Santra et al.
Journal of Electrical Systems and Inf Technol (2020) 7:3
https://doi.org/10.1186/s43067-020-0011-2

Journal of Electrical Systems and Information Technology

RESEARCH Open Access

Dynamic economic dispatch using hybrid metaheuristics

Dipankar Santra[1*], Anirban Mukherjee[2], Krishna Sarker[3] and Subrata Mondal[4]

*Correspondence: santra.dipankar@gmail.com
[1] Department of Electrical Engineering, RCC Institute of Information Technology, Canal South Road, Beliaghata, Kolkata, West Bengal 700015, India
Full list of author information is available at the end of the article

Abstract

Dynamic economic dispatch problem or DED is an extension of static economic dispatch problem or SED which is used to determine the generation schedule of the committed units so as to meet the predicted load demand over a time horizon at minimum operating cost under ramp rate constraints and other constraints. This work presents an efficient hybrid method based on particle swarm optimization (PSO) and termite colony optimization (TCO) for solving DED problem. The hybrid method employs PSO for global search and TCO for local search in an interleaved mode towards finding the optimal solution. After the first round iteration of local search by TCO, the best local solutions are considered by PSO to update the schedules globally. In the next round, TCO performs local search around each solution found by PSO. This paper reports the methodology and result of application of PSO–TCO hybrid to 5-unit, 10-unit and 30-unit power dispatch problems; the result shows that the PSO–TCO (HPSTCO) gives improved solution compared to PSO or TCO (when applied separately) and also other hybrid methods.

Keywords: Dynamic economic dispatch (DED), Ramp rate limits, Valve point loading effect, Heuristic optimization techniques, Metaheuristics

Introduction

Usually the economic load dispatch problem (ELD) implies static economic dispatch problem or SED where the objective is to determine the optimal schedule of online generating units' outputs so as to meet the load demand at a certain time at the minimum operating cost under various system and operational constraints. In contrast, the objective of the dynamic economic dispatch (DED) problem is to schedule the generator outputs over a certain period of time economically. The DED problem takes into consideration the limits on the generator ramping rate coupled with real time intervals to keep the thermal stress on the generation equipment like the turbines and boilers within the safe limits and thus protect their life [1]. The DED problem divides the dispatch period into a number of small time intervals, and a SED is employed to solve the problem in each interval.

Since the DED problem was introduced in 1980s, several optimization techniques and procedures have been used for solving the DED problem with complex objective functions or constraints. There were a number of classical methods that have been applied to solve this problem such as the lambda iterative method [2], gradient projection method

[3], Lagrange relaxation [4], linear programming [5], dynamic programming [6] and interior point method [7]. Most of these methods are not applicable for non-smooth or non-convex cost functions. To overcome this problem, many heuristic optimization methods have been employed to solve the DED problem; such methods include ant colony optimization (ACO) [8], particle swarm optimization (PSO) [9–11], Levenberg–Marquardt back-propagation algorithm (LMBPA) [12], differential evolution (DE) [13], artificial immune system (AIS) algorithm [14], harmony search (HS) [15] and bee swarm optimization (BSO) algorithm [16] among others. Many of these techniques have proved their effectiveness in solving the DED problem without any or fewer restrictions on the shape of the cost function curves. These approaches solve the DED by employing an initial population of individuals each of which represents a candidate solution for the problem. Then, they evolve the initial population by successively applying a set of operators on the old solutions to transform them into new solutions.

In recent years, the trend of solving DED problems has changed from single-heuristic techniques to hybrid metaheuristics—a combination of two or more techniques like PSO–ACO, DE–SQP, PSO–SQP, etc. It is proved that these hybrid techniques have capability to solve the DED problems better than the single-heuristic problems as the hybridization causes the individual techniques to mitigate their limitations and complement each other with their characteristic strength.

Earlier hybrids

In 2005, Victoire et al. proposed hybrid EP–SQP made up of evolutionary programming and sequential quadratic programming technique to solve DED problems [17]. They also experimented with a hybrid of PSO and SQP techniques to solve DED problem with valve point loading (VPL) effect [18]. In 2009, Yuan et al. [19] hybridized PSO with differential evolution method for solving DED with VPL. In 2010, hybrid SOA–SQP method that combined seeker optimization algorithm (SOA) with SQP was used by Sivasubramani and Swarup [20] to solve DED with VPL. In 2012, Cai et al. [21] reported application of hybrid CPSO–SQP in DED with VPL. Again in 2012, Swain et al. [22] hybridized gravitational search algorithm (GSA) with SQP (GSA–SQP) to solve DED with VPL.

Elaiw et al. [23] compared, in 2013, the efficacy of hybrids DE–SQP and PSO–SQP in solving DED with VPL effect. Chen et al. [24] used a combination of three methods, namely low-discrepancy sequences (LDS), improved shuffled frog leaping algorithm (ISFLA) and SQP, to solve DED problem. In this hybrid (termed as LDISS), LDS is used to generate initial population, ISFLA is liable for global search and SQP is used for local search. In 2013, Mohammadi-Ivatloo et al. [25] introduced hybrid immune genetic algorithm to solve DED considering VPL and prohibited operating zone and ramp rate constraints along with transmission losses. Zhang et al. [26] proposed hybrid bare bones (BB)—PSO or BBPSO in 2014 to solve DED with VPL only.

Among recent developments are two hybrid techniques, BBO–PSOTVAC and FA–PSOTVAC, developed in 2018 by Hamed et al. [27], combining firefly algorithm (FA) and biogeography-based optimization (BBO) with time-varying acceleration-based particle swarm optimization (PSOTVAC) to improve the solution of DED. In the same year, Pan et al. [28] solved the DED problem with VPL using a hybrid technique MILP–IPM involving mixed-integer linear programming (MILP) and interior point method (IPM). Another very recent development (in 2018) by Xiong and Shi [29] is a hybrid of BBO and brain storm optimization (BSO) to get a better solution for DED with VPL.

In the present work, the authors have applied for the first time a hybrid computational approach HPSTCO that combines PSO and TCO to solve the DED problem. The hybrid method employs PSO iterations for global search and TCO iterations for exploring the locality near the global solutions, interleaving both the search processes to overcome the drawback of fast convergence to (selection of) global optimal solution in the original PSO method. The interleaving process requires PSO and TCO pass their solutions to each other. The solutions of TCO are updated in PSO iterations by considering the global best solution. Similarly, the solutions of PSO are adjusted by considering the locally observed information by TCO. A solution point searched by the PSO method can be used as an initial condition in the TCO method. The hybrid HPSTCO model has been programmed in MATLAB and simulation run executed for 5-unit, 10-unit and 30-unit DED system with parameters referred from the literature. Performance of the HPSTCO, as compared using a benchmark function, is quite encouraging.

Motivation and contribution

The novelty of the present study lies in the fact that PSO and TCO together, i.e., their hybrid combination (HPSTCO) has never been tried before to optimize small to large-scale economic load dispatch problem. However, PSO and TCO individually and in combination with other metaheuristics have been applied earlier in different ELD problems. The contribution of the paper lies in the fact that it has established by reporting four distinct test cases and comparing the result with other hybrid methods for each of these four test cases that the HPSTCO hybrid is quite effective and advantageous in dealing with small-scale (5- and 10-unit) as well as medium-scale (30-unit) DED problem. In medium to large-scale systems with higher-capacity turbines, the fuel cost function is highly non-smooth and non-convex and contains discontinuous values at each boundary, forming multiple local optima. The complexity of the problem also increases significantly with the increase in the number of generating units because of their combinatorial nature. The present work has tackled this challenge nicely having no earlier precedence of application of this particular (HPSTCO) hybrid optimization mechanism. Therefore, it can be said that this paper introduces a new metaheuristics in DED with significant results.

The current study is done on four different test cases of DED involving 5, 10 and 30 generating units:

Case 1 5-unit system with valve point effects, ramp rate constraints, prohibited operating zones and transmission losses
Case 2 10-unit system with valve point effects, ramp rate constraints and transmission losses
Case 3 10-unit system with valve point effects and ramp rate constraints without transmission losses
Case 4 30-unit system with valve point effects and ramp rate constraints without transmission losses.

