀₃	1006.0	P ₁₂₃	4.0
P ₄	190.0	P ₂₄	505.0	P ₄₄	246.4	P ₆₄	160.0	P ₈₄	115.0	P ₁₀₄	1013.0	P ₁₂₄	15.0
P ₅	168.5	P ₂₅	537.0	P ₄₅	250.0	P ₆₅	490.0	P ₈₅	115.0	P ₁₀₅	1020.0	P ₁₂₅	9.0
P ₆	190.0	P ₂₆	537.0	P ₄₆	250.0	P ₆₆	196.0	P ₈₆	207.0	P ₁₀₆	954.0	P ₁₂₆	12.0
P ₇	490.0	P ₂₇	549.0	P ₄₇	241.4	P ₆₇	490.0	P ₈₇	207.0	P ₁₀₇	952.0	P ₁₂₇	10.0
P ₈	490.0	P ₂₈	549.0	P ₄₈	250.0	P ₆₈	490.0	P ₈₈	175.0	P ₁₀₈	1006.0	P ₁₂₈	112.0
P ₉	496.0	P ₂₉	501.0	P ₄₉	250.0	P ₆₉	130.0	P ₈₉	175.0	P ₁₀₉	1013.0	P ₁₂₉	4.0
P ₁₀	496.0	P ₃₀	501.0	P ₅₀	250.0	P ₇₀	234.7	P ₉₀	175.0	P ₁₁₀	1021.0	P ₁₃₀	5.0
P ₁₁	496.0	P ₃₁	506.0	P ₅₁	165.0	P ₇₁	137.0	P ₉₁	175.0	P ₁₁₁	1015.0	P ₁₃₁	5.0
P ₁₂	496.0	P ₃₂	506.0	P ₅₂	165.0	P ₇₂	325.5	P ₉₂	580.0	P ₁₁₂	94.0	P ₁₃₂	50.0
P ₁₃	506.0	P ₃₃	506.0	P ₅₃	165.0	P ₇₃	195.0	P ₉₃	645.0	P ₁₁₃	94.0	P ₁₃₃	5.0
P ₁₄	509.0	P ₃₄	506.0	P ₅₄	165.0	P ₇₄	175.0	P ₉₄	984.0	P ₁₁₄	94.0	P ₁₃₄	42.0
P ₁₅	506.0	P ₃₅	500.0	P ₅₅	180.0	P ₇₅	175.0	P ₉₅	978.0	P ₁₁₅	244.0	P ₁₃₅	42.0
P ₁₆	505.0	P ₃₆	500.0	P ₅₆	180.0	P ₇₆	175.0	P ₉₆	682.0	P ₁₁₆	244.0	P ₁₃₆	41.0
P ₁₇	506.0	P ₃₇	241.0	P ₅₇	103.0	P ₇₇	175.0	P ₉₇	720.0	P ₁₁₇	244.0	P ₁₃₇	17.0
P ₁₈	506.0	P ₃₈	241.0	P ₅₈	198.0	P ₇₈	330.0	P ₉₈	718.0	P ₁₁₈	95.0	P ₁₃₈	7.0
P ₁₉	505.0	P ₃₉	774.0	P ₅₉	312.0	P ₇₉	531.0	P ₉₉	720.0	P ₁₁₉	95.0	P ₁₃₉	7.0
P ₂₀	505.0	P ₄₀	769.0	P ₆₀	281.2	P ₈₀	531.0	P ₁₀₀	964.0	P ₁₂₀	116.0	P ₁₄₀	26.0
Total Power (MW)			49,342			Total cost (\$/hr)			1,559,711.999				

4.4. Case study 4

The last case study utilizes the Korean power system which is a real large scale system with 140 generating units [39], provides the data of this system. The system has four units with prohibited operating zones and twelve units with valve point effects. The ramp rate limits and the transmission losses are not considered in this case study. The system has a total load equal to 49,342 MW. In Table 9, the TFE, minimum cost, mean cost, maximum cost, mean computational time, and standard deviation values of the MSSA are compared with the corresponding values of the SSA-I, SSA-II, CCPSO [39], CTPSO [39], CSA [48], MPSO [50], IDE [51], and GWO [52].

From Table 9, it can be observed that the best results are obtained using the MSSA, and that the mean value of the MSSA is very close to the minimum value. 100 independent trails are executed to compute the statistical results of the studied algorithms. Fig. 5 shows the convergence characteristics of the studied algorithms. In Table 9, the MPSO uses much lower number of TFE than that used by the MSSA; however, if the same TFE value used by this algorithm has been considered for the MSSA, better results can still be obtained by the MSSA. It should be observed also that the mean simulation time of the MSSA is lower than that of this algorithm. The best solution for the 140 units system produced by the MSSA is presented in Table 10.

