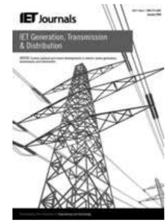
www.ietdl.org

Published in IET Generation, Transmission & Distribution
Received on 20th October 2012
Revised on 30th January 2013
Accepted on 17th February 2013
doi: 10.1049/iet-gtd.2012.0616



ISSN 1751-8687

Robust, fast and optimal solution of practical economic dispatch by a new enhanced gradient-based simplified swarm optimisation algorithm

Rasoul Azizipanah-Abarghooee, Taher Niknam, Masihallah Gharibzadeh, Faranak Golestaneh

Department of Electronic and Electrical Engineering, Shiraz University of Technology, Modares Boulevard, Shiraz, Iran
E-mail: r.azizipanah@sutech.ac.ir

Abstract: Nowadays, the rising concern about the costs of fuel and operation of generating units in the energy control centre deserve progress of solution methodologies for the practical economic dispatch (ED) problems. ED aims to schedule the committed units' output power while satisfying practical constraints and load demand. The generator ramp rate limits, non-convex and discontinuous nature of prohibited operating zones, non-smooth characteristic of valve-point effects, multi-fuel type of generation units, transmission line losses and the large number of units in practical power plants make this problem very hard to solve. In this study, a new solution method of integrating the classical gradient-based optimisation technique and a new enhanced simplified swarm optimisation algorithm is comprehensively presented and successfully applied to determine the feasible, robust, fast and globally or near-globally optimal solution within a rapid timeframe for the ED problems. The simulations are carried out on four-test systems, including 10-, 15-, 40- and 80-units using the proposed optimisation technique in the Fortran Power Station 4.0 software. The current proposal outperforms other methods showcased in the recent state-of-the art literature in the area.

Nomenclature

Indices

f	fuel-type index
g	iteration index of the gradient method (GM)
i, j	generating unit indices
iter	iteration index of the enhanced gradient-based simplified swarm optimisation algorithm (EGSSOA)
k	POZ index
n	particle index
t	time step index

Constants

a_i, b_i, c_i, d_i, e_i	cost coefficients of generating unit i
$a_{i,f}, b_{i,f}, c_{i,f}, d_{i,f}, e_{i,f}$	cost coefficients of generating unit i with fuel type f
$B_{i,j}(t)$	loss coefficient relating the productions of generating units i and j at time t (MW^{-1})
$B_{0,i}(t)$	loss coefficient associated with the production of generating unit i at time t
$B_{00}(t)$	loss coefficient parameter (MW)
DR_i, UR_i	ramp-down/ramp-up rate of generating unit i, respectively (MW/h)
$g_{\max}$	maximum number of iteration for GM
$\text{iter}_{\max}$	maximum number of iteration for EGSSOA
K_i	number of prohibited operating zones (POZs) for generating unit i
N_1, N_2, N_3	number of particles that select the mutation strategy 1, 2 and 3, respectively
NG	number of generating units
NF_i	number of fuel types for generating unit i
NP	number of particles in the population of EGSSOA
$P_D(t)$	load demand at time t (MW)
$P_{i,\max}$	power capacity of generating unit i (MW)
$P_{i,\min}$	minimum power output of generating unit i (MW)
$P_{i,k}^L$, $P_{i,k}^U$	lower/upper boundary of the kth POZ for generating unit i, respectively
rand(·), rand1(·), rand2(·)	random function generators in the range [0,1]
S	iteration constant for the mutation strategy
α^g	constant parameters in the gth iteration of the GM

Variables

$F(\boldsymbol{P}_G)$	fuel cost ($)
$F(\boldsymbol{X}_n)$	fitness function of the nth particle
$F_{\min}$	lower bound of fitness function in which all of the variables are set to the lower bounds
$\boldsymbol{G}\mathbf{best}^{\text{iter}}$	best particle with the cheapest fuel cost among all particles in iteration iter of EGSSOA
$\boldsymbol{P}\mathbf{best}_n^{\text{iter}}$	best position of particle n at the iter-th iteration of EGSSOA
$\boldsymbol{P}_G$	generating unit vector
$P_i(t)$	power output of generating unit i at time t (MW)
$P_{\text{Loss}}(t)$	total real power losses at time t (MW)
$\bar{P}_{i,t}$, $\underline{P}_{i,t}$	upper/lower limit of the ith generating unit output power at time t, respectively (MW)
$P\_\text{violate}_g(t)$	power mismatch at the gth step of GM (MW)
$P_i^g(t)$	power output of the ith generating unit at time t in the gth step of GM (MW)
$P_{\text{Loss}}^g(t)$	total real power losses at time t in the gth step of GM (MW)
$x_{n,i}^{\text{iter}}$	the ith member of the nth particle at the iter-th iteration of EGSSOA
$\mathbf{Worst}^{\text{iter}}$	worst solution with the most expensive fuel cost among all particles in iteration iter of EGSSOA
λ^g	Lagrange multiplier in the gth step of GM

1 Introduction

Economic dispatch (ED) is one of the backbone optimisation tools of electric power grids for both the generating companies (GENCOs), which contest in a free electricity market, and the systems operator (SO), who charges a fair handling of transactions between GENCOs and their distributing companies (DISCOs). From the point of view of GENCOs, the objective of optimisation tool is to maximise the profit of GENCOs. Besides, from the point of view of SO, the objective of optimisation tool is to minimise the fuel cost function satisfying load demand and many practical constraints like ramping limitation of generating units, multi-fuel type of units, prohibited operating zones (POZs), valve-point effects and transmission line losses [1]. By adding a sinusoidal term to the traditional quadratic cost function and taking valve-point effect into account, the cost function of generators takes a non-smooth characteristic. Also, the modern generators with valve-point effects have many POZs. The POZ arises from physical limitations of individual power plant components and may lead to instabilities in operation for certain loads. To avoid these instabilities, the concept of POZ is considered in many technical literatures. This leads to non-convexity and non-continually forms in the problem search domain. Considering the multi-fuel type of units causes the objective function to get more complicated. Meanwhile, the number of local optima is increased by considering all of the above constraints. Thus, an accurate optimisation method is needed to solve the ED problems.

In the past decades, numerous mathematical and heuristic-based optimisation techniques are reported in the literatures to handle the ED problems. The first group consists of gradient method (GM) [2], linear programming [3], lambda iteration method [4], quadratic programming [5], non-linear programming [6], Lagrange multipliers [7], novel direct search [8] and dynamic programming (DP) [9] techniques. It should be pointed out that although some of these methods are widely used by GENCOs or SOs due to their excellent convergence speed and unique solution, but they have some drawbacks such as: (i) The theoretical concept behind these methods except DP tailors them to suit for a specific ED formulation; (ii) If the initial conjecture for the mathematical-based optimisation methods happens to be in the neighbourhood of a local solution, then they may converge to local optima and get it as a global ones to SO or GENCO; and (iii) The DP method suffers from the curse of dimensionally leading to enormous execution time. To overcome these insufficiencies, the second meta-heuristic optimisation techniques have been introduced to solve the various ED problems. The recent state-of-the art literatures published in 2012 in the area of second group are GA-API [1], modified group search optimiser (MGSO) [10], Firefly algorithm (FA) [11], iteration particle swarm optimisation with time varying acceleration coefficients (IPSO-TVAC) [12], improved PSO (IPSO) [13], continuous quick group search optimiser (CQGSO) [14], differential harmony search (DHS) [15], modified PSO [16], fuzzy adaptive chaotic ant swarm optimisation (FCASO) [17], augmented Lagrange hopfield network (ALHN) [18], hybrid chaotic PSO and sequential quadratic programming (CPSO-SQP) [19], seeker optimisation algorithm (SOA) [20], enhanced augmented Lagrange hopfield network (EALHN) [21], hybrid accelerated biogeography-based optimisation and modified differential evolution (aBBOmDE) [22] and a new gumption approach [23]. The solution to the ED problems using these meta-heuristic optimisation methods proposed in the aforementioned literatures consumes huge executing time. Also, these methods do not always promise to obtain the globally or near-globally optimal solution.

To date, little attention has been paid to establish a powerful solution method in order to find the global or near-global solution to the ED problems. In this paper, a new hybrid method that profits from the combining of mathematical gradient-based and enhanced simplified swarm optimisation algorithm (ESSOA) as a novel meta-heuristic technique is proposed to obtain a robust, fast and globally or near-globally optimal solution that actually examines all of the presented equality and inequality constraints. Simplified swarm optimisation algorithm (SSOA) is a powerful population-based meta-heuristic optimisation algorithm proposed by Yeh in 2012, which overcomes the drawback of the PSO algorithm [24]. Also, SSOA is a very simple, robust and fast algorithm, and it has been applied to solve various optimisation problems [24] successfully. However, the original SSOA often suffers problems such as trapped in local optima or converge to optimal value in long time. In order to avoid these problems, a new mutation strategy is utilised, with modifying SSOA called ESSOA that can overcome the above problems. In addition, in order to reach global or near-global solution in an acceptable computation time, for each particle of the population, a local optimiser gradient-based method is implemented. This method starts from the ESSOAs generated particle in each iterate of the ESSOA. It works on the principle that the minimum of cost function of each particle can be found by a series of steps that always take us in a downward direction. It is noteworthy that this hybrid algorithm is called enhanced gradient-based SSOA (EGSSOA). The proposed framework

www.ietdl.org

is applied on four small, medium and large-scale test systems to show the superiority of the EGSSOA. In this regard, some metrics are used in order to authenticate the performance of the obtained solutions.

The rest of this paper is organised as follows: Section 2 proposes the problem formulation and constraints, Section 3 expresses the proposed hybrid optimisation algorithm, Section 4 describes the implementation of the proposed algorithm on the ED problem and Section 5 depicts the simulation results.

2 Problem formulation

The aim of this section is to define the fuel cost minimisation while satisfying operational constraints; basically, the problem is formulated as follows:

Minimise fuel cost [1]

$$F(\boldsymbol{P}_G) = \sum_{i=1}^{NG} F_i(P_i(t)) = \sum_{i=1}^{NG} \left(a_i + b_i P_i(t) + c_i P_i^2(t) + \left|d_i \sin\left(e_i\left(P_{i,\min} - P_i(t)\right)\right)\right|\right) \quad (1)$$

subject to

$$\sum_{i=1}^{NG} P_i(t) = P_D(t) + P_{\text{Loss}}(t) \quad (3)$$

$$P_{\text{Loss}}(t) = \sum_{i=1}^{NG}\sum_{j=1}^{NG} P_i(t) B_{i,j}(t) P_j(t) + \sum_{i=1}^{NG} B_{0,i}(t) P_i(t) + B_{00}(t) \quad (4)$$

$$P_i(t) - P_i(t-1) \le UR_i \quad i = 1, \ldots, NG \quad (5)$$

$$P_i(t-1) - P_i(t) \le DR_i \quad i = 1, \ldots, NG \quad (6)$$

$$\underline{P}_i(t) \le P_i(t) \le \bar{P}_i(t) \quad i = 1, \ldots, NG \quad (7)$$

$$\begin{aligned} \bar{P}_i(t) &= \min\left(P_{i,\max}, P_i(t-1) + UR_i\right) \quad i = 1, \ldots, NG \\ \underline{P}_i(t) &= \max\left(P_{i,\min}, P_i(t-1) - DR_i\right) \quad i = 1, \ldots, NG \end{aligned} \quad (8)$$

$$P_i(t) \in \begin{cases} P_{i,\min} \le P_i(t) \le P_{i,1}^L \\ P_{i,k-1}^U \le P_i(t) \le P_{i,k}^L, & k = 2, 3, \ldots, K_i, \\ P_{i,K_i}^U \le P_i(t) \le P_{i,\max} \end{cases} \quad (9)$$
$$i = 1, \ldots, NG$$

where, $\boldsymbol{P}_G = \left[P_1(t) \quad P_2(t) \quad \ldots \quad P_{NG}(t)\right]$. In some literatures, the cost function is considered to be a square function [18]. Power plants generators usually have multiple valves in order to obtain the power output of the unit under control. When steam admission valves in thermal units are first opened, a sudden increase in losses is observed. This sudden increase leads to ripples in the cost function, which is known as the valve-point loading effect and showcases as a sinusoidal component in the fuel cost function (1).