Organization

The rest of this paper is organized as follows: The "Model" section presents the DED problem formulation with all constraints and limitations. The "Method" section explains the basic concepts and searching principle of each of PSO and TCO methods. In the same section, the proposed HPSTCO algorithm is introduced and described with flowchart and details of the stages. Case studies, simulation results, result analysis and comparison are presented in "Result and discussion" section. Finally, "Conclusion" section draws some concluding remarks on the limitation of the present work and future scope of research.

Model

A comprehensive study of basic DED problem is done here. A non-smooth, non-convex, non-differentiable single- and multi-objective multi-constraint model of ED problem is formulized in this section.

Objective function

The objective function of DED problem, which is to minimize the total production cost over the operating horizon, can be written as:

$$\min C_{\mathrm{T}} = \sum_{t=1}^{\mathrm{T}} \sum_{i=1}^{n} C_{i,t}\left(P_{i,t}\right) \tag{1}$$

where C_{T} (in\$/h) is the total generation cost, $C_{i,t}$ is the generation cost of ith unit at time t, n is the number of dispatch-able power generation units; here, $n = 5$, 10 and 30, and $P_{i,t}$ (in MW) is the power output of ith unit at time t. T is the total number of hours from operational point of view. The basic ELD objective function is represented by a non-smooth curve (quadratic polynomial) with VPL effect (ripple effect) modeled with a sinusoidal function as shown in Eq. (2).

$$C_T = \sum_{i=1}^{n} C_i(P_i) = \sum_{i=1}^{n} a_i + b_i P_i + c_i (P_i)^2 + \left| e_i \sin\left(f_i \left(P_i^{\min} - P_i \right) \right) \right| \tag{2}$$

where a_i (in \$/h), b_i (in \$/MWh) and c_i (in \$/MW^2 h) are the cost coefficients of the ith unit, and e_i (in \$/h) and f_i (in 1/MW) are the VPL coefficients of the ith unit. $\mathrm{P}^{\min}{}_{,i}$ (in MW) is the minimum generation capacity limit of unit i. In the generation cost function, the term $\left| e_i \sin(f_i (P_i^{\min} - P_i)) \right|$ represents the VPL effect.

The objective function (Eq. 1) of the DED problem should be minimized subject to the following constraints.

Real power balance constraint

$$\sum_{i=1}^{n} P_i = P_{\mathrm{D}} + P_{\mathrm{L}} \tag{3}$$

In Eq. (3), P_i (in MW) is the power generated by the ith unit, P_{D} (in MW) is the total load demand and P_{L} (in MW) is the total transmission loss of the system at time t. P_{L} is computed using *B-coefficients* that can be calculated by using Kron's loss formula known as B-matrix coefficients. In this work, B-matrix coefficients method is used to calculate system loss, as follows:

$$P_{\mathrm{L}}(t)=\sum_{i=1}^{n}\sum_{j=1}^{n}P_{i,t}B_{ij}P_{j,t}+\sum_{i=1}^{n}P_{i,t}B_{0i}+B_{00},\quad \text{where } t=1,2,3,\ldots T \tag{4}$$

Generator capacity constraint

$$P_i^{\min}\le P_{i,t}\le P_i^{\max}\quad i=1,2\ldots,n;\quad t=\ 1,2,\ldots,T \tag{5}$$

The generator power output (P_{i}) of ith generator is within minimum power $P_i^{\min}$ and maximum power $P_i^{\max}$ (in MW).

Ramp rate limit (RRL)

A production unit, which is used for generating power P_{i0}, can increase or decrease its active power output ($P_{i,t}$) within upper ramp rate (UR_i) limit (in MW/h) and down ramp rate (DR_i) limits (in MW/h) as shown in Eqs. (6) and (7).

$$\text{For increase in output power: } P_i-P_{i0}\le \mathrm{UR}_i \tag{6}$$

$$\text{For decrease in output power: } P_{i0}-P_i\le \mathrm{DR}_i \tag{7}$$

Combining Eqs. (6) and (7), we get

$$\max\left(P_i^{\min},P_{i0}-\mathrm{DR}_i\right)\le P_i\le\min\left(P_i^{\max},P_{i0}+\mathrm{UR}_i\right) \tag{8}$$

where $i=1, 2 \ldots, n$; $t=1, 2, \ldots, T$.

Prohibited operating zone (POZ)

Prohibited operating zone demarcates the scope of active power output of a generator which is otherwise affected due to the technical operation of shaft (unreasonable vibrations of bearing). Usually, modification of power is not allowed in the prohibited spans. The allowable operating range of a generator is given as in Eq. (8).

$$\begin{aligned}&P_i^{\min}\le P_{i,t}\le P_{i,1}^{\mathrm{lower}}\\&P_{i,j-1}^{\mathrm{upper}}\le P_{i,t}\le P_{i,j}^{\mathrm{lower}};\quad i=1,\ldots,n;\quad j=2,3,\ldots,n_i;\quad t=1,2,\ldots,T\\&P_{i,n_i}^{\mathrm{upper}}\le P_i\le P_i^{\max}.\end{aligned} \tag{9}$$

here j is the number of POZs, $P_{i,j-1}^{\mathrm{upper}}$ is the 'upper boundary' and $P_{i,j}^{\mathrm{lower}}$ is the 'lower boundary' of the jth POZ of the ith unit. n_i is the number of prohibited operation zones of unit i. The main objective of DED is to minimize the generation cost C_{T} and optimize the power generation schedule ($P_{i,t}$) as in Eqs. (1) or (2) subject to satisfying the constraints in Eqs. (3) to (9) used with different combinations in different test cases.

Method

The hybrid approach HPSTCO taken up in this study comprises two basic metaheuristics, namely particle swarm optimization (PSO) and termite colony optimization (TCO). A brief outline and working principle of these two optimization techniques are first discussed in this section.

Particle swarm optimization (PSO)

PSO is a swarm intelligence technique inspired from social behavior of bird flocking and fish schooling. Birds and fish follow the neighbor that is nearest to the food, when they search for food. Each individual solution in PSO is named as 'particle' and represents a bird or a fish in the search space.

Each particle has a position, velocity and fitness value. While they move in the solution space of fitness function, the particles aim to improve their next position based on their past experience and the best position in the swarm. Therefore, every individual is gravitated toward a stochastically weighted average of the previous best position of its own and that of its neighborhood companions [30]. In every iteration of PSO, the position and velocity of every particle is updated and the value of fitness function at its current location is evaluated.

Mathematically, given a swarm of particles, each particle i is associated with a position vector $\vec{X}_i = \{X_{i1}, X_{i2}, \dots X_{iD}\}$, which is a feasible solution for an optimal problem in the D-dimensional search space S. Let the previous best position or *pbest* of a particle i be denoted by $\vec{X}_{pi}$ and the best position that has ever been found by any particle or *gbest* be denoted by X_{gi}. At the start of search, all the positions and velocities are initialized randomly. At each iteration, the position vector of each particle i is updated by adding an increment vector or velocity $\vec{V}_i = \{V_{i1}, V_{i2}, \dots, V_{iD}\}$ as per Eq. (11). The velocity is updated according to Eq. (10):

$$\vec{V}_i\left\{k+1\right\} = \vec{V}_i\{k\} + c_i r_i\left\{\overrightarrow{X}_{pi} - \overrightarrow{X}_i\{k\}\right\} + c_2 r_2\left(\overrightarrow{X}_{gi} - \overrightarrow{X}_i\{k\}\right) \tag{10}$$

$$\vec{X}_i\left\{k+1\right\} = \vec{X}_i\{k\} + \vec{V}_i\{k-1\} \tag{11}$$

$\vec{V}_i k$ and $\vec{V}_i\{k+1\}$ represent velocity vectors for particle i in the previous and current iterations, c_1 and c_2 are two positive constants, and r_1 and r_2 are two random parameters of uniform distribution in range of [0, 1], which limit the velocity of the particle in the coordinate direction. The new location of each particle should be compared with the *pbest* value. If the new location of the particle is better than the *pbest* value, then the *pbest* is updated for the new location. Otherwise the original value of *pbest* is stored unchanged. The new global optimum solution *gbest* is updated according to the *gbest* of the new particle swarm. This iterative process will continue until a stop criterion is satisfied or maximum number of iterations has been done. Eventually, the particle swarm will converge to the global optimum solution.