The total cost of the solution presented in Table 10 is 1,559,711.999 \$/hr. This cost is slightly different than the minimum cost reported in Table 9, since the output power values presented in Table 10 are rounded up to one decimal place, whereas the minimum cost value presented in Table 9 is computed based on the exact output power values obtained using the MATLAB.

5. Conclusion

A modified social spider algorithm is proposed for solving the non-convex economic dispatch problem. The proposed modification is to replace the binary mask based random walk with a new mutation process followed by a selection process. In order to demonstrate the significance of this modification, the MSSA has been compared with two variants of the original SSA: SSA-I and SSA-II. Each one of these two variants consists of the original SSA algorithm tuned with an optimized set of parameters reported in previous literature. The results obtained from the MSSA have also been compared with results presented in recent literature. The results obtained by the MSSA outperform the results obtained by the SSA-I, SSA-II and recently proposed algorithms. From the improved results provided by the MSSA, it is clear that the MSSA is less prone to premature convergence compared to the original SSA algorithm. In addition to that, the MSSA has less tunable parameters compared to the original SSA. Four benchmark test systems are used for comparing the proposed algorithm with the other algorithms. These systems are 6-units, 40 units, 80 units, and 140 units system.

References

- [1] Vinay Kumar Jadoun, Nikhil Gupta, K.R. Niazi, Anil Swarnkar, Modulated particle swarm optimization for economic emission dispatch, *Electr. Power Energy Syst.* 73 (2015) 80–88.
- [2] Ehab E. Elattar, A hybrid genetic algorithm and bacterial foraging approach for dynamic economic dispatch problem, *Electr. Power Energy Syst.* 69 (2015) 18–26.
- [3] Hari Mohan Dubey, Manjaree Pandit, B.K. Panigrahi, Hybrid flower pollination algorithm with time-varying fuzzy selection mechanism for wind integrated multi-objective dynamic economic dispatch, *Renewable Energy* 83 (2015) 188–202.
- [4] N. Jayakumar, S. Subramanian, S. Ganesan, E.B. Elanchezian, Grey wolf optimization for combined heat and power dispatch with cogeneration systems, *Electr. Power Energy Syst.* 74 (2016) 252–264.
- [5] Soheil Derafshi Beigvand, Hamdi Abdi, Massimo La Scala, Combined heat and power economic dispatch problem using gravitational search algorithm, *Electr. Power Syst. Res.* 133 (2016) 160–172.
- [6] W.T. Elsayed, E.F. El-Saadany, A fully decentralized approach for solving the economic dispatch problem, *IEEE Trans. Power Syst.* 30 (4) (2015) 2179–2189.
- [7] T. Ding, R. Bo, F. Li, H. Sun, A bi-level branch and bound method for economic dispatch with disjoint prohibited zones considering network losses, *IEEE Trans. Power Syst.* 30 (6) (2015) 2841–2855.
- [8] D.C. Walters, G.B. Sheble, Genetic algorithm solution of economic dispatch with valve point loading, *IEEE Trans. Power Syst.* 8 (3) (1993) 1325–1332.
- [9] Z.L. Gaing, Particle swarm optimization to solving the economic dispatch considering the generator constraints, *IEEE Trans. Power Syst.* 18 (3) (2003) 1187–1195.
- [10] N. Nomana, H. Iba, Differential evolution for economic load dispatch problems, *Electr. Power Syst. Res.* 78 (3) (2008) 1322–1331.
- [11] B. Shaw, V. Mukherjee, S.P. Ghoshal, Seeker optimisation algorithm: application to the solution of economic load dispatch problems, *IET Gener. Transm. Distrib.* 5 (1) (2011) 81–91.
- [12] Ling Wang, Ling-po Li, An effective differential harmony search algorithm for the solving non-convex economic load dispatch problems, *Int. J. Electr. Power Energy Syst.* 44 (1) (2013) 832–843.
- [13] Dinu Calin Secui, The chaotic global best artificial bee colony algorithm for the multi-area economic/emission dispatch, *Energy* 93 (2015) 2518–2545.