It is necessary to note that definition of cost function in (1) is related to the units with one type fuel, therefore this equation is not valid for multi-fuel-type units. Thus, the cost function of multi-fuel-type units is defined as constraint (2). Constraint (3) represents the power balance equation in each period. Using the approximation based on the B-coefficients [25], transmission network losses are expressed in (4). Constraints (5) and (6) impose up and down ramp rate limits, respectively. Generation limits are set in (7) and (8). Finally, constraint (9) defines the POZs for each thermal unit. A generating unit with multiple POZs has discontinuous input–output characteristics, since it is difficult to determine the actual POZs by real performance testing or operating records, so normally the best economy is achieved by avoiding operation in areas that are in actual operation. Therefore it is necessary to determine a mathematic formulation for POZs. Each generating unit i with K_i POZ is modelled by $(K_i + 1)$ disjoint feasible operating sub-regions, that is, $\left[P_{i,\min}, P_{i,1}^L\right], \ldots, \left[P_{i,k-1}^U, P_{i,k}^L\right], \ldots, \left[P_{i,K_i}^U, P_{i,\max}\right]$, in which all of them are determined by K_i POZ, that is, $\left(P_{i,1}^L, P_{i,1}^U\right), \ldots, \left(P_{i,k}^L, P_{i,k}^U\right), \ldots, \left(P_{i,K_i}^L, P_{i,K_i}^U\right)$. To better illustrate the role of POZ in the operating region of generating unit, the cost function of a generating unit with two POZs is depicted in Fig. 1.

3 Enhanced gradient-based simplified swarm optimisation algorithm

3.1 Overview of the original SSOA

The SSOA is a newly developed optimisation algorithm, which is introduced by Yeh [24]. In this study, the SSOA consists of a number of particles ($\boldsymbol{X}_n$, $n = 1, \ldots,$ NP) with different fitness function $F(\boldsymbol{X}_n)$. Each particle comprises a

$$F(\boldsymbol{P}_G) = \begin{cases} \text{if } P_{i,\min} \le P_i(t) \le P_{i,1}; \quad \text{Fuel type 1} \\ \sum_{i=1}^{NG} \left(a_{i,1} + b_{i,1} P_i(t) + c_{i,1} P_i^2(t) + \left|d_{i,1} \sin\left(e_{i,1}\left(P_{i,\min} - P_i(t)\right)\right)\right|\right) \\ \text{if } P_{i,1} \le P_i(t) \le P_{i,2}; \quad \text{Fuel type 2} \\ \sum_{i=1}^{NG} \left(a_{i,2} + b_{i,2} P_i(t) + c_{i,2} P_i^2(t) + \left|d_{i,2} \sin\left(e_{i,2}\left(P_{i,\min} - P_i(t)\right)\right)\right|\right) \\ \vdots \\ \text{if } P_{i,NF_i-1} \le P_i(t) \le P_{i,\max}; \quad \text{Fuel type } NF_i \\ \sum_{i=1}^{NG} \left(a_{i,NF_i} + b_{i,NF_i} P_i(t) + c_{i,NF_i} P_i^2(t) + \left|d_{i,NF_i} \sin\left(e_{i,NF_i}\left(P_{i,\min} - P_i(t)\right)\right)\right|\right) \end{cases} \quad (2)$$

doi: 10.1049/iet-gtd.2012.0616

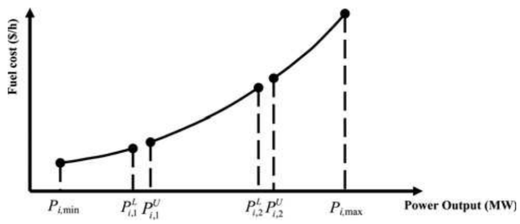


Fig. 1 *Fuel cost function of a generating unit with two POZs*

set of real numbers, each of which representing the MW output of a generating unit. The diagram of each particle of the population is delineated in Fig. 2. The best solution, that is, the closest to the requirements of the SO, is a solution with lowest fitness function. This algorithm is a simplified version of the PSO algorithm, which does not need to update its velocity, but the position of each particle should be changed and its updated equation is different from the traditional PSO. Most of the evolutionary algorithms, such as PSO, SSOA, and so on, initialise in their search space and go through the best optimal solution based on their updating technique. The main difference between all of the evolutionary algorithms is the updating technique for their particles. The basic idea behind the SSOA is that each variable value $x_{n,i}^{\text{iter}}$, $n = 1, \ldots, \text{NP}$ in the control vector $\boldsymbol{X}_n^{\text{iter}}$, $n = 1, \ldots, \text{NP}$ can be generated from four positions:

1. Its previous value in the previous iteration ($x_{n,i}^{\text{iter}-1}$, $n = 1, \ldots, \text{NP}$);
2. The same position in the best solution with the lowest fitness function found for $\boldsymbol{X}_n^{\text{iter}-1}$, $n = 1, \ldots, \text{NP}$ until iteration iter named $\boldsymbol{P}\mathbf{best}_n^{\text{iter}-1}$; $n = 1, \ldots, \text{NP}$;
3. The same position in the global solution found until iteration iter named $\boldsymbol{G}\mathbf{best}^{\text{iter}-1}$;
4. A random number generated in the lower bound and upper bound limits of $x_{n,i}^{\text{iter}-1}$, $n = 1, \ldots, \text{NP}$; this random number is shown as $x_{n,i}'^{\text{iter}-1}$, $n = 1, \ldots, \text{NP}$.

SSOA is characterised by $\boldsymbol{P}\mathbf{best}_n^{\text{iter}}$ and $\boldsymbol{G}\mathbf{best}^{\text{iter}}$, which can be defined as $\boldsymbol{P}\mathbf{best}_n^{\text{iter}} = [\,P\text{best}_{n,1}^{\text{iter}} \quad P\text{best}_{n,2}^{\text{iter}} \quad \ldots \quad P\text{best}_{n,NG}^{\text{iter}}\,]$ and $\boldsymbol{G}\mathbf{best}^{\text{iter}} = [\,G\text{best}_1^{\text{iter}} \quad G\text{best}_2^{\text{iter}} \quad \ldots \quad G\text{best}_{NG}^{\text{iter}}\,]$, respectively.

According to the above discussion, the updating mechanism of the SSOA can be formulated as follows [24]

$$x_{n,i}^{\text{iter}} = \begin{cases} x_{n,i}^{\text{iter}-1}, & \text{if } 0 \le \text{rand}(\cdot) < C_w \\ Pbest_{n,i}^{\text{iter}-1}, & \text{if } C_w \le \text{rand}(\cdot) < C_p \\ Gbest_i^{\text{iter}-1}, & \text{if } C_p \le \text{rand}(\cdot) < C_g \\ x_{n,i}'^{\text{iter}}, & \text{if } C_g \le \text{rand}(\cdot) \le 1 \end{cases} \tag{10}$$

Note that the settings of C_w, C_p and C_g have significant effects in the exploration and exploitation procedure in the SSOA. They represent the probabilities of the new variable value generated from the $\boldsymbol{X}_n^{\text{iter}-1}$, $\boldsymbol{P}\mathbf{best}_n^{\text{iter}}$ and $\boldsymbol{G}\mathbf{best}^{\text{iter}}$, respectively. Therefore, in this paper an enriched version of SSOA is utilised in which these parameters are tuned according to the combination of self-adaptive parameter control 1 (SPC1) and SPC2 described in (11)–(15), such that better solutions are likely to produce the new solution in iteration iter [24, 26]

$$F^{\#}(\bullet) = \frac{F(\bullet)}{F(\bullet) - F_{\min}} \tag{11}$$

$$FF_n^{\text{iter}} = F^{\#}\left(\boldsymbol{G}\mathbf{best}^{\text{iter}-1}\right) + F^{\#}\left(\boldsymbol{P}\mathbf{best}_n^{\text{iter}-1}\right) + F^{\#}\left(\boldsymbol{X}_n'^{\text{iter}}\right) + F^{\#}\left(\boldsymbol{X}_n^{\text{iter}-1}\right) \tag{12}$$

$$C_g = \begin{cases} 0.85, & \text{if } \dfrac{F^{\#}\left(\boldsymbol{X}_n'^{\text{iter}}\right)}{FF_n^{\text{iter}}} < 0.15 \\ 1 - \dfrac{F^{\#}\left(\boldsymbol{X}_n'^{\text{iter}}\right)}{FF_n^{\text{iter}}}, & \text{otherwise} \end{cases} \tag{13}$$

$$C_w = F^{\#}\left(\boldsymbol{X}_n^{\text{iter}-1}\right) \frac{C_g}{FF_n^{\text{iter}} - F^{\#}\left(\boldsymbol{X}_n'^{\text{iter}}\right)} \tag{14}$$

$$C_p = F^{\#}\left(\boldsymbol{X}_n^{\text{iter}-1}\right)\left(F^{\#}\left(\boldsymbol{X}_n^{\text{iter}-1}\right) + F^{\#}\left(\boldsymbol{P}\mathbf{best}_n^{\text{iter}-1}\right)\right) \tag{15}$$

It is necessary to note that the formulations of equations (11), (12), (14) and (15) used in the proposed scheme are quite similar to those of the SPC1 of [24, 26]. Moreover, according to the formulation of SPC2 of [24, 26], we apply the corresponding equation of SPC2 for (13).