Termite colony optimization

TCO is an optimization method inspired from intelligent behaviors of termites during their mound structure building process. Initially the termites arbitrarily search for soil pallets, and after finding it, they deposit it on the mound. Later on, the termites move on the basis of observed trail of pheromone (a chemical) that they deposit on the path on returning after depositing soil pallets on the termite mound. The pheromone acts as attractive stimulus to other members of the colony to follow smaller paths with higher intensities as it is a volatile chemical that evaporates with time. Termites that travel the shortest path reinforce this path with more amount of pheromone, thereby helping others to follow them.

Assuming that the size of the termite population M is within the D-dimensional search space, the position of the ith termite is denoted by $\vec{X}_i = \{X_{i1}, X_{i2}, \ldots, X_{iD}\}$ which indicates a possible solution of an optimization problem. The cost/fitness function value for each position $\vec{X}_i$ is *fit* $(\vec{X}_i)$ which represents amount of pheromone deposited on a hill. The basic steps of TCO can be summarized as follows:

1. Initialize the population as weights, position of termites and the number of iterations. (Every termite has its distinct random position, velocity, desirability and rate of evaporation of pheromone).
2. Evaluate the fitness function value for each termite.
3. Determine the best position and evaporation rate of pheromone of each termite.
4. Determine the position of the best termite.
5. Update the evaporation rate of pheromone, velocity and position of each termite.
6. Stop if the condition of optimization is satisfied. If not, repeat from step 2.

If $\tau_i\{t-1\}$ and $\tau_i\{t\}$ stand for the pheromone level at the current and previous locations, respectively, of ith termite, then the pheromone updates rule states:

$$\tau_i\{t\} = (1-\rho)\tau_i\{t-1\} + \frac{1}{\left(\text{fit}(\vec{X}_i) + 1\right)}, \tag{12}$$

where ρ is the evaporation rate of pheromone taken in the range of [0–1].

After updating the pheromone level, each termite adjusts its route and moves to a new location. Therefore, termite movement is a function of pheromone level at the visited location and the distance between a termite location and the visited locations. Now there are two possible directions of movement: if there is no previously visited location (by the swarm) in the neighborhood of a termite, it moves randomly; if there are one or more visited locations, then the termite selects the location with highest level of pheromone and moves to that position. When the termite moves randomly to search a new gainful position, then position is updated as:

$$\vec{X}_i\{t\} = \vec{X}_i\{t-1\} + R_{\text{w}}\left(\tau, \vec{X}_i\{t-1\}\right) \tag{13}$$

here $\vec{X}_i(t-1)$ and $\vec{X}_i(t)$ represent the current and new position of the termite, respectively; R_w is a random walk function of current position and radius of search.

When the termite moves toward a gainful position or best local position $\vec{B}_i$ having higher level of pheromone compared to the current position, then the position is updated as:

$$\vec{X}_i\{t\} = \vec{X}_i\{t-1\} + w_b \tau_b, \left\{\vec{B}_i - \vec{X}_i\{t-1\}\right\}, \quad \text{if } \{\tau_i\{t-1\} < \tau_{bi}\{t-1\}\} \tag{14}$$

here $1 < w_b \le 2$ and $0 < r_b < 1$ probabilistically controls the attraction of the termite toward local best position.

Hybrid of PSO and TCO (HPSTCO)

The present work adopts a hybrid of PSO and TCO algorithm (call it as HPSTCO), expecting their usefulness in solving DED problems would be enhanced when used as a combination in complementary mode. The HPSTCO exploits the global search potential of the PSO along with the local search potential of TCO in a given search space. While PSO iterations produce globally distributed solutions (overlooking the localized search space around each global solution), its hybrid partner TCO complements PSO by exploring in more detail any potential localized solution. The solutions obtained by the PSO iterations are fed to the TCO iterations in order to gravitate more termites toward gainful positions. Again the solution found by the termites in TCO updates the positions of the corresponding particles, thereby giving a good starting point of the particles in the global search space.

The basic input parameters of HPSTCO are: maximum number of iterations (*max_iter*), population size (s), number of PSO iterations (n_1), number of TCO iterations (n_2) and number of solutions which are fed from PSO (TCO) to the TCO (PSO) at the switching time (η). The parameter n_1 (n_2), respectively, shows how many times PSO iterations (TCO iterations) should be executed before a switching time, implying n_1 iterations of PSO are followed by n_2 iterations of TCO.

The HPSTCO has six stages: (1) initialization, (2) global search, (3) switch from global search to local search, (4) local search, (5) switch from local to global search and (6) constraint handling.

The pseudocode of the hybrid algorithm is given in Fig. 1.

Initialization

The HPSTCO algorithm starts with PSO iterations with n number of particles placed in random position in the solution space. A position is a candidate for the priority list $\vec{P} = (p_1, p_2, \ldots p_n)$. Each element of the list represents an activity, and its corresponding value shows the priority of that activity. Hence, the position vector $\vec{X}_i = \{X_{i1}, X_{i2}, \ldots, X_{iD}\}$ of each individual i represents the priority values of n activities.

A solution space of priorities will be created where the lower and upper bounds will be defined as Lb = 0:0 and Ub = 1.0. The value of each element must be limited to [Lb, Ub].