- [14] M.M. Dalvand, B.M. Ivatloo, A. Najafi, A. Rabiee, Continuous quick group search optimizer for solving nonconvex economic dispatch problems, *Electr. Power Syst. Res.* 93 (2012) 93–105.
- [15] V. Aragón, S. Esquivel, C.C. Coello, An immune algorithm with power redistribution for solving economic dispatch problems, *Inf. Sci.* 295 (2015) 609–632.
- [16] M. Modiri-Delshad, N. Abd Rahim, Solving non-convex economic dispatch problem via backtracking search algorithm, *Energy* 77 (2014) 372–381.
- [17] Q. Niu, H. Zhang, X. Wang, K. Li, G.W. Irwin, A hybrid harmony search with arithmetic crossover operation for economic dispatch, *Int. J. Electr. Power Energy Syst.* 62 (2014) 237–257.
- [18] Chiranjeevi Karri, Umaranjan Jena, Fast vector quantization using a Bat algorithm for image compression, *Eng. Sci. Technol. Int. J.* 19 (2016) 769–781.
- [19] K. Premkumar, B.V. Manikandan, Bat algorithm optimized fuzzy PD based speed controller for brushless direct current motor, *Eng. Sci. Technol. Int. J.* 19 (2016) 818–840.
- [20] D.K. Sambariya, R. Gupta, R. Prasad, Design of optimal input–output scaling factors based fuzzy PSS using bat algorithm, *Eng. Sci. Technol. Int. J.* 19 (2016) 991–1002.
- [21] P. Balachennaiah, M. Suryakalavathi, Palukuru Nagendra, Optimizing real power loss and voltage stability limit of a large transmission network using firefly algorithm, *Eng. Sci. Technol. Int. J.* 19 (2016) 800–810.
- [22] Janmenjoy Nayak, Bighnaraj Naik, H.S. Behera, A novel nature inspired firefly algorithm with higher order neural network: performance analysis, *Eng. Sci. Technol. Int. J.* 19 (2016) 197–211.
- [23] A. Gupta, P.K. Padhy, Modified Firefly Algorithm based controller design for integrating and unstable delay processes, *Eng. Sci. Technol. Int. J.* 19 (2016) 548–558.
- [24] Pratap Chandra Pradhan, Rabindra Kumar Sahu, Sidhartha Panda, Firefly algorithm optimized fuzzy PID controller for AGC of multi-area multi-source power systems with UPFC and SMES, *Eng. Sci. Technol. Int. J.* 19 (2016) 338–354.
- [25] A.Y. Abdelaziz, E.S. Ali, S.M. Abd Elazim, Flower pollination algorithm to solve combined economic and emission dispatch problems, *Eng. Sci. Technol. Int. J.* 19 (2016) 980–990.
- [26] A.Y. Abdelaziz, E.S. Ali, S.M. Abd Elazim, Optimal sizing and locations of capacitors in radial distribution systems via flower pollination optimization algorithm and power loss index, *Eng. Sci. Technol. Int. J.* 19 (2016) 610–618.
- [27] Rabindra Kumar Sahu, Tulasichandra Sekhar Gorripotu, Sidhartha Panda, Automatic generation control of multi-area power systems with diverse energy sources using Teaching Learning Based Optimization algorithm, *Eng. Sci. Technol. Int. J.* 19 (2016) 113–134.
- [28] R. Venkata Rao, Dhiraj P. Rai, Optimization of fused deposition modeling process using teaching learning-based optimization algorithm, *Eng. Sci. Technol. Int. J.* 19 (2016) 587–603.
- [29] Dharmbir Prasada, V. Mukherjee, A novel symbiotic organisms search algorithm for optimal power flow of power system with FACTS devices, *Eng. Sci. Technol. Int. J.* 19 (2016) 79–89.
- [30] Apoorva Aggarwal, Tarun Kumar Rawat, Manjeet Kumar, D.K. Upadhyay, Optimal design of FIR high pass filter based on L1 error approximation using real coded genetic algorithm, *Eng. Sci. Technol. Int. J.* 18 (2015) 594–602.
- [31] M. Balasubbarreddy, S. Sivanagaraju, Chintalapudi V. Suresh, Multi-objective optimization in the presence of practical constraints using non-dominated sorting hybrid cuckoo search algorithm, *Eng. Sci. Technol. Int. J.* 18 (2015) 603–615.
- [32] V. Sakthivel, Elizabeth Elias, Design of low complexity sharp MDFT filter banks with perfect reconstruction using hybrid harmony-gravitational search algorithm, *Eng. Sci. Technol. Int. J.* 18 (2015) 648–657.
- [33] Mohammadreza Azimi, Ahmad Mozaffari, Heat transfer analysis of unsteady graphene oxide nanofluid flow using a fuzzy identifier evolved by genetically encoded mutable smart bee algorithm, *Eng. Sci. Technol. Int. J.* 18 (2015) 106–123.