3.2 Enhanced SSOA (ESSOA)

For increasing the SSOA ability in finding the global or near-global optima, a powerful probabilistic approach is used in this paper. The principle of designing a probabilistic framework is to enhance the performance or robustness of SSOA to obtain good results in the numerical optimisation with effective yet, diverse characteristics. The basic idea is to select adaptively the presented three effective mutation strategies based on their previous experiences of generating promising solutions and applied to achieve the mutation operation. It means that at different iterations of the optimisation procedure, multiple mutation strategies may be assigned a different probability based on their success rate in generating improved solutions within a certain number of previous generations. In order to improve the solutions of the proposed problem and further increase the population diversity and enhance the globally search capabilities, the below mutation methods are implemented as follows:

nth particle:	$x_{n,1}$	$x_{n,2}$	$\cdots$	$x_{n,i}$	$\cdots$	$x_{n,NG}$

Fig. 2 *Structure of each particle*

Method 1

$$\begin{aligned} \boldsymbol{X}_{n,\text{method1}}^{\text{iter}} &= \boldsymbol{X}_n^{\text{iter}} \\ &+ \text{rand}(\cdot)\big(\boldsymbol{G}\mathbf{best}^{\text{iter}} - \mathbf{Learning\ Factor}^{\text{iter}}\mathbf{Mean}^{\text{iter}}\big) \\ n &= 1, \ldots, N_1 \end{aligned} \tag{16}$$

Method 2

$$\begin{aligned} \boldsymbol{X}_{n,\text{method2}}^{\text{iter}} &= \boldsymbol{X}_n^{\text{iter}} + \text{rand1}(\cdot)\big(\boldsymbol{G}\mathbf{best}^{\text{iter}} - \boldsymbol{X}_n^{\text{iter}}\big) \\ &+ \text{rand2}(\cdot)\big(\boldsymbol{G}\mathbf{best}^{\text{iter}} - \mathbf{Worst}^{\text{iter}}\big), \\ n &= 1, \ldots, N_2 \end{aligned} \tag{17}$$

Method 3 (see (18))

In order to produce the trial solutions for each particle n, two other particles are randomly chosen from the population as $r_1 \neq r_2 \neq n$. It means that in each iterate of the EGSSOA, for each particle of the population which select mutation Method 3, two vectors r_1 and r_2 are selected randomly from the existing population in order to cover the algorithm searching domain uniformly. **Learning Factor**$^{\text{iter}}$ is typically expressed as: $\mathbf{Learning\ Factor}^{\text{iter}} = \big[\text{Learning Factor}_1^{\text{iter}}, \text{Learning Factor}_2^{\text{iter}}, \ldots, \text{Learning Factor}_{NG}^{\text{iter}}\big]$ and $\text{Learning Factor}^{\text{iter}} = \text{round}(1 + \text{rand}(\cdot))$. In this study, it is proposed as $\text{Learning Factor}_i^{\text{iter}} = \text{Mean}_i^{\text{iter}}/\text{Gbest}_i^{\text{iter}}$. The accomplishment of this type of learning factor is moderately due to its natural ability of processing a population of potential solutions, which allows them to utilise an expanding exploration or diversification and exploitation or intensification in the search space of the optimisation problem. In the optimisation procedure, the lower value of this metric allows the superior search process in the first iteration of the algorithm but causes slow convergence. A larger value of it accelerates the search process, but it reduces the exploration capability [27].

The occurrence of mutation operator is followed from the requirements of the ESSOA search process. All the particles in the population will have a chance to improve, controlled by the probability of their methods of mutations. Based on a probability model, each particle selects one of these three methods. Denote $\text{Prob}_{\text{method}a}^{1} = 1/3, \quad a = 1, \ldots, 3$ as the initial probability of implementing ath mutation strategy. $\text{Prob}_{\text{method}a}^{\text{iter}}$ is updated after the S iterations according to the following form

$$\text{Prob}_{\text{method}a}^{\text{iter}} = \frac{\text{Success Rate}_{\text{method}a}^{\text{iter}}}{\sum_{a=1}^{3} \text{Success Rate}_{\text{method}a}^{\text{iter}}}; \quad a = 1, \ldots, 3 \tag{19}$$

where the $\text{Success Rate}_{\text{method}a}^{\text{iter}}$ represents the success rate of the trial solutions generated by the ath mutation strategy and successfully entering the next step within the previous S iterations with respect to the iter-th iteration. Thus, the $\text{Success Rate}_{\text{method}a}^{\text{iter}}$ can be formulated as follows

$$\begin{aligned} \text{Success Rate}_{\text{method}a}^{\text{iter}} &= \frac{\sum_{\text{it}=\text{iter}-S}^{\text{iter}-1} \text{NS}_{\text{method}a}^{\text{it}}}{\sum_{\text{it}=\text{iter}-S}^{\text{iter}-1} \big(\text{NS}_{\text{method}a}^{\text{it}} + \text{NF}_{\text{method}a}^{\text{it}}\big)} \\ &+ \varepsilon; \quad a = 1, \ldots, 3; \quad \text{iter} > S \end{aligned} \tag{20}$$

where $\text{NS}_{\text{method}a}^{\text{it}}$ and $\text{NF}_{\text{method}a}^{\text{it}}$ are the respective numbers of the trial solutions generated by the ath mutation strategy, which remain or fail in the selection process in the last S iterations. The small constant value $\varepsilon = 0.01$ is used to avoid the possible null values for $\text{Success Rate}_{\text{method}a}^{\text{iter}}$. In order to ensure that the summing of the probabilities of choosing strategies is always equal to one, (19) can be used. It can be expected that the larger the value of the $\text{Success Rate}_{\text{method}a}^{\text{iter}}$, the larger is the probability of implementing it to generate the trial solutions at the current iteration k. To this end, the Roulette Wheel Mechanism (RWM) selection method is applied to choose the ath modification strategy for each particle.

According to the above discussion, the trial solution $\boldsymbol{X}_{n,\text{mut}}^{\text{iter}}$ of the ESSOA can be generated as follows

$$\boldsymbol{X}_{n,\text{mut}}^{\text{iter}} = \begin{cases} \boldsymbol{X}_{n,\text{method1}}^{\text{iter}}, & \text{if } \text{rand}(\cdot) < \text{Prob}_{\text{method1}}^{\text{iter}} \\ \boldsymbol{X}_{n,\text{method2}}^{\text{iter}}, & \text{else if } \text{rand}(\cdot) < \text{Prob}_{\text{method1}}^{\text{iter}} + \text{Prob}_{\text{method2}}^{\text{iter}} \\ \boldsymbol{X}_{n,\text{method3}}^{\text{iter}}, & \text{else} \end{cases} \tag{21}$$

Equation (22) shows the RWM selection method, which is used to choose the ath method for each of the particle in the population. Afterwards, the mutant vector $\boldsymbol{X}_{n,\text{mut}}^{\text{iter}}$ is mixed with $\boldsymbol{X}_n^{\text{iter}}$, which generates $\boldsymbol{X}_{n,\text{new}}^{\text{iter}}$ as follows

$$x_{n,i,\text{new}}^{\text{iter}} = \begin{cases} x_{n,i,\text{mut}}^{\text{iter}}, & \text{if } (\text{rand}(\cdot) \leq 0.5) \\ x_{n,i}^{\text{iter}}, & \text{otherwise} \end{cases}, \quad i = 1, \ldots, \text{NG}; \quad n = 1, \ldots, \text{NP} \tag{22}$$

where $x_{n,i,\text{new}}^{\text{iter}}$ and $x_{n,i,\text{mut}}^{\text{iter}}$ are the ith member of the nth particle at the iter-th iteration in the new generated vector $\boldsymbol{X}_{n,\text{new}}^{\text{iter}}$ and mutant vector $\boldsymbol{X}_{n,\text{mut}}^{\text{iter}}$, respectively. The new solutions can replace the original solution based on their fitness functions as follows:

$$\boldsymbol{X}_n^{\text{iter}} = \begin{cases} \boldsymbol{X}_{n,\text{new}}^{\text{iter}}, & \text{if } F\big(\boldsymbol{X}_{n,\text{new}}^{\text{iter}}\big) \leq F\big(\boldsymbol{X}_n^{\text{iter}}\big) \\ \boldsymbol{X}_n^{\text{iter}}, & \text{otherwise} \quad n = 1, \ldots, \text{NP} \end{cases} \tag{23}$$

This technique enables the ESSOA to provide better optimal solutions and it also affects the convergence capability of the algorithm significantly.

$$\boldsymbol{X}_{n,\text{method3}}^{\text{iter}} = \begin{cases} \boldsymbol{X}_n^{\text{iter}} + \text{rand}(\cdot)\Big(\boldsymbol{X}_{r_1}^{\text{iter}} - \boldsymbol{X}_{r_2}^{\text{iter}}\Big), & \text{if } F\Big(\boldsymbol{X}_{r_1}^{\text{iter}}\Big) < F\Big(\boldsymbol{X}_{r_2}^{\text{iter}}\Big) \\ \boldsymbol{X}_n^{\text{iter}} + \text{rand}(\cdot)\Big(\boldsymbol{X}_{r_2}^{\text{iter}} - \boldsymbol{X}_{r_1}^{\text{iter}}\Big), & \text{otherwise} \end{cases}, \quad n = 1, \ldots, N_3 \tag{18}$$

doi: 10.1049/iet-gtd.2012.0616

3.3 Gradient-based method (GM)

For increasing the local search capability of the proposed ESSOA around the particles, for each particle the gradient search method is used in several iterations or steps. So, this method concerns to find the better solution for each particle while satisfying all of the presented equality and inequality constraints. In this regard, the GM is applied directly to the Lagrange function. It can be defined as follows [25]

$$L^g = \sum_{i=1}^{NG} F_i\big(P_i^g(t)\big) + \lambda^g \left(P_D(t) + P_{\text{Loss}}^g(t) - \sum_{i=1}^{NG} P_i^g(t) \right) \tag{24}$$

Afterwards, the gradient of this function can be calculated as follows

$$\nabla L^g = \begin{bmatrix} \dfrac{\partial L^g}{\partial P_1^g(t)} \\ \vdots \\ \dfrac{\partial L^g}{\partial P_{NG}^g(t)} \\ \dfrac{\partial L^g}{\partial \lambda^g} \end{bmatrix} = \begin{bmatrix} \dfrac{\mathrm{d}}{\mathrm{d}P_1^g(t)} F_1(P_1^g(t)) + \lambda^g \left(\dfrac{\mathrm{d}}{\mathrm{d}P_1^g(t)} P_{\text{Loss}}^g(t) - 1 \right) \\ \vdots \\ \dfrac{\mathrm{d}}{\mathrm{d}P_{NG}^g(t)} F_{NG}(P_{NG}^g(t)) + \lambda^g \left(\dfrac{\mathrm{d}}{\mathrm{d}P_{NG}^g(t)} P_{\text{Loss}}^g(t) - 1 \right) \\ P_D(t) + P_{\text{Loss}}^g(t) - \sum_{i=1}^{NG} P_i^g(t) \end{bmatrix} \tag{25}$$

It is worthwhile to note that for each generating unit $P_i^g(t)$, the $\mathrm{d}F_i(P_i^g(t))/\mathrm{d}P_i^g(t)$ can be calculated as follows

$$\begin{aligned} &\frac{\mathrm{d}}{\mathrm{d}P_i^g(t)} F_i\big(P_i^g(t)\big) \\ &= b_i + 2c_i P_i^g(t) \\ &\quad + \frac{\big(-e_i d_i \cos\big(e_i\big(P_{i,\min} - P_i^g(t)\big)\big)\big)\big(d_i \sin\big(e_i\big(P_{i,\min} - P_i^g(t)\big)\big)\big)}{\big|d_i \sin\big(e_i\big(P_{i,\min} - P_i^g(t)\big)\big)\big|} \end{aligned} \tag{26}$$

As seen from (26), there may exist a power generating unit that makes the $d_i \sin\big(e_i\big(P_{i,\min} - P_i^g(t)\big)\big)$ to be equal to zero. These points are the non-differentiable points and should be fixed to their generation. It means that they do not change during the GM process. Then, the new particle of GM in iteration g of the GM, that is, $P_1^g(t), \ldots, P_{NG}^g(t)$ and λ^g is computed as follows [25]

$$\begin{bmatrix} P_1^g(t) \\ \vdots \\ P_{NG}^g(t) \\ \lambda^g \end{bmatrix} = \begin{bmatrix} P_1^{g-1}(t) \\ \vdots \\ P_{NG}^{g-1}(t) \\ \lambda^{g-1} \end{bmatrix} + \alpha^g \begin{bmatrix} \Delta P_1^g(t) \\ \vdots \\ \Delta P_{NG}^g(t) \\ \Delta \lambda^g \end{bmatrix} \tag{27}$$

where

$$\begin{bmatrix} \Delta P_1^g(t) \\ \vdots \\ \Delta P_{NG}^g(t) \\ \Delta \lambda^g \end{bmatrix} = m^g \begin{bmatrix} \Delta P_1^{g-1}(t) \\ \vdots \\ \Delta P_{NG}^{g-1}(t) \\ \Delta \lambda^{g-1} \end{bmatrix} + \big(1 - m^g\big) \nabla L^g \tag{28}$$

where m^g and α^g are the parameters of the GM, which is updated in this paper as follows [28] (Fig. 3).