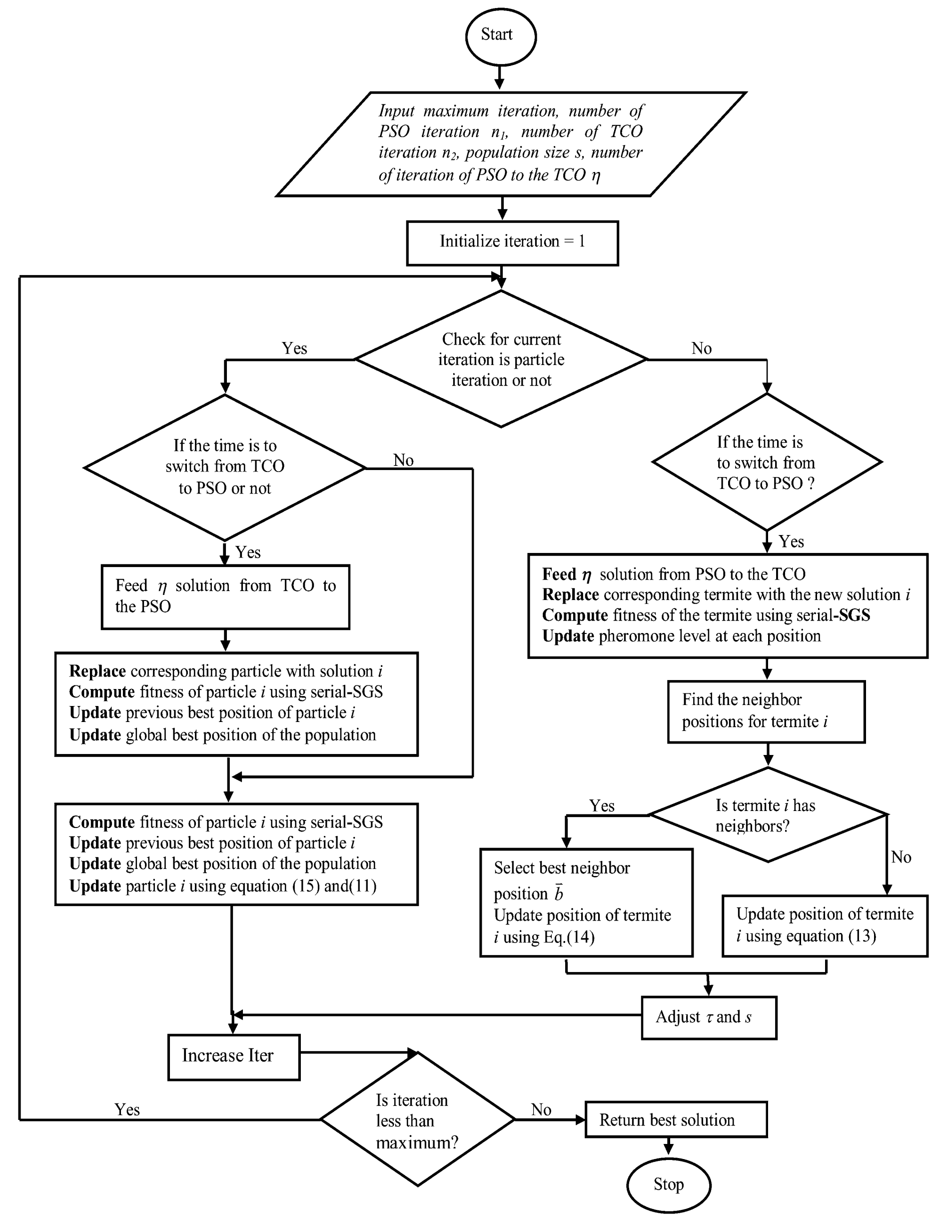

Fig. 1 Flowchart of the proposed HPSTCO algorithm

Global search

Each particle presents a possible schedule for the DED problem. The velocity of the particles is updated by Eq. (13) which is a modified form of Eq. (10) of original PSO:

$$\vec{V}_i\{k+1\} = \gamma\left\{\vec{V}\{k\} + c_1 r_1\{\vec{X}_{pi} - \vec{X}_i\{k\}\} + c_2 r_2\left(\vec{X}_{gi} - \vec{X}_i\{k\}\right)\right\} \tag{15}$$

where $0 < \gamma < 1$ is a constriction factor that improves the convergence speed. The position of each particle is updated by considering its current position and *pbest* and *gbest* values are determined by calculating the fitness of the proposed schedules.

Switching from global to local search

The HPSTCO method simply switches from PSO to TCO, while switching a part of solutions found by the PSO that is passed to the TCO. Each solution determines the start position of a termite in the next iteration of TCO. Basically each particle switches its type as termite.

Local search

The TCO uses the solutions which are passed from PSO as the start positions of its termites. Next, TCO tries to find improved solutions in the local neighborhoods of those solutions (now the termites). To determine the neighborhood for each termite, the Euclidian distances of all termites from the candidate termite are calculated. If the distance is smaller than a threshold, the corresponding termite is considered as a neighbor of the candidate termite. The threshold value is dynamically adjusted, gradually decreasing as the algorithm proceeds. The termite with no neighbor moves randomly following Eq. (13); the termite having one or more neighbors selects one of them randomly as its neighbor and updates its position following Eq. (14).

Switching from local to global search

In this phase, each termite switches its type as particle. The solution found by termites updates the positions of the corresponding particles in the PSO. The earlier best position of each particle (*pbest*) and the global best position of the entire swarm is updated accordingly. The updated fitness of the new solution for a particle is compared to its previous fitness value; if the new fitness value is better, it will be considered as the new *pbest*. Similarly, the *gbest* position is compared with this new *pbest* position, and if the later has better fitness compared to the *gbest*, then the *gbest* value is updated with the current *pbest*.

Constraint handling

In each cycle of HPSTCO, a new population of feasible and infeasible solutions is generated. An infeasible solution is the one which violates the constraints of the problem. After detection of an infeasible solution, it is recovered as a feasible solution. The activity which violates the constraints is changed with the next activity (in the activity list) with lesser priority, and the constraint handling process is applied on the new activity list. This process is iterated until the infeasible solution is converted to a feasible solution.

Pseudocode

```
Input:
  max_iter: maximum number of iterations
  n1 : number of PSO iterations
  n2 : number of TCO iterations
  s : population size
  η : number of solutions which are feed from PSO to the TCO and vice
  versa
Initialization:
  Distribute randomly PSO particles over the solution space
Update:
Do for each iteration iter, while 1 ≤ iter ≤ max_iter
     If switch1=1 for TCO to PSO then
          Pass solution from TCO to PSO
          Do for each new solution i
               Replace corresponding particle with the solution i
               Compute fitness value of particle i
               Update pbest of particle i
               Update gbest of the population
          End Do
     End If
     Do for each particle i
          Compute fitness value of particle i
          Update gbest of the population
          Update particle i using equations (15) and (11)
     End Do
  Else
  If switch2=1 for TCO to PSO then
     Pass solution from PSO to TCO
        Replace corresponding termite with the solution i
        Compute fitness value of the termites
        Update pheromone level at each position
     End if
     Do for each termite i
        Find the neighbor positions for termite i
        If termite i has neighbor then
          Select best neighbor position bi
          Update position of termite i using equations (14)
        Else
          Update position of termite i using equations (13)
        End if
     End Do
     Adjust radius r and step size s
   End Do
   Terminate:
   Return Best Solution
```

Table 1 Settings of the HPSTCO parameters

Parameter	Value	Parameter	Value
c_1: Acceleration coefficient	1.0	n_1: No. of PSO iteration	1.0
c_2: Acceleration coefficient	1.0	n_2: No. of TCO iteration	1.0
η: Constriction factor	0.7	Q: Radius parameter	0.4
w_b: Weight	1.0	s: Population size	50

Result and discussion

In order to review the effectiveness of HPSTCO, it is applied to solve the DED problem on three test systems having 5, 10 and 30 generators, considering valve point loading effect. The algorithm has been coded using MATLAB and implemented on a 64-bit PC with the detailed settings as follows:

> *Hardware* CPU: Intel® Core™ i5-6200U, frequency: 2.30 GHz, RAM: 8.0 GB, hard drive: 500 GB
> *Software* Operating system: Windows 10, package: MATLAB 8.1 (R2014a).

The values of the input parameters of the algorithm are depicted in Table 1.

The simulation in MATLAB is done on four different test cases of DED involving 5, 10 and 30 generating units:

Case 1 5-unit system with valve point effects, ramp rate constraints, prohibited operating zones and transmission losses
Case 2 10-unit system with valve point effects, ramp rate constraints and transmission losses
Case 3 10-unit system with valve point effects and ramp rate constraints without transmission losses
Case 4 30-unit system with valve point effects and ramp rate constraints without transmission losses

Test case 1: 5-unit system

In this test system, the valve point loading effects, ramp rate constraints, prohibited operating zones, transmission and generation limits have been considered. The essential input data of the 5-unit system are enlisted in Table 2 [22] that includes prohibited zones of units 1 to unit 5. These zones result in two disjoint subregions for each of units 1, 2, 3, 4 and 5. The *B*-coefficients matrix used for calculating power loss is given in Table 3. The load demand of the system is separated into 24 dispatch intervals of a day as shown in Table 4.