- [34] J.J.Q. Yu, V.O.K. Li, A social spider algorithm for solving the non-convex economic load dispatch problem, *Neurocomputing* 171 (2016) 955–965.
- [35] Erik Cuevas, Miguel Cienfuegos, Daniel Zaldivar, Marco Pérez-Cisneros, A swarm optimization algorithm inspired in the behavior of the social-spider, *Expert Syst. Appl.* 40 (2013) 6374–6384.
- [36] J.J.Q. Yu, V.O.K. Li, A social spider algorithm for global optimization, *Appl. Soft Comput.* 30 (2015) 614–627.
- [37] N. Sinha, R. Chakrabarti, P.K. Chattopadhyay, Evolutionary programming techniques for economic load dispatch, *IEEE Trans. Evol. Comput.* 7 (1) (2003) 83–94.
- [38] A. Selvakumar, T. Khanushkodi, Optimization using civilized swarm: solution to economic dispatch with multiple minima, *Electr. Power Syst. Res.* 79 (1) (2009) 8–16.
- [39] J.-B. Park, Y.-W. Jeong, J.-R. Shin, K. Lee, An improved particle swarm optimization for nonconvex economic dispatch problems, *IEEE Trans. Power Syst.* 25 (1) (2010) 156–166.
- [40] Saravuth Pothiya, Issarachai Ngamroo, Waree Kongprawechnon, Application of multiple tabu search algorithm to solve dynamic economic dispatch considering generator constraints, *Energy Convers. Manage.* 49 (2008) 506–516.
- [41] I. Selvakumar, K. Thanushkodi, A new particle swarm optimization solution to nonconvex economic dispatch problems, *IEEE Trans. Power Syst.* 22 (1) (2007) 42–51.
- [42] Jun Sun, Vasile Palade, Wu Xiao-Jun, Wei Fang, Zhenyu Wang, Solving the power economic dispatch problem with generator constraints by random drift particle swarm optimization, *IEEE Trans. Industr. Inf.* 10 (1) (2014) 222–232.
- [43] S. Banerjee, D. Maity, C.K. Chanda, Teaching learning based optimization for economic load dispatch problem considering valve point loading effect, *Int. J. Electr. Power Energy Syst.* 73 (2015) 456–464.
- [44] Mohammed Azmi Al-Betar, Mohammed A. Awadallah, Ahamad Tajudin Khader, Asaju L. Bolaji, Tournament-based harmony search algorithm for non-convex economic load dispatch problem, *Appl. Soft Comput.* 47 (2016) 449–459.
- [45] Jiejun Cai, Qiong Li, Lixiang Li, Haipeng Peng, Yixian Yang, A hybrid CPSO–SQP method for economic dispatch considering the valve-point effects, *Energy Convers. Manage.* 53 (2012) 175–181.
- [46] Jiejun Cai, Qiong Li, Lixiang Li, Haipeng Peng, Yixian Yang, A hybrid FCASO–SQP method for solving the economic dispatch problems with valve-point effects, *Energy* 38 (2012) 346–353.
- [47] Xiangzhu He, Yunqing Rao, Jida Huang, A novel algorithm for economic load dispatch of power Systems, *Neurocomputing* 171 (2016) 1454–1461.
- [48] D.N. Vo, P. Schegner, W. Ongsakul, Cuckoo search algorithm for nonconvex economic dispatch, *IET Gener. Transm. Distrib.* 7 (6) (2013) 645–654.
- [49] Taher Niknam, Hasan Doagou Mojarad, Hamed Zeinoddini Meymand, Non-smooth economic dispatch computation by fuzzy and self-adaptive particle swarm optimization, *Appl. Soft Comput.* 11 (2011) 2805–2817.
- [50] M. Basu, Modified particle swarm optimization for nonconvex economic dispatch problems, *Int. J. Electr. Power Energy Syst.* 69 (2015) 304–312.
- [51] M. Basu, Improved differential evolution for economic dispatch, *Int. J. Electr. Power Energy Syst.* 63 (2014) 855–861.
- [52] M. Pradhan, P. Kumar Roy, T. Pal, Grey wolf optimization applied to economic load dispatch problems, *Int. J. Electr. Power Energy Syst.* 83 (2016) 325–334.