It should be noted that all of the constraints (3)–(9) should be satisfied in each step of the GM. In order to handle these constraints the following steps should be applied:

Step 1: According to the ramping rate limitation and the previous power output data, the generation capacity of each unit is updated. The feasibility of constraints (5)–(8) is checked. If any element of the particle breaks its inequality constraints, then the position of the individual is fixed at its maximum/minimum operating point. Therefore this can be formulated as

$$P_i^g(t) = \begin{cases} P_i^g(t), & \text{if } \underline{P}_i(t) \langle P_i^g(t) \langle \bar{P}_i(t) \\ \underline{P}_i(t), & \text{if } P_i^g(t) \langle \underline{P}_i(t) \\ \bar{P}_i(t), & \text{if } P_i^g(t) \rangle \bar{P}_i(t) \end{cases} \tag{29}$$

If constraint (9) is violated, it means that the power generated output of this unit falls into one of its POZs. If the power generation value is closer to the lower bound of the POZ, the value of it sets to the lower limit, else it sets to the upper limit. In other words, if the generation of unit i is settled in its kth POZ, that is

$$P_{i,k}^L \le P_i^g(t) \le P_{i,k}^U \tag{30}$$

Then the value of the generation output of unit i is cut to the nearest bound of its kth POZ as follows

$$\begin{aligned} P_{i,k}^{\text{ave}} &= \frac{P_{i,k}^L + P_{i,k}^U}{2} \to P_i^g(t) \\ &= \begin{cases} P_{i,k}^L & \text{if } P_{i,k}^L < P_i^g(t) \le P_{i,k}^{\text{ave}} \\ P_{i,k}^U & \text{if } P_{i,k}^{\text{ave}} < P_i^g(t) \le P_{i,k}^U \end{cases} \end{aligned} \tag{31}$$

Step 2: Power balance handling: For satisfying the constraint (3), the values of power mismatch is calculated as follows

$$P\_\text{violate}_g(t) = \sum_{i=1}^{NG} P_i^g(t) - P_D(t) - P_{\text{Loss}}^g(t) \tag{32}$$

If $P\_\text{violate}_g(t) = 0$, return.

If $L^g < L^{g-1}$

$m^g = 0.8; \alpha^g = 1.1\alpha^{g-1}$

Else

$m^g = 0; \alpha^g = 0.9\alpha^{g-1}$

End

Fig. 3 *Parameters updating of GM*

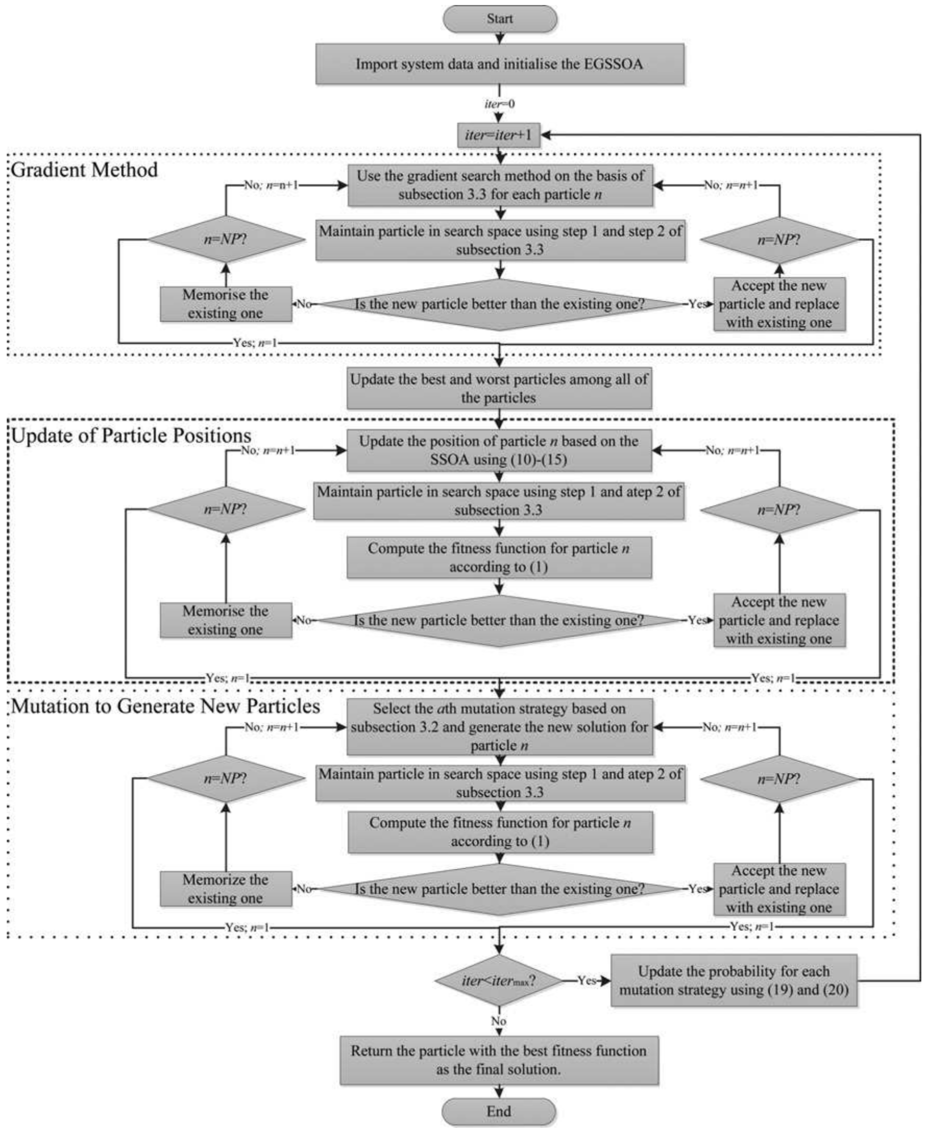


Fig. 4 *Flowchart of the proposed EGSSOA*

If $P\_\text{violate}_g(t) \neq 0$, choose one unit $P_i^g(t)$ randomly as the slack unit and subtract $P\_\text{violate}_g(t)$ from it. If any unit of the individual violates (5)–(8), then the position of the unit output power is fixed to its minimum/maximum power output. In addition, if any unit of the individual violates (9), then the position of the unit output power is fixed to its lower/upper limit of the corresponding POZ. This procedure proceeds to reach the zero value of $P\_\text{violate}_g(t)$ and to mend the power mismatch it has been guaranteed that different units will be chosen. If all the units are checked and no one can cover the $P\_\text{violate}_g(t)$ to meet the power balance, then two units will be selected as slack unit and share the $P\_\text{violate}_g(t)$, and so on.

4 Application of EGSSOA to the ED problem

In order to better and clearly illustrate the structure of the proposed approach, its flowchart is given in Fig. 4. It is clear that in the iteration procedure of the EGSSOA, the SSOA updating mechanism and probabilistic mutation framework, which constitute the ESSOA, are applied on the population, consecutively. Moreover, in order to increase the local search capability of the proposed ESSOA around the particles, for each particle the gradient search method is used in several steps.

In the rest part of this section, an application of the EGSSOA for solving the ED problem is described in details as follows:

Step 1: Enter input date.

In this step, the input data containing the generator data, that is, fuel cost coefficients, up and down ramp-rate limits, generation capacity, POZs of each unit, load demand, and B-loss coefficients should be defined.

Step 2: Generate the initial population.

An initial population $\boldsymbol{X}_n$; $n = 1, \ldots,$ NP in the problem search space that must meet constraints is generated randomly.

Step 3: Apply the following steps for all of the particles in the population. Set the counter iter equal to one.

Step 4: Use the GM for each particle of the initial population in order to search locally and obtain better solution. The new generated particles in the steps of the GM must satisfy the equality and inequality constraints (3)–(9). To handle these constraints, Steps 1 and 2 of Section 3.3 can be applied.

Step 5: Calculate the objective functions of each individual.

Step 6: Determine the best and worst solutions among the population.

Step 7: Implement the movement procedure based on ESSOA according to Sections 1 and 2.

Step 8: Compare the fitness function, that is, the fuel cost. Compare the fitness function of the new solution with previous individuals and select the better individuals for being participated in the next iteration.

Step 9: Update the probability for each mutation strategy, that is, by using (19) and (20).

Step 10: Set iter = iter + 1. Select the best solution. If the current iteration number obtains the preordained maximum iteration number, the algorithm is stopped, otherwise go to Step 6.

5 Simulation results

5.1 Validation of EGSSOA

In order to determine the performance of the proposed EGSSOA regarding the rest, different algorithms are applied to the four renowned benchmark functions in this subsection. The tested benchmarks are chosen so that they will provide different features like separability and multimodality. In definition, a function with several local optima in the search space is called multimodal. Also, a function is defined as separable if it can be computed by dividing it to the sum of different parts. Among these types of functions, the non-separable functions are more complicated to optimise while the complexity increases if it is also multimodal. Meanwhile, by increasing the number of dimensions and also the randomness of the local optimal solutions, the complexity of the function increases consequently. Therefore, in order to see the performance of the suggested EGSSOA, this paper knowledgeably takes advantage of four different benchmarks with different features as the test functions. The detailed data of the benchmark problems are tabulated in Table 1. In this validation, the mean value (Mean) and the standard deviation (SD) metric are utilised for comparison with the GA, PSO and SSOA.