The population size is 50. The fuel cost and transmission losses obtained by the HPSTCO technique are 42,151.3377 $/day and 194.3182 MW, respectively, as shown in Table 5. The graphical representation of Table 5 is shown in Fig. 2. Table 6 shows the comparison results for the fuel cost obtained for 5-unit DED system by HPSTCO with other hybrid methods as reported in the literature. Table 6 shows that the minimum cost yielded by CMIWO [36], MGDE [37], BBOSB [29], MILP–IPM [28], HIGA [25], BBPSO [26], LDISS-2 [24] are 43,017.9597$/day, 43,084$/day, 43,125.365$/day, 43,233$/day and

Table 2 Input data of the 5-unit test system

Unit	a_i (\$)	b_i (\$/MWh)	c_i (\$/MW^{2}h)	e_i (\$/h)	f_i (rad/MW)	P_{min} (MW)	P_{max} (MW)	UR (MW/h)	DR (MW/h)	POZs (MW)
U_1	25	2	0.008	100	0.042	10	75	30	30	[25 30], [55 60]
U_2	60	1.8	0.003	140	0.04	20	125	30	30	[45 50], [80 90]
U_3	100	2.1	0.0012	160	0.038	30	175	40	40	[60 70], [125 140]
U_4	120	2	0.001	180	0.037	40	250	50	50	[95 110], [160 180]
U_5	40	1.8	0.0015	200	0.035	50	300	50	50	[85 100], [175 200]

Table 3 ***B*-matrix coefficients (per MW) [31]**

$$B_{ij} = \begin{bmatrix} 0.000049 & 0.000014 & 0.000015 & 0.000015 & 0.000020 \\ 0.000014 & 0.000045 & 0.000016 & 0.000020 & 0.000018 \\ 0.000015 & 0.000016 & 0.000039 & 0.000010 & 0.000012 \\ 0.000015 & 0.000020 & 0.000010 & 0.000040 & 0.000014 \\ 0.000020 & 0.000018 & 0.000012 & 0.000014 & 0.000035 \end{bmatrix}$$

43,213$/day, respectively, whereas the cost for HPSTCO is 42,151.3377$ only. The average execution time required for one complete solution was 0.98 min till eighth iteration, and thereafter, the convergence curve becomes a straight line, which is acceptable for DED solutions, though it is not the least in comparison to the time taken by other methods. The convergence characteristic of the proposed algorithm is depicted in Fig. 3.

Test case 2: 10-unit system

In this test case, the valve point effect, ramp rate constraints, transmission and generation limits are considered. The basic input data of the 10-unit system are listed in Table 7. The *B*-matrix coefficients (per MW) for calculating power loss are given in Table 8. The load demand of the system is divided into 24 dispatch intervals as shown in Table 9.

Results obtained by MATLAB simulation are presented in Table 10. The graphical representation of Table 10 is shown in Fig. 4. From Table 11 that compares the output of HPSTCO with that of other recently published hybrid methods such as hybrid MILP–IPM [28], hybrid BBOSB [29], HIGA [25], hybrid LDISS [24] and hybrid EP–SQP [32], it is found that the cost (in $/day) yielded by these methods for 10-unit system is 1,040,676, 1,038,362.014, 1,041,087.802, 1,039,083 and 1,035,748, respectively. In comparison, cost and loss yielded by HPSTCO are 1,035,730.203 $ and 811.6073 MW only which is the least among all. The average execution time required for one complete solution was 1.85 min, which is not the least of all but less than many DED solutions. The convergence characteristic of the HPSTCO is depicted in Fig. 5 which shows that the result converged after 50 generations and 1.85 min.

Test case 3: 10-unit system without transmission loss

Unlike test case 2, here a 10-unit system is considered without transmission loss. Like test case 2, valve point effect, ramp rate constraint and generation limits are considered. Input data or DED parameters of the 10-unit system are sane as listed in Table 7. The load demand of the system is divided into 24 dispatch intervals same as shown in Table 9. Results of best generation schedule at each hourly interval as obtained through MATLAB simulation are presented in Table 12. The fuel cost yielded by the HPSTCO method is 1,015,438.967 $/day. The graphical representation of Table 12 is shown in Fig. 6.

In Table 13, output of proposed HPSTCO for test case 3 is compared with the output given (for the same case) by other recently published hybrid methods, namely ADE–SA [34], hybrid MILP–IPM [28], FA–PSOTVAC [27], BBOSB [26], HIGA [25], LDISS [24], hybrid GSA–SQP [22], hybrid DE [19], EPSO–GM [33] or hybrid PSO–SQP [17]. The generation cost yielded by these methods (taken in the same order) are 1016412.81$/day,

Table 4 Load demand for 24 h for 5-unit system

Time (h)	Load (MW)	Time (h)	Load (MW)	Time (h)	Load (MW)	Time (h)	Load (MW)	Time (h)	Load (MW)	Time (h)	Load (MW)
1	410	5	558	9	690	13	704	17	558	21	680
2	435	6	608	10	704	14	690	18	608	22	605
3	475	7	626	11	720	15	654	19	654	23	527
4	530	8	654	12	740	16	580	20	704	24	463

Table 5 Best fuel cost obtained by using HPSTCO for 5-unit system

Hour	Demand (MW)	U_1	U_2	U_3	U_4	U_5	Loss (MW)	Cost ($/h)
1	410	20.3374	98.5552	30.0314	125.0858	139.8067	3.8164	1229.6038
2	435	53.3657	98.5127	112.305	124.7849	50.0573	4.0257	1319.4197
3	475	10.11	96.0123	110.7303	122.6369	140.2756	4.7651	1401.2008
4	530	60.6107	98.2357	112.4767	124.7797	139.732	5.8349	1587.648
5	558	10.8884	97.8235	112.7349	203.6139	139.6816	6.7422	1603.4944
6	608	57.3389	98.5283	112.526	208.3053	139.1617	7.8601	1770.1108
7	626	74.2931	98.5974	112.5902	209.118	139.7157	8.9287	1787.2622
8	654	12.0297	98.6812	112.5217	210.1569	229.8739	9.2634	1801.0425
9	690	50.7741	98.1661	112.5615	209.145	229.5174	10.1641	1984.7981
10	704	65.9002	98.3341	112.1563	208.974	229.1928	10.5573	2004.6265
11	720	74.448	98.5729	114.9597	210.468	232.5796	11.0283	2049.0192
12	740	74.8962	101.3824	128.4828	212.6752	234.15	11.2567	2214.9176
13	704	65.9002	98.3341	112.1563	208.974	229.1928	10.5073	2004.6265
14	690	49.2195	98.814	112.9205	210.0223	229.192	10.1683	1984.3179
15	654	11.409	99.333	113.2733	209.5672	229.6794	9.2619	1802.9118
16	580	20.534	98.5767	113.401	125.2608	229.428	7.2005	1658.1718
17	558	10.0654	95.734	112.7926	205.8223	140.3331	6.7433	1601.2262
18	608	49.4414	98.4987	113.2245	125.0287	229.6532	7.8464	1786.1253
19	654	11.8197	98.928	112.2354	209.9334	230.3501	9.2666	1805.2276
20	704	64.8938	98.2649	112.4795	209.7271	229.1929	10.5582	2000.2886
21	680	38.1861	98.5191	112.9917	209.8135	230.3942	9.2045	1950.8657
22	605	45.5373	99.2293	30.101	209.6147	228.9415	8.4238	1788.5538
23	527	52.104	98.6116	112.7946	40.6615	228.9582	6.13	1610.998
24	463	74.9426	98.4372	30.4952	124.4585	139.431	4.7645	1404.8809
Total loss (MW)							194.3182	–
Total cost ($/day)								42,151.3377

Table 6 Comparison (of cost and computation time) with other hybrid methods

Method	Minimum cost ($/day)	Transmission loss (MW)	Computation time (min)
HPSTCO	42,151.3377	194.3182	0.98
CMIWO [36]	43,136.787824	–	–
MGDE [37]	43,184.465450	–	–
BBOSB [29]	43,017.9597	–	–
MILP–IPM [28]	43,084	195.2668	0.87
HIGA [25]	43,125.365	194.804	2.06
BBPSO [26]	43,222.7	–	1.48
LDISS-2 [24]	43,213	–	3.17

1,016,311$/day, 1,024,163$/day, 1,017,530.3328$/day, 1,018,473.380$/day, 1,018,166$/day, 1,027,247.78$/day, 1,031,077$/day, 1,023,691.11$/day and 1,027,334$/day, respectively. The cost given by HPSTCO is 1,015,438.967 $/day which is lesser than the best cost produced by the other hybrids. Hence, from cost point of view the proposed HPSTCO is better than others. The convergence time is 1.79 min; the convergence rate (only eight iterations) as evident from Fig. 7 is quite acceptable.