The Mean metric describes the average ability of the algorithm for reaching the global optima and the SD defines the variation from the Mean. It is vital to note that the number of run terms for the algorithms depends on the number of times that objective function is evaluated. The maximum number of function evaluations for PSO, GA and SSOA is determined as 50 000, whereas for the proposed EGSSOA it is supposed to be more lower than 50 000, that is, 30 000. In the experimental studies of EGSSOA, we set the $g_{\max}$, NP, S, and α^0 to 10, 10, 5 and 0.01, for all of the benchmark functions, respectively. For all tested benchmarks, the algorithms carry out 30 independent runs. In order to see the performance of the projected algorithm with regard to the dimensionality (D) of the benchmarks, the $D = 30$ and $D = 100$ are evaluated and shown in Table 2. From this table, it can be observed that the EGSSOA outperforms other algorithms from both the Mean and SD metric. Also, since by increasing D of the tested

Table 1 Detailed data of the benchmark functions

F_i	Function	Equation	Domain	$X_{\max}$	Global optimum
F_1	*generalised Rosenbrock*	$F_1(\boldsymbol{X}) = \sum_{i=1}^{D-1}\left(100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right); \quad \boldsymbol{X} = [x_1, \ldots, x_D]$	[−30,30]	30	$F_{1,\min} = 0$, $X = [1,\ldots,1]$
F_2	*generalised Rastrigin*	$F_2(\boldsymbol{X}) = \sum_{i=1}^{D}\left(x_i^2 - 10\cos(2\pi x_i) + 10\right); \quad \boldsymbol{X} = [x_1, \ldots, x_D]$	[−5.12,5.12]	5.12	$F_{2,\min} = 0$, $X = [0,\ldots,0]$
F_3	*generalised Griewank*	$F_3(\boldsymbol{X}) = \frac{1}{4000}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1; \quad \boldsymbol{X} = [x_1, \ldots, x_D]$	[−600 600]	600	$F_{3,\min} = 0$, $X = [0,\ldots,0]$
F_4	*generalised Ackley*	$F_4(\boldsymbol{X}) = -20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)\right) + 20 + \exp(1); \quad \boldsymbol{X} = [x_1, \ldots, x_D]$	[−32,32]	32	$F_{4,\min} = 0$, $X = [0,\ldots,0]$

Table 2 Values of mean and SD for GA, PSO, SSOA and proposed EGSSOA

Function	D	GA		PSO		SSOA		Proposed EGSSOA	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD
F_1	30	3.39 e + 02	3.61 e + 02	3.74 e + 01	3.21 e + 01	3.31 e + 01	1.13 e + 01	0.00 e + 00	0.00 e + 00
	100	2.01 e + 08	2.97 e + 07	1.42 e + 07	2.14 e + 06	6.39 e + 02	8.17e + 02	0.00 e + 00	0.00 e + 00
F_2	30	6.50 e − 01	3.59 e − 01	2.08 e + 01	5.94 e + 00	7.37 e − 01	6.98 e − 01	0.00 e + 00	0.00 e + 00
	100	4.09 e + 03	6.67 e + 01	6.43 e + 02	4.50 e + 01	3.12 e + 02	2.30 e + 01	0.00 e + 00	0.00 e + 00
F_3	30	1.00 e + 00	6.75 e − 02	2.32 e − 01	4.43 e − 01	1.40 e − 01	3.53 e − 01	0.00 e + 00	0.00 e + 00
	100	2.93 e + 01	7.21 e − 01	1.87 e + 01	8.48 e − 01	1.30 e + 01	4.88 e − 01	0.00 e + 00	0.00 e + 00
F_4	30	1.00 e + 00	6.75 e − 02	2.32 e − 01	4.43 e − 01	1.40 e − 01	3.53 e − 01	0.00 e + 00	0.00 e + 00
	100	3.41 e + 02	2.47 e + 01	7.91 e + 02	7.94 e + 01	1.76 e + 02	1.96 e + 01	0.00 e + 00	0.00 e + 00

doi: 10.1049/iet-gtd.2012.0616

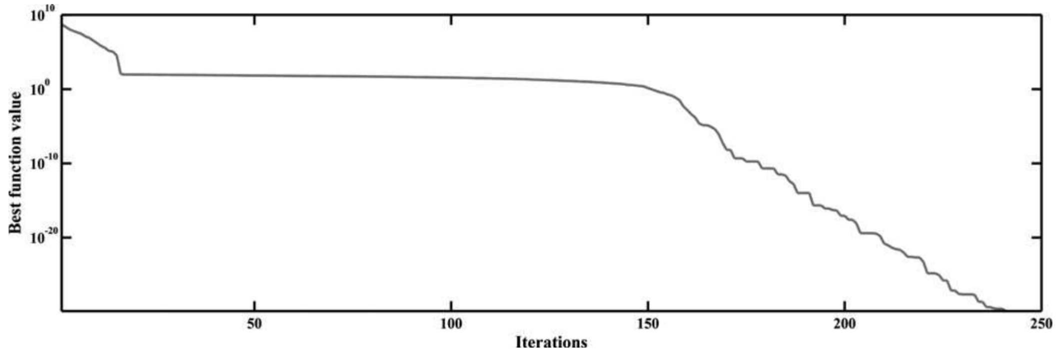


Fig. 5 *Convergence characteristic of the proposed EGSSOA for the first benchmark function*

benchmark, the obtained results by EGSSOA have no changes, then the consistency, robustness and optimality of the proposed algorithm can be concluded. Moreover, the convergence characteristic of the best fitness function with regard to the iterations obtained by the proposed EGSSOA for the first complicated test function is depicted in Fig. 5. This function can be treated as a multimodal problem. It has a narrow valley from the perceived local optima to the global optimum. It will be hard to find the global optimum if many particles fall into one of these local optima. However, Fig. 5 proves the ability of proposed algorithm in finding the global optima once again.

5.2 Test systems and parameters setting

In order to assess the effectiveness and efficiency of the proposed EGSSOA to solve the ED problems, it is tested with four test systems that have different number of generating units. In these test systems, various combinations of the ED settings like with/without valve-point effect, with/without POZs, with/without multi-fuel type of generating units, with/without losses and with/without ramping rate limitations are considered. The following test systems are studied, analysed and compared with other algorithms in the area of the ED problems:

- The first test system includes 10 generating units. In this test system, two different scenarios are studied: (1) Considering multi-fuel constraints as well as valve-point effects and (2) Considering multi-fuel constraints and neglecting valve point effects. The data for this system are adapted from [29].
- The second test system consists of 15 units. In this case, POZs and transmission losses are studied under two different conditions: (1) Considering ramp rate limits and (2) Neglecting this limitation. The data for this case is taken from [30].
- The third test case contains of 40 generation units. Here, the valve-point effects are considered under the following two scenarios: (1) without POZs and (2) with POZs. This is a complicated non-convex and non-smooth test case that has many local minima and the global minimum is very difficult to find. It is noted that considering POZs in this complicated test system requires a strength and powerful solution method to obtain global or near-global optima. The data for this test system is taken from [31].
- From the point of view of meta-heuristic optimisation and evolutionary computation, a test system of 10, 15 or 40 units is not large enough to display the scalability of the suggested approach. Thus, a power system of 80 generating units with valve point loading effects is considered as test system 4. This test system is generated by duplicating the previous case study.

All the programs are developed using FORTRAN Power Station 4.0 on a Pentium IV personal computer with 2.4 GHz speed processor and 2 GB RAM. In the experimental studies, different parameters including $\text{iter}_{\max}$, $g_{\max}$, NP, S, α^0 and λ^0 are used for the four test systems, which are listed in Table 3. For all test systems, the algorithms carry out 30 independent runs.

5.3 Performance criterion

In order to evaluate the performance of the suggested EGSSOA, five performance criteria are selected as follows:

5.3.1 Best, mean and worst solutions: The statistical analysis like the best, mean and worst solutions are extracted as single characteristics of a 30 independent runs for fuel cost function. They provide a scheme about concentration of the values in the central part of the distribution. If the best, mean and worst solutions be closed or similar to each other, it means that the proposed algorithm has a robust characteristic in obtaining reliable output.

5.3.2 Error: The error of a solution is corresponded to the difference between the global or near-global solution (the best optimal solution found until now) and the obtained solution. The best value for error factor is zero which

Table 3 Simulation parameters for different test systems

Parameters	Test system 1	Test system 2	Test system 3	Test system 4
$\text{iter}_{\max}$	8	13	30	60
$g_{\max}$	4	8	20	40
NP	30	60	100	200
S	2	3	5	10
α^0	0.001	0.01	0.01	0.01
λ^0	10	15	8	14

IET Gener. Transm. Distrib., 2013, Vol. 7, Iss. 6, pp. 620–635
doi: 10.1049/iet-gtd.2012.0616

Table 4 Compared results of different methods for 10-unit test system (first scenario)

Methods	CQGSO [14]		IPSO [13]		DHS [15]		CIHBMO [32]		EGSSOA	
Unit	Gen	Fuel type	Gen	Fuel type	Gen	Fuel type	Gen	Fuel type	Gen	Fuel type
1	218.59	2	217.5692	2	218.5940	2	218.1050	2	218.1050	2
2	211.4	1	211.2166	1	211.7117	1	211.6596	1	211.6596	1
3	280.66	1	279.6488	1	280.6571	1	280.6571	1	280.6571	1
4	239.64	3	240.1769	3	239.6394	3	239.6864	3	239.6864	3
5	279.93	1	276.5743	1	279.9348	1	279.9344	1	279.9343	1
6	239.24	3	239.9082	3	239.6394	3	239.6610	3	239.6610	3
7	287.73	1	285.3796	1	287.7275	1	287.7275	1	287.7275	1
8	239.51	3	240.4456	3	239.6394	3	239.5520	3	239.5520	3
9	427.42	3	430.0665	3	426.5879	3	427.1485	3	427.1485	3
10	275.87	1	279.0143	1	275.8686	1	275.8687	1	275.8686	1
per-unit CPU time, s	8.76		NR		0.10		2.50		0.013	
total power to satisfy $P_D = 2700$, MW	2699.9900		2700.0000		2699.9998		2700.0002		2700.0000	
fuel cost from generation, $	624.4242		624.5684		624.5201		623.8230		623.8229	
reported fuel cost, $	623.8276		623.8730		623.8266		623.5960		623.8229	

NR: Not reported in the literature

Table 5 Compared results of different methods for 10-unit test system (second scenario)

Optimisation technique	EALHN [21]		IPSO [13]		ALHN [2]		EGSSOA	
Generating unit	Gen	Fuel type	Gen	Fuel type	Gen	Fuel type	Gen	Fuel type
1	218.2502	2	218.2345	2	218.250	2	218.2499	2
2	211.6627	1	211.7060	1	211.663	1	211.6626	1
3	280.7230	1	280.7208	1	280.723	1	280.7228	1
4	239.6316	3	239.6091	3	239.632	3	239.6315	3
5	278.4975	1	278.4580	1	278.498	1	278.4973	1
6	239.6316	3	239.6704	3	239.632	3	239.6315	3
7	288.5847	1	288.6079	1	288.585	1	288.5845	1
8	239.6316	3	239.6341	3	239.632	3	239.6316	3
9	428.5203	3	428.4787	3	428.518	3	428.5216	3
10	274.8671	1	274.8798	1	274.866	1	274.8667	1
per-unit CPU time, s	0.01		1.17		0.05		0.008	
total power to satisfy $P_D = 2700$, MW	2700.0003		2699.9993		2699.9990		2700.0000	
fuel cost from generation, $	623.8093		623.8089		623.8090		**623.8092**	
reported fuel cost, $	623.8090		623.8089		623.8086		**623.8092**	

means the ability of each algorithm to reach the global or near-global optima.

5.3.3 Successful percentage: The successful percentage is defined as the number of successful runs that converge to the best solution divided by all runs (30 runs).

5.3.4 Convergence graph: The convergence graph is also plotted in this paper to examine the quality of the best solution over the evolution procedure.

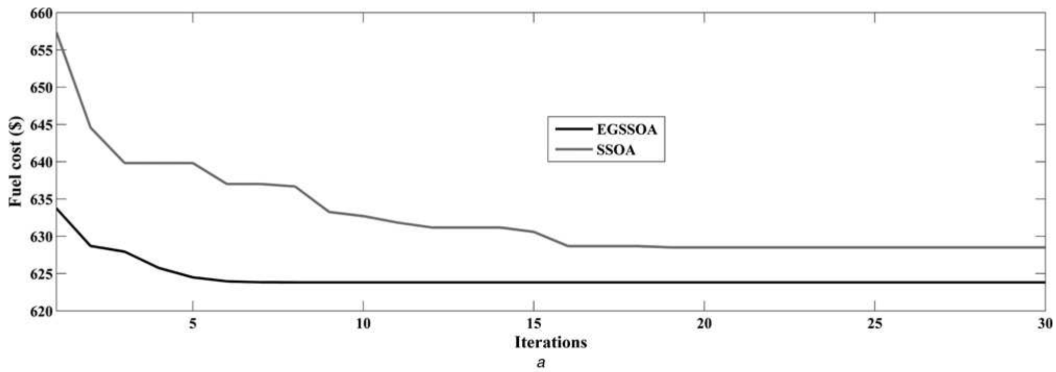
5.3.5 Per-unit CPU time: Since the average CPU execution time is very dependent on the characteristic of the computer systems implemented for other algorithm in other literature, it cannot be directly compared for different methods. For this reason, the per-unit CPU time is used in this paper for each of the mentioned techniques. The per-unit base speed is 2.4 GHz and the per-unit CPU time is calculated as follows:

$$\text{per-unit CPU time} = \frac{\text{given CPU speed}}{2.4\,\text{GHz}} \times \text{given average CPU time} \qquad (33)$$

By comparing the per-unit CPU time of the proposed approach with different techniques in other literature, it can be concluded that the suggested algorithm is faster than other methods. Moreover, since MATLAB software is not fast enough to handle this kind of optimisation problems, so most of researchers are using other software, for example, Visual Studio [15] for this purpose. In this study, the Fortran Power Station 4.0 software is used to implement the proposed method. This programming language is the fastest high-level programming language and is very faster than MATLAB, which is used to implement the ED solution methodology in some papers, for example, [17–21].

5.4 Test system 1: 10-units

For the sake of comparison and to validate the performance of the proposed EGSSOA algorithm, two different scenarios are considered and the obtained results are compared with those in the literatures. It can be seen that the proposed approach can obtain better values with respect to other algorithms. This statement demonstrates the potential and effectiveness of the proposed approach in solving the optimisation problem. It is worthwhile to note that due to integrating



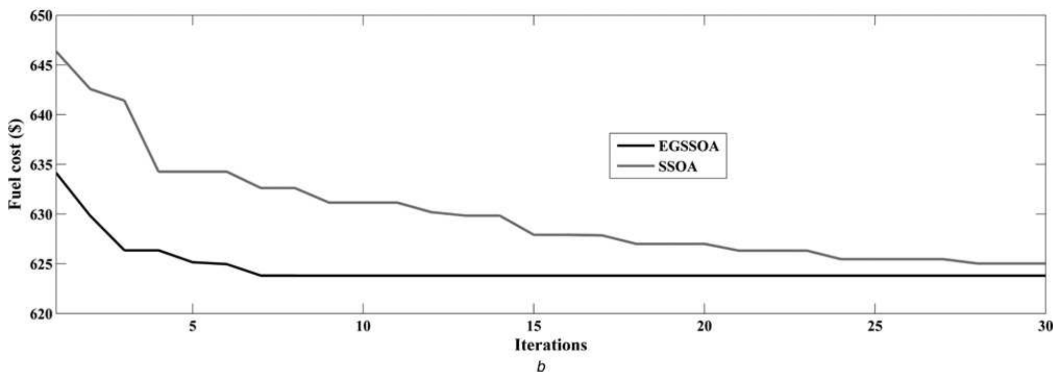


Fig. 6 *Convergence characteristics of EGSSOA and SSOA for test system 1*

a First scenario
b Second scenario

the mathematical-based and heuristic-based optimisation techniques, the proposed algorithm can obtain better results. Hence, the proposed EGSSOA can benefit from both local and global search capabilities and so it can find global or near-global optimum solution.

In order to show that all generation unit outputs are remained within their permissible limits, Tables 4 and 5 are provided for the first and second scenarios, respectively. For

Table 6 Best dispatch found by proposed EGSSOA for the first and second scenario of test system 2

Units, MW	First scenario	Second scenario
P_1	455.0000	455.0000
P_2	455.0000	380.0000
P_3	130.0000	130.0000
P_4	130.0000	130.0000
P_5	230.5328	170.0000
P_6	460.0000	460.0000
P_7	465.0000	430.0000
P_8	60.0000	60.0000
P_9	25.0000	48.9629
P_{10}	35.5819	160.0000
P_{11}	74.8908	80.0000
P_{12}	79.9994	80.0000
P_{13}	25.0000	25.0000
P_{14}	15.0000	33.0491
P_{15}	15.0000	15.0000
P_{Loss}, MW	26.0048	27.0120
fuel cost, $	32 540.0900	32 680.1038

Table 7 Results obtained by different methods for the first and second scenarios of test system 2

Solution technique	Fuel cost, $			Per-unit CPU time, s
	Best value	Mean value	Worst value	
First scenario				
NAPSO [33]	32 548.5859	32 548.5869	32 548.5904	6
FAPSO [33]	32 659.7940	32 663.1900	32 676.0700	12
CSO [34]	32 588.9189	32 679.8775	32 796.7792	NR
DHS [15]	32 588.8454	32 588.8653	32 588.8775	0.30
GM		33 078.5572		0.10
SSOA	32 731.6903	32 845.5416	32 930.9734	0.55
proposed EGSSOA		32 540.0900		0.23
Second scenario				
FA [11]	32 704.4501	32 856.1000	33 175.0000	NR
SOH-PSO [30]	32 751	32 878	32 945	NR
BF [35]	32 784.5000	32 796.8000	NR	NR
PSO [36]	32 858	33 039	33 331	NR
GA [36]	33 113	33 228	33 337	NR
GAAPI [1]	32 732.9500	32 735.0600	32 756.0100	NR
GM		33 569.0512		0.15
SSOA	32 819.1554	32902.1631	32 999.4008	0.62
proposed EGSSOA		32 680.1038		0.27

NR: Not reported in the literature

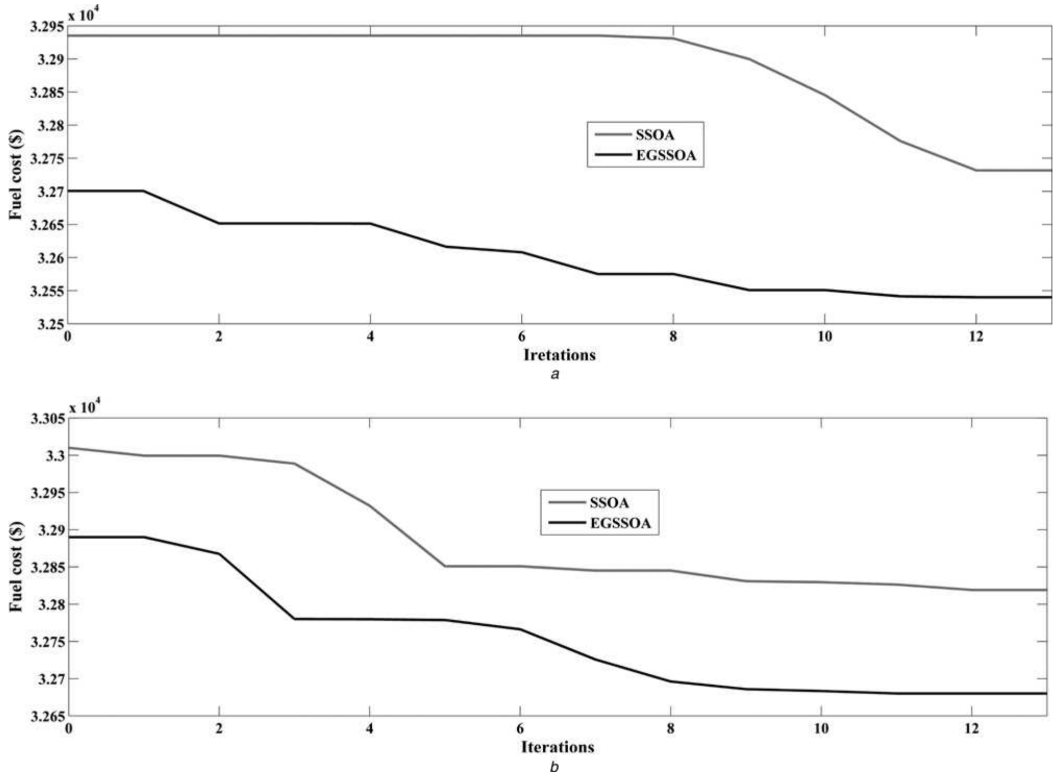


Fig. 7 *Convergence characteristics of EGSSOA and SSOA for test system 2*
a First scenario
b Second scenario

the first scenario of this test system where the valve-point effects are also included, the optimal value of fuel cost using EGSSOA is $623.8229. It is evident that considering the non-smooth nature of the valve-point effects for generating units changes the control variables, and especially, increases the fuel costs. The total reported cost, total cost from generation, total power generation and per-unit CPU time of the proposed algorithm and other algorithms are also shown in the last rows of Tables 4 and 5. According to these tables, there are some inconsistencies and inaccuracies in the reported results in other optimisation techniques of other literatures. It is clear that the reported total fuel cost in [2, 13–15, 20, 21, 32] is less than the total fuel costs from the generation schedule in all of the referred paper. It means that because of the enormous input data for this test system, many authors have been mistaken. On the other hand, the proposed approach can reach the feasible accurate solution. Moreover, it is necessary to note that the lower values of fuel cost for IPSO [13] and ALHN [18] for test system 1 in the second scenario are due to power mismatch with the amount of (P_violate = 2700–2699.9993 = 0.0007 MW) and (P_violate = 2700–2699.9993 = 0.0010 MW). So, these references are wrong and should be discarded from the comparison process.

The convergence characteristics of this test system with EGSSOA and original SSOA for the first and second scenarios are shown in Figs. 6*a* and *b*, respectively. From this figure, we can see that the proposed method can reduce the chance of getting stuck in local minima and result in better performance.