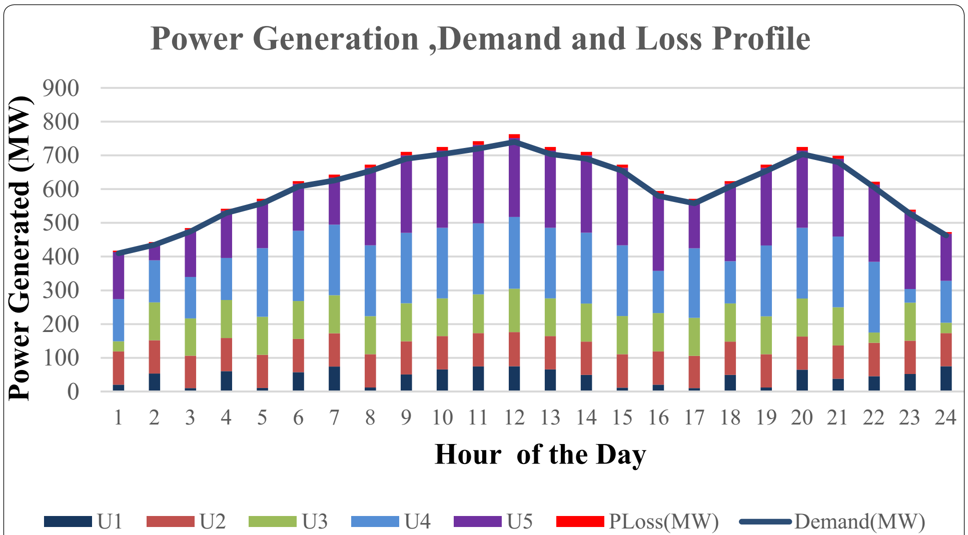

Fig. 2 Power generation, power demand and loss profile over 24-h period in a day for test case 1

Table 7 Characteristic input data of the 10-unit system

Unit	a_i ($)	b_i ($/MWh)	c_i ($/MW^{2}h)	e_i ($/h)	f_i (rad/MW)	P_{min} (MW)	P_{max} (MW)	UR (MW/h)	DR (MW/h)
U_1	958.2	21.6	0.00043	450	0.041	150	470	80	80
U_2	1313.6	21.05	0.00063	600	0.036	135	460	80	80
U_3	604.97	20.81	0.00039	320	0.028	73	340	80	80
U_4	471.6	23.9	0.0007	260	0.052	60	300	50	50
U_5	480.29	21.62	0.00079	280	0.063	73	243	50	50
U_6	601.75	17.87	0.00056	310	0.048	57	160	50	50
U_7	502.7	16.51	0.00211	300	0.086	20	130	30	30
U_8	639.4	23.23	0.0048	340	0.082	47	120	30	30
U_9	455.6	19.58	0.10908	270	0.098	20	80	30	30
U_{10}	692.4	22.54	0.00951	380	0.094	55	55	30	30

Table 8 *B*-matrix coefficients (per MW)

$$B_{ij} = \begin{bmatrix} 8.7 & 0.43 & -4.61 & 0.36 & 0.32 & -0.66 & 0.96 & -1.6 & 0.8 & -0.1 \\ 0.43 & 8.3 & -0.97 & 0.22 & 0.75 & -0.28 & 5.04 & 1.7 & 0.54 & 7.2 \\ -4.61 & -0.97 & 9 & -2 & 0.63 & 3 & 1.7 & -4.3 & 3.1 & -2 \\ 0.36 & 0.22 & -2 & 5.3 & 0.47 & 2.62 & -1.96 & 2.1 & 0.67 & 1.8 \\ 0.32 & 0.75 & 0.63 & 0.47 & 8.6 & -0.8 & 0.37 & 0.72 & -0.9 & -0.69 \\ -0.66 & -0.28 & 3 & 2.62 & -0.8 & 11.8 & -4.9 & 0.3 & 3 & -3 \\ 0.96 & 5.04 & 1.7 & -1.96 & 0.37 & -4.9 & 8.24 & -0.9 & 5.9 & -0.6 \\ -1.6 & 1.7 & -4.3 & 2.1 & 0.72 & 0.3 & -0.9 & 1.2 & -0.96 & 0.56 \\ 0.8 & 0.54 & 3.1 & 0.67 & -0.9 & 3 & 5.9 & -0.96 & 0.93 & -0.3 \\ -0.1 & 7.2 & -2 & 1.8 & 0.69 & -3 & -0.6 & 0.56 & -0.3 & 0.99 \end{bmatrix} \times 10^{-5}$$

Test case 4: 30-unit system

In this test case, input data [15] are obtained by tripling the data of 10-unit system given in Tables 7 and 8. The load demand of the system as divided into 24 dispatch intervals is given in Table 14. In this case, the VPL effects, ramp rate constraints and generation limits are considered. DED results obtained by MATLAB simulation are presented in

Table 9 Load demand for 24 h for 10-unit system

Time (h)	Load (MW)	Time (h)	Load (MW)	Time (h)	Load (MW)	Time (h)	Load (MW)	Time (h)	Load (MW)	Time (h)	Load (MW)
1	1036	5	1480	9	1924	13	2072	17	1480	21	1924
2	1110	6	1628	10	2072	14	1924	18	1628	22	1628
3	1258	7	1702	11	2146	15	1776	19	1776	23	1332
4	1406	8	1776	12	2220	16	1554	20	2072	24	1184

Table 10 Simulation results for optimal 24-h cost, hourly generation schedule and loss of 10-unit test system with transmission loss

Hour	U_1	U_2	U_3	U_4	U_5	U_6	U_7	U_8	U_9	U_{10}	Loss (MW)	Cost ($/h)
1	150.538	136.2751	161.3644	62.8806	123.3856	121.6485	130	85.3465	20.3405	55	10.7793	28,825.0096
2	226.5922	222.5594	74.7063	60.7017	169.9639	121.304	130	47.2415	20.4012	55	16.4703	30,365.2647
3	226.7336	135.023	309.1783	63.4571	123.4365	125.2567	130	85.1702	20.1536	55	14.409	33,537.9408
4	380.2625	135.0776	299.266	68.7463	126.1347	124.9338	130	85.2673	20.4929	55	18.1811	36,913.0554
5	380.2509	221.5574	294.8976	61.8072	172.4872	122.1908	130	47.8042	20.1021	55	25.0973	38,408.517
6	456.2443	309.2544	290.2498	61.3994	172.1029	121.6925	129.9107	47.3854	20.3907	55	34.6303	41,855.1792
7	381.8778	396.8053	297.7609	121.3926	124.4519	126.9332	129.7198	85.3978	20.0366	55	36.3759	43,661.4048
8	456.1594	397.6526	300.6672	113.917	172.8827	126.1655	130	47.034	20.1504	55	42.6289	45,387.5997
9	379.4479	397.9094	300.3963	300	218.9901	122.1732	128.3873	47.4176	20.1142	55	44.8359	49,104.8605
10	456.302	400.3316	297.3236	300	219.9175	122.5257	130	120	20.0282	55	48.4287	52,644.5974
11	456.3859	397.2392	307.4963	300	222.4828	160	129.7913	120	49.9987	55	51.3943	54,662.013
12	456.6368	460	322.4282	300	222.9124	160	129.6536	120	52.1062	55	56.7371	56,921.1764
13	456.764	395.115	298.4546	300	223.776	122.3104	129.6816	120	20.0429	55	48.1445	52,579.0891
14	456.5067	309.5792	297.3688	240.5994	173.7713	160	130	120	20.1401	55	37.9655	49,105.379
15	456.9433	397.7644	299.3203	62.4159	222.9798	128.4779	130	48.0664	20.5892	55	44.5572	45,333.725
16	456.5742	222.7853	292.8654	61.2936	174.2671	122.4949	130	47.2974	21.4309	55	29.0088	40,162.1122
17	380.009	222.0942	300.6441	62.3416	124.3203	123.3403	129.6503	85.8338	20.3664	55	22.6001	38,443.1731
18	379.905	313.0685	296.7714	118.9663	175.6568	124.6082	129.6456	47.268	20.0601	55	31.95	41,946.8421
19	456.1337	307.6835	292.1048	120.2049	227.0954	121.1324	130	84.2329	20.2607	55	36.8484	45,442.0016
20	456.7193	397.3807	298.2643	300	221.5664	122.1813	130	120	20.1409	55	48.2531	52,563.7115
21	455.5566	398.0045	294.7719	185.9709	223.8794	121.7237	130	85.4028	20.1746	55	45.4843	48,923.3223
22	455.8407	309.6992	290.3381	60.5314	172.5354	122.415	130	47.0892	20.2267	55	34.6757	41,813.7803
23	226.731	135.2984	292.7148	60.3422	222.9619	122.3569	129.0647	85.549	20.4408	55	17.4596	35,120.92
24	304.4504	136.3426	189.0092	61.4709	128.558	127.758	129.72	47.1861	20.1967	55	14.692	32,009.5283
Total loss (MW)											811.6073	–
Total cost ($/day)												1,035,730.203