5.5 Test system 2: 15-units

In order to study the effectiveness and superiority of the proposed optimisation method, the ED is implemented. In the 15-unit test system, first, the ramp rate limits are neglected and the best fuel cost and its power losses using EGSSOA are $32 540.0900 and 26.0048 MW, respectively. The provided loss is smaller than the recently reported literature [33], that is, 26.8959. The detailed results of the best solution of EGSSOA are given in Table 6 to check whether the constraints of the problem are satisfied or not. Secondly, the effect of the ramp rate limits is studied and the ED problem is evaluated. The fuel cost and power losses of the best dispatch result of the test obtained by the proposed EGSSOA are $32 680.1038 and 27.0120 MW, respectively. The corresponding loss for this test system is greatly smaller than the transmission losses of recently reported approach FA [11], that is, 30.6614 MW. The detailed results of the best solution of EGSSOA considering ramping rate limitations is shown in Table 6.

The complete comparison of the results and performance of EGSSOA with respect to the SSOA and GM for two different scenarios is given in Table 7. It can be seen from the simulation results that the proposed method reveals better solutions compared to other methods and outperforms them.

Table 8 Results obtained by different methods for the first and second scenarios of test system 3

Solution technique	Fuel cost, $			Per-unit CPU time, s
	Best value	Mean value	Worst value	
First scenario				
FCASO [17]	121 516.4700	122 082.5900	NR	120.9833
IPSO-TVAC [12]	121 412.5450	121 419.3000	121 423.8000	NR
CQGSO [14]	121 412.5512	121 423.5200	121 438.6850	8.5500
FA [11]	121 415.0500	121 416.5700	121 424.5600	NR
MGSO [10]	121 412.5693	NR	NR	NR
aBBOmDE [22]	121 414.8734	121 487.8532	121 568.3254	NR
SOH-PSO [30]	121 501.1400	122 446.3000	121 853.6900	NR
BF [35]	121 423.6379	121 814.9465	NR	NR
NAPSO [33]	121 412.6102	121 412.7373	121 412.9109	15.8750
GM		124 590.4610		0.22
SSOA	123 650.1554	123 963.0471	124 433. 5164	2.49
proposed EGSSOA		121 412.5355		1.14
Second scenario				
NAPSO [33]	121 491.0662	121 491.2756	121 491.5261	NR
FAPSO [33]	122 261.3706	122 471.0751	122 597.5196	NR
PSO [33]	124 875.8523	125 162.7011	125 368.9204	NR
GM		127 183.3546		0.38
SSOA	124 504.2505	124 803.7875	125 303.4571	3.12
SSOA with mutation method 3	121 876.8734	122 042.7218	122 339.1845	2.96
SSOA with mutation method 1	121 514.6639	121 604.8746	121 734.4739	2.50
SSOA with mutation method 2	121 490.6623	121 492.0769	121 494. 0692	2.37
Proposed EGSSOA		121 487.7650		1.35

NR, Not reported in the literature

Table 9 Best dispatch found by proposed EGSSOA for the first and second scenario of test system 3

Units, MW	First scenario	Second scenario
P_1	110.799825	110.799824
P_2	110.799825	110.799825
P_3	97.399913	97.399912
P_4	179.733100	179.733100
P_5	87.799905	87.799904
P_6	140	140
P_7	259.599650	259.599650
P_8	284.599650	284.599649
P_9	284.599650	284.599650
P_{10}	130	130
P_{11}	94	168.799822
P_{12}	94	168.037472
P_{13}	214.759790	125
P_{14}	394.279370	400
P_{15}	394.279370	394.279369
P_{16}	394.279370	394.279370
P_{17}	489.279370	489.279370
P_{18}	489.279370	489.279370
P_{19}	511.279370	511.279371
P_{20}	511.279370	511.279370
P_{21}	523.279370	523.279370
P_{22}	523.279370	523.279370
P_{23}	523.279370	523.279370
P_{24}	523.279370	523.279370
P_{25}	523.279370	523.279370
P_{26}	523.279370	523.279370
P_{27}	10	10
P_{28}	10	10
P_{29}	10	10
P_{30}	87.799905	87.799904
P_{31}	190	190
P_{32}	190	190
P_{33}	190	190
P_{34}	164.799825	164.799825
P_{35}	200	164.799825
P_{36}	194.397778	164.799825
P_{37}	110	110
P_{38}	110	110
P_{39}	110	110
P_{40}	511.279370	511.279371
fuel cost, $	121 412.535519	121 487.765035

Also, Figs. 7*a* and *b* show the variation in the best solution of fuel cost considering and neglecting ramp rate limits with the number of iterations in the population during search procedure for SSOA and EGSSOA. It should be mentioned that in each generation, the EGSSOA can produce diverse solution even with a small population and less maximum iteration number. Besides, the EGSSOA with the probabilistic mutation strategy and GM can better manage transition from each generation to the next one in comparison with SSOA. One main disadvantage of SSOA with respect to the proposed method for solving the ED problems is their slow and premature convergence to a near-optimal solution.

5.6 Test system 3: 40-units

As mentioned, two different scenarios, that is, neglecting and considering POZs are considered in this test system to show the effectiveness of the proposed EGSSOA in handling complicated ED problems. It has to be noted that the POZs are embedded in five units 10, 11, 12, 13 and 14. This problem exhibits mainly challenging because these POZs cause in a non-convex decision space. This challenging problem not only needs the appropriate handling of the constraints, but also employs a powerful search in different sub-regions without idling too much time on the POZs. This means that the parameters and modifications which are used in the EGSSOA should be sufficient enough so that they can distribute the population of particles in most promising regions in the search space where they can also move from the local solutions to other regions when necessary. A complete comparison of the results and performance of EGSSOA with respect to the SSOA with mutation strategy 1 (uses mutation method 1 (16) for all the extracted solutions of the SSOA), SSOA with mutation strategy 2 (uses mutation method 2 (17) for all the extracted solutions of the SSOA), SSOA with mutation strategy 3 (uses mutation method 3 (18) for all the

doi: 10.1049/iet-gtd.2012.0616

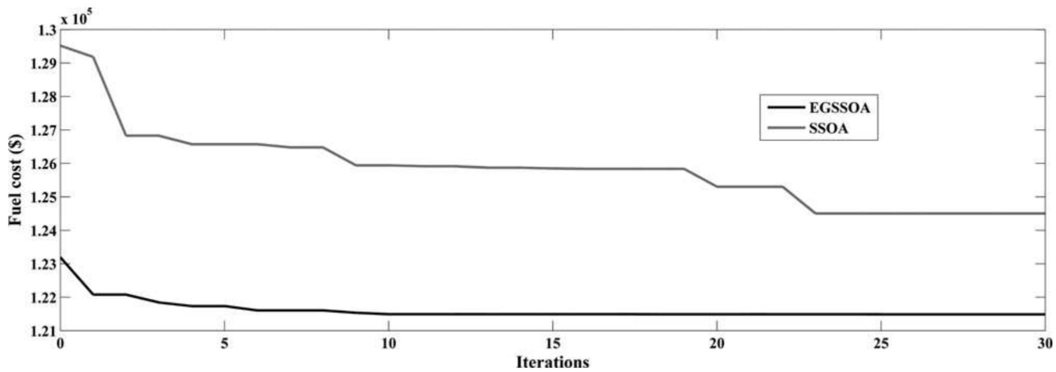


Fig. 8 *Convergence characteristics of EGSSOA and SSOA for the second scenario of test system 3*

extracted solutions of the SSOA), original SSOA, GM and those of the other known methods in two different scenarios is given in Table 8. The probabilistic characteristics of the proposed mutation framework are analysed using the candidate strategies in the pool separately to solve the problem. It is necessary to note that in each separate method, that is, SSOA with mutation strategy 1, SSOA with mutation strategy 2 and SSOA with mutation strategy 3, mutation techniques 1, 2 and 3 are implemented for all of the output particles of the SSOA updating mechanism, respectively.

It should be stressed that the worst value of the fuel cost obtained by the proposed method is better than the best solutions of all other methods in two scenarios. By comparison of the best values in this table, the effectiveness of the proposed method is clearly specified. Also, comparing the mean values demonstrates the EGSSOA's better convergence characteristics. The optimum scheduling of generating units using the proposed method for the first and second scenarios are given in Table 9.

Moreover, in this study, the error analysis is implemented to show the robustness of the proposed approach. The corresponding errors of the GM, original SSOA, FCASO [17], SOH-PSO [30] BF [35], FA [11], mBBOaDE [22], NAPSO [33], MGSO [10], CQGSO [14], IPSO-TVAC [12] and EGSSOA are $3,177.9255, $2,237.6199, $103.9345, $88.6045, $11.1024, $2.5145, $2.3379, $0.0747, $0.00338, $0.0157, $0.0095 and $0.0000 for the first scenario. The same justification can be applied to other test cases. In addition, the convergence characteristic of the second scenario is depicted in Fig. 8 for demonstrating the speed quality of the proposed algorithm in searching the global or near-global optima.

The following objects can be obtained from the convergence characteristic of Fig. 8:

1. Premature convergence when solving non-convex, non-smooth and non-linear problems is still the main deficiency of the original SSOA.
2. Using the proposed strategies in the EGSSOA ensures that the diversity of the population is preserved to discourage premature convergence.
3. Moreover, it is clear that the proposed EGSSOA successfully avoids falling into the local optimum, which is far from the global optimum and obtain the global or near-global optima in small maximum iteration.
4. The EGSSOA performs better on more complex problem when the SSOA miss the global optimum basin.
5. The EGSSOA effectively surpasses SSOA to reach the global or near-global optima.

5.7 Test system 4: 80-units

To be able to cope with a more complicated test system with highly non-linearity and non-smoothness is one of the major goals of the EGSSOA applications. Meanwhile, with regards to mathematical and meta-heuristic optimisation techniques and similar approaches, the 10, 15 and 40-unit test systems are not adequate to clearly demonstrate the scalability of the suggested approach. Therefore, the 80-unit test system is applied. The ED problem is handled by EGSSOA and is compared with other methods in Table 10.

In this case, the best, worst and mean fuel cost is $242 795.3327, which is lower than the results of the other methods. The simulation results in this table demonstrate the effectiveness of the proposed method in handling ED with all practical constraints in real-size test system.

The convergence characteristic of the proposed method is presented in Fig. 9 and compared with convergence characteristic of original SSOA. In this case study, a successful percentage is applied to show the robustness of the proposed algorithm. The successful percentages for all

Table 10 Results obtained by different methods for test system 4

Solution technique	Fuel cost, $			Per-unit CPU time, s
	Best value	Mean value	Worst value	
NAPSO [33]	242 844.1172	NR	NR	28.7500
FAPSO [33]	244 273.5429	NR	NR	46.5250
PSO [33]	249 248.3751	NR	NR	85
MGSO [10]	242 815.4714	242 818.296	242 823.291	NR
CSO [34]	243 195.3781	243 546.6283	244 038.7352	NR
proposed EGSSOA		242 795.3327		3.15

NR, not reported in the literature

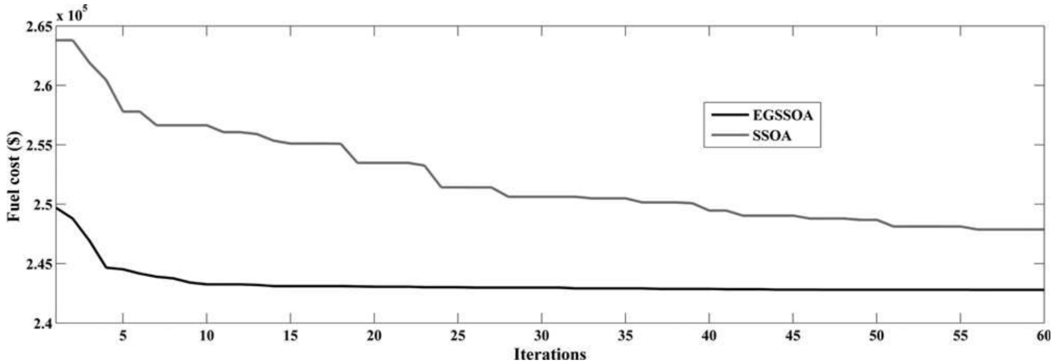


Fig. 9 *Convergence characteristics of EGSSOA and SSOA for test system 4*

Table 11 Successful percentage (%) of different methods for test system 4 out of 30 trail runs

Solution technique	Range of fuel cost, $				
	242 795–246 456	246 456–250 117	250 117–253 777	253 777–257 438	257 438–261 099
success rate of GM	0	0	0	0	100%
success rate of SSOA	33.3%	66.7%	0	0	0
success rate of EGSSOA	100%	0	0	0	0

the methods and the proposed EGSSOA technique to solve the ED problem over 80-unit test system are depicted in Table 11. When the complexity of the problem increases, its overall successful percentage decreases, especially for non-smooth problems. One important point is obtaining the better result with respect to other algorithms, which shows that this algorithm can search in the complex optimisation problems' search space very well.

6 Conclusions

This paper proposed a generalised and versatile EGSSOA approach on the basis of integrating the mathematical gradient-based optimisation technique and the novel presented meta-heuristic ESSOA. Although there are many mathematical-based and meta-heuristic optimisation techniques in the area of solving ED problems, but all of them have some drawbacks. This paper removed the drawbacks of each technique, that is, ESSOA by aiding other technique, that is, GM. In addition, a probabilistic mutation strategy is devised for original SSOA to increase the diversity of the population and to prevent them from trapping in local optima. These attempts are made to find the robust, fast and global or near-global solution of the constrained ED problems with non-smooth, non-convex, non-linear and multi-modal characteristics. The successful performance on the small, medium and large-scale test systems demonstrated that unlike the meta-heuristic optimisation method, the suggested method provides more robust and consistent solution in affordable time. Also, unlike the mathematical-based optimisation method, the EGSSOA is a reliable tool for obtaining global or near-global optima even in multi-dimensional test systems.

The research study is under way in order to apply the proposed method in other power system combinatorial optimisation problems, like unit commitment, optimal power flow, reactive power and voltage control in the search of better quality results.

7 References

1 Ciornei, I., Kyriakides, E.: 'A GA-API solution for the economic dispatch of generation in power system operation', *IEEE Trans. Power Syst.*, 2012, **27**, (1), pp. 233–242

2 Dodu, J.C., Martin, P., Merlin, A., Pouget, J.: 'An optimal formulation and solution of short-range operating problems for a power system with flow constraints', *IEEE Proc.*, 1972, **60**, (1), pp. 54–63

3 Jabr, R.A., Coonick, A.H., Cory, B.J.: 'A homogeneous linear programming algorithm for the security constrained economic dispatch problem', *IEEE Trans. Power Syst.*, 2000, **15**, (3), pp. 930–936

4 Chen, C.L., Wang, S.C.: 'Branch-and bound scheduling for thermal generating units', *IEEE Trans. Energy Conver.*, 1993, **8**, (2), pp. 184–189

5 Dos, L., Coelho, S., Mariani, V.C.: 'Combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect', *IEEE Trans. Power Syst.*, 2006, **21**, (2), pp. 989–996

6 Nanda, J., Hari, L., Kothari, M.L.: 'Economic emission load dispatch with line flow constraints using a classical technique', *IEE Proc. Gener. Transm. Distrib.*, 1994, **141**, (1), pp. 1–10

7 El-Keib, A.A., Ma, H., Hart, J.L.: 'Environmentally constrained economic dispatch using the Lagrangian relaxation method', *IEEE Trans. Power Syst.*, 1994, **9**, (4), pp. 1723–1729

8 Lin, W.-M., Gow, H.-J., Tsai, M.T.: 'Combining of direct search and signal-to-noise ratio for economic dispatch optimization', *Energy Convers. Manage.*, 2011, **52**, (1), pp. 487–493

9 Lowery, P.G.: 'Generating unit commitment by dynamic programming', *IEEE Trans. Power Appar. Syst.*, 1996, **PAS-85**, (5), pp. 422–426

10 Zare, K., Tarafdar Haque, M., Davoodi, E.: 'Solving non-convex economic dispatch problem with valve point effects using modified group search optimizer method', *Electr. Power Syst. Res.*, 2012, **84**, (1), pp. 83–89

11 Yang, X.-S., Sadat Hosseini, S.S., Gandomi, A.H.: 'Firefly algorithm for solving non-convex economic dispatch problems with valve loading effect', *Appl. Soft Comput.*, 2012, **12**, (3), pp. 1180–1186

12 Mohammadi-Ivatloo, B., Rabiee, A., Soroudi, A., Ehsan, M.: 'Iteration PSO with time varying acceleration coefficients for solving non-convex economic dispatch problems', *Int. J. Electr. Power Energy Syst.*, 2012, **42**, (1), pp. 508–516

13 Barisal, A.K.: 'Dynamic search space squeezing strategy based intelligent algorithm solutions to economic dispatch with multiple fuels', *Int. J. Electr. Power Energy Syst.*, 2013, **45**, (1), pp. 50–59

14 Moradi-Dalvand, M., Mohammadi-Ivatloo, B., Najafi, A., Rabiee, A.: 'Continuous quick group search optimizer for solving non-convex economic dispatch problems', *Electr. Power Syst. Res.*, 2012, **93**, pp. 93–105

15 Wang, L., Li, L.P.: 'An effective differential harmony search algorithm for the solving non-convex economic load dispatch problems', *Int. J. Electr. Power Energy Syst.*, 2013, **44**, (1), pp. 832–843

16 Saber, A.Y.: 'Economic dispatch using particle swarm optimization with bacterial foraging effect', *Int. J. Electr. Power Energy Syst.*, 2012, **34**, (1), pp. 38–46

17 Cai, J., Li, Q., Li, L., Peng, H., Yang, Y.: 'A fuzzy adaptive chaotic ant swarm optimization for economic dispatch', *Int. J. Electr. Power Energy Syst.*, 2012, **34**, (1), pp. 154–160

18 Dieu, V.N., Ongsakul, W., Polprasert, J.: 'The augmented Lagrange Hopfield network for economic dispatch with multiple fuel options', *Math. Comput. Model.*, 2013, **57**, (1–2), pp. 30–39

19 Cai, J., Li, Q., Li, L., Peng, H., Yang, Y.: 'A hybrid CPSO-SQP method for economic dispatch considering the valve-point effects', *Energy Convers. Manage.*, 2012, **53**, (1), pp. 175–181

20 Shaw, B., Mukherjee, V., Ghoshal, S.P.: 'Solution of economic dispatch problems by seeker optimization algorithm', *Expert Syst. Appl.*, 2012, **39**, (1), pp. 508–519

21 Vo, D.N., Ongsakul, W.: 'Economic dispatch with multiple fuel types by enhanced augmented Lagrange Hopfield network', *Appl. Energy*, 2012, **91**, (1), pp. 281–289

22 Lohokare, M.R., Panigrahi, B.K., Pattnaik, S.S., Devi, S., Mohapatra, A.: 'Neighborhood search-driven accelerated biogeography-based optimization for optimal load dispatch', *IEEE Trans. Syst., Man, Sybern. C, Appl. Rev.*, 2012, **42**, (5), pp. 641–652

23 Sajjadi, S.M., Yazdankhah, A.S., Ferdowsi, F.: 'A new gumption approach for economic dispatch problem with losses effect based on valve-point active power', *Electr. Power Syst. Res.*, 2012, **92**, pp. 81–86

24 Yeh, W.C.: 'Simplified swarm optimization in disassembly sequencing problems with learning effects', *Comput. Oper. Res.*, 2012, **39**, pp. 2168–2177

25 Wood, A.J., Wollenberg, B.F.: 'Power generation, operation and control' (Wiley, New York, 1996, 2nd edn.)

26 Yeh, W.C.: 'Optimization of the disassembly sequencing problem on the basis of self-adaptive simplified swarm optimization', *IEEE Trans. Syst., Man Cybern–A, Syst. Hum.*, 2012, **42**, pp. 250–261

27 Niknam, T., Azizipanah-Abarghooee, R., Narimani, M.R.: 'Reserve constrained dynamic optimal power flow subject to valve-point effects, prohibited zones and multi-fuel constraints', *Energy*, 2012, **47**, pp. 451–464

28 Hagan, M.T., Dcmuth, H.B., Beale, M.: 'Neural network design' (PWS Publishing Company, USA, 1996, 1st edn.)

29 Chiang, C.-L.: 'Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels', *IEEE Trans. Power Syst.*, 2005, **20**, (4), pp. 1690–1699

30 Gaing, Z.L.: 'Particle swarm optimization to solving the economic dispatch considering generator constraints', *IEEE Trans. Power Syst.*, 2003, **18**, (3), pp. 1718–1727

31 Sinha, N., Chakrabarti, R., Chattopadhyay, P.K.: 'Evolutionary programming techniques for economic load dispatch', *IEEE Trans. Evol. Comput.*, 2003, **7**, (1), pp. 83–94

32 Niknam, T., Doagou Mojarrad, H., Zeinoddini Meymand, H., Bahmani Firouzi, B.: 'A new honey bee mating optimization algorithm for non-smooth economic dispatch', *Energy*, 2011, **36**, (2), pp. 896–908

33 Niknam, T., Doagou Mojarrad, H., Zeinoddini Meymand, H.: 'Non-smooth economic dispatch computation by fuzzy and self-adaptive particle swarm optimization', *Appl. Soft Comput.*, 2011, **11**, (2), pp. 2805–2817

34 Selvakumar, A., Thanushkodi, K.: 'Optimization using civilized swarm: solution to economic dispatch with multiple minima', *Electr. Power Syst. Res.*, 2009, **79**, (1), pp. 8–16

35 Panigrahi, B.K., Pandi, V.R.: 'Bacterial foraging optimization: Nelder–Mead hybrid algorithm for economic load dispatch', *IET Gener. Transm. Distrib.*, 2008, **2**, (4), pp. 556–565

36 Chaturvedi, K.T., Pandit, M., Srivastava, L.: 'Self-organizing hierarchical particle swarm optimization for nonconvex economic dispatch', *IEEE Trans. Power Syst.*, 2008, **23**, (3), pp. 1079–1087