Table 11 Cost and computation time comparison of optimization results in case 2

Method	Best cost ($/day)	Mean cost ($/day)	Worst cost ($/day)	Loss (MW)	Computation time (min)
HPSTCO	1,035,730.203	1,036,236.632	1,036,987.342	811.6073	1.85
MILP–IPM [28]	1,040,676	–	–	882.7374	1.12
BBOSB [29]	1,038,362.014	1,039,968.7779	1,041,538.9613	819.8625	–
HIGA [25]	1,041,087.802	1,042,980.147	1,044,926.653	853.53	3.8
LDISS-2 [24]	1,039,083	1,041,091	1,042,630	812.5324	13.64
Hybrid EP–SQP [32]	1,035,748	–	–	–	20.51
MVMO–SH [34]	1,036,260	–	–	–	3.7

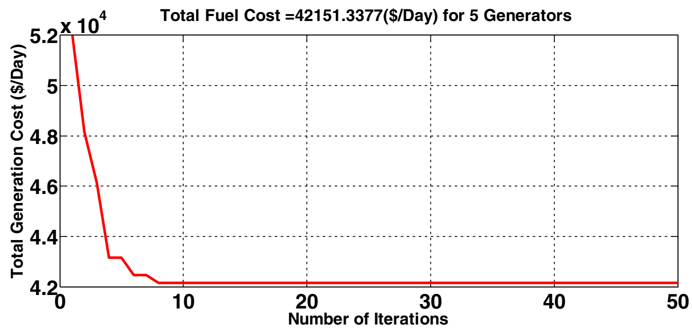

Fig. 3 Convergence graphs of test case 1

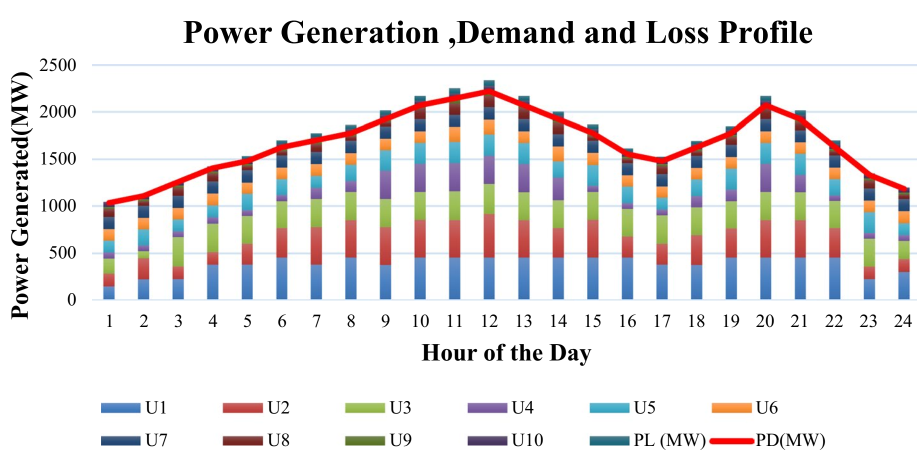

Fig. 4 Power generation, power demand and loss profile over 24-h period in a day for test case 2

Table 15, the graphical representation of which is shown in Fig. 8. The fuel cost obtained by the proposed method is 1,051,964.4$/day.

In Table 16, the simulation result of proposed HPSTCO is compared with other recently published hybrid methods, namely BBOSB [29], FA–PSOTVAC [27], BBPSO [26], HIGA [25], LDISS [24], HHS [15] and hybrid EP–SQP [17], which are 3,054,190.6032$/day, 3,105,700$/day, 3,062,144$/day, 3,055,435.068$/day, 3,051,259.9486$/day, 3,057,313.39$/day and 3,159,204$/day, respectively. In comparison,

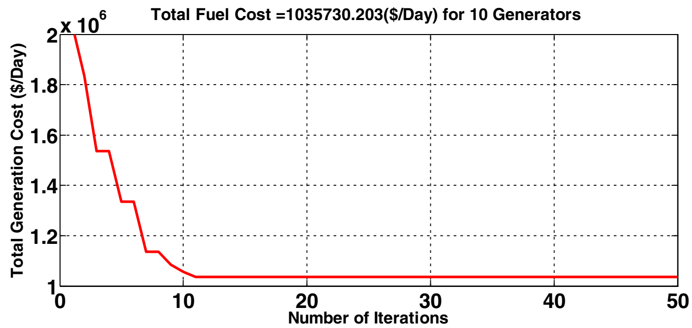

Fig. 5 Convergence graphs of test case 2

the cost yielded by HPSTCO is 3,051,105.813 $/day only. The average execution time required for one complete solution was 4.25 min, which is better than time required by many other methods. Figure 9 shows that the technique takes only nine iterations to reach to steady state which is a good convergence characteristics compared to other methods.

Conclusion

In this paper, HPSTCO has been taken up as a cost minimization and schedule optimization method for 24-h time interval in four test cases representing small- to medium-scale thermal power generation system. Such a hybrid method was never implemented before for dynamic emission dispatch. A synergistic combination of two popular techniques for optimization has been able to mitigate the limitations of the individual techniques. Besides improving the convergence rate, the exploration of neighborhood area for finding local optima has bettered. The hybrid has overcome the problem of convergence to local optima and yields a good globally optimal solution.

From the trial runs of the test cases, it can be concluded that HPSTCO is reliable, robust and can consistently provide high-quality solutions of DED considering practical operational constraints, such us valve point effects and multiple fuel changes. The convergence characteristics of HPSTCO are also quite acceptable though not one of the best.

The performance of HPSTCO in terms of cost minimization and dispatch schedule optimization when compared with different other hybrids is found to be quite competitive and can be safely used as an effective metaheuristic for small to medium scale, simple to complex DED problems. In future, this hybrid method can be used to solve the problem of dynamic economic emission dispatch (DEED) problem, multi-objective economic dispatch (MOED) and multi-objective economic emission dispatch (MEED) problem and multi-area economic dispatch (MAED) problem of large-scale power generation system. The HPSTCO method can also be applied to find the impact on optimum dispatch problem of renewable energy like solar and wind energy.

Table 12 Simulation results for optimal 24-h cost and loss of 10-unit test system without transmission loss

Hour	U_1	U_2	U_3	U_4	U_5	U_6	U_7	U_8	U_9	U_{10}	Cost ($/h)
1	303.2419	136.6306	81.0279	64.1764	75.2884	122.5108	130	47.9478	20.1763	55	28,513.4181
2	151.467	136.838	305.967	62.9884	74.4954	123.474	130	47.9136	21.8575	55	30,000.1586
3	150.7314	222.8537	282.4076	65.8081	122.359	121.5551	130	87.0882	20.1968	55	33,301.1277
4	306.0519	221.9859	305.5451	74.5658	122.8544	122.4764	130	47.2	20.3207	55	36,455.6487
5	226.4725	400.7468	303.5602	61.5428	74.4874	122.9884	130	84.8855	20.3163	55	37,947.1907
6	382.4602	392.2271	299.1057	61.1304	121.7389	118.801	130	47.1286	20.4081	55	41,171.6082
7	303.3307	391.3567	300.3431	60.5259	224.8973	130.1784	130	85.5744	20.7934	55	42,890.2074
8	379.9733	397.1637	301.2939	61.2164	221.3332	125.6823	129.8462	84.157	20.3339	55	44,358.2277
9	380.0662	397.244	297.822	181.0074	223.5854	119.4004	129.245	120	20.6297	55	48,001.4959
10	457.3512	397.8553	310.5531	229.8779	226.7787	160	130	84.2324	20.3514	55	51,620.278
11	455.9245	397.08	320.5675	300	224.6006	125.9658	130	85.4729	51.3888	55	53,333.0741
12	454.0273	460	303.4063	300	221.0132	160	130	84.9784	51.5747	55	55,405.1741
13	456.9087	399.2232	295.9235	240.9988	222.8825	130.5953	130	120	20.468	55	51,453.3554
14	456.7427	393.4524	302.4211	172.2332	223.6674	122.8211	130	47.4093	20.2528	55	45,890.9537
15	456.8321	394.9745	296.9837	64.8243	177.767	130.8763	129.649	49.0117	20.0818	55	44,488.1713
16	301.0479	397.3489	294.6123	61.1527	124.6318	121.6107	130	47.8794	20.7164	55	39,459.5558
17	305.4404	311.1491	295.7393	60.259	126.9807	127.6389	130	47.1622	20.6305	55	37,928.7241
18	303.3246	395.2245	295.8566	64.3487	174.5676	142.0629	129.6454	47.9583	20.0114	55	41,233.5201
19	315.2346	397.7267	304.9993	120.2766	222.9819	124.2927	129.9458	85.5351	20.0073	55	44,660.622
20	462.6054	393.2256	302.7059	300	225.7948	135.4818	130	47.0972	20.0892	55	51,604.9223
21	456.8711	394.7525	340	62.9638	223.4313	155.7431	130	84.6267	20.6115	55	48,040.6009
22	304.8849	397.0071	310.2237	60.565	171.7679	130.8583	130	47.6765	20.0165	55	41,158.4624
23	226.738	220.8902	279.8324	60.1397	172.9954	117.2937	130	48.5199	20.5907	55	34,854.0116
24	227.1409	219.3182	183.8525	66.6019	73.8632	122.1084	130	85.3044	20.8105	55	31,668.4584
Total cost ($/day)											1,015,438.967

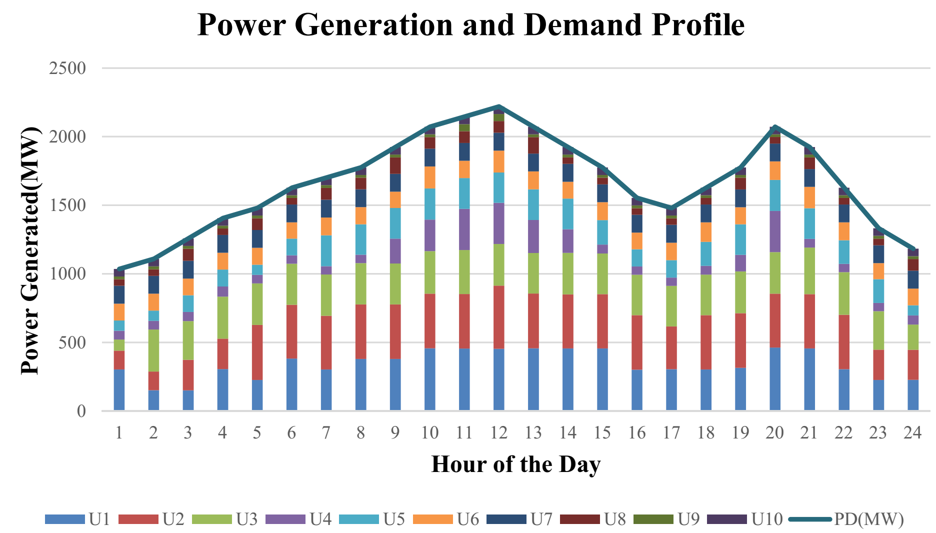

Fig. 6 Power generation and power demand profile over 24-h period in a day for test case 3

Table 13 Cost and computation time comparison of optimization results in case 3

Method	Pop size	Iteration	Best cost ($)	Mean cost ($)	Worst cost ($)	Computation time (min)
Proposed method	50	13	1,015,438.967	1,016,235.527	1,017,983.495	1.79
SaANS–SDP with VRS [38]	240	–	1,041,100	1,044,300	1,047,140	–
MVMO–SH [34]	–	–	1,015,903	–	–	2.8
ADE–SA [35]	–	–	1,016,412.81	–	–	1.805
MILP–IPM [28]	–	–	1,016,311	–	–	0.08
FA–PSOTVAC [27]	50	200	1,024,163	1,794,400	13,793,000	8.4934
BBOSB [29]	–	–	1,017,530.3328	1,018,487.8504	1,019,954.8584	–
HIGA [25]	–	–	1,018,473.380	1,019,328.460	1,022,283.542	3.53
LDISS-2 [24]	50	30	1,018,166	1,019,344	1,020,030	9.25
Hybrid GSA–SQP [22]	–	6000	1,027,247.78	–	–	5.21
Hybrid DE [19]	–	–	1,031,077	–	–	–
EPSO–GM [33]	20	20,000	1,023,691.11	1,026,034.14	1,029,736.00	
Hybrid PSO–SQP [17]	–	–	1,027,334	–	–	18.12

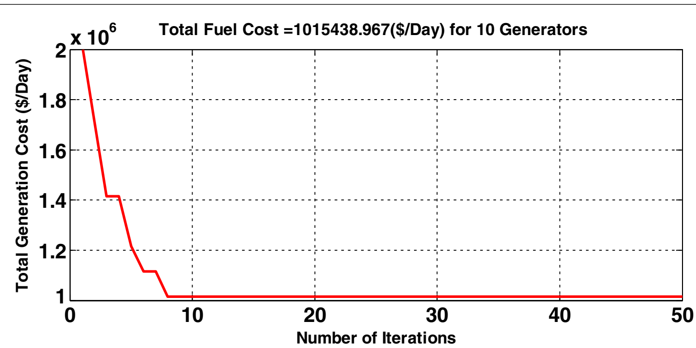

Fig. 7 Convergence graph of test case 3

Table 14 Optimal 24-h schedule of 30-unit test system

Time (h)	Load (MW)	Time (h)	Load (MW)	Time (h)	Load (MW)	Time (h)	Load (MW)	Time (h)	Load (MW)	Time (h)	Load (MW)
1	3108	5	4440	9	5772	13	6216	17	4440	21	5772
2	3330	6	4884	10	6216	14	5772	18	4884	22	4884
3	3774	7	5106	11	6438	15	5328	19	5328	23	3996
4	4218	8	5328	12	6660	16	4662	20	6216	24	3552

Table 15 Simulation results for optimal 24-h cost and loss of 30-unit test system

Hour	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Demand (MW)	**3108**	**3330**	**3774**	**4218**	**4440**	**4884**	**5106**	**5328**
Unit								
U_1	156.6361	152.7871	227.016	306.5455	299.7807	381.0959	451.3755	458.2256
U_2	137.3883	216.7005	206.1411	223.0697	221.1078	386.3257	402.5933	386.7338
U_3	183.5077	77.8266	185.1793	171.2173	317.6486	200.4145	313.0271	302.1198
U_4	67.3196	64.0967	69.648	61.7487	120.2736	80.0232	62.9823	118.2491
U_5	115.0444	172.8201	73.5132	172.2926	221.1498	124.6679	222.2963	74.3905
U_6	160	160	112.7483	125.3848	132.9651	119.5061	160	113.516
U_7	130	130	130	130	130	130	130	130
U_8	53.3852	47.1425	56.6072	77.1069	48.3576	86.6343	83.3684	87.9544
U_9	22.4203	51.7661	22.3595	21.438	21.7893	20.0857	52.0115	21.0687
U_{10}	55	55	55	55	55	55	55	55
U_{11}	150.7162	164.6547	225.9203	244.0467	381.6101	463.5108	302.305	453.867
U_{12}	137.4399	231.2731	222.998	396.115	223.1169	307.9706	311.7996	392.7023
U_{13}	74.6049	184.9427	75.5644	183.6572	196.0591	300.8891	303.7	324.501
U_{14}	61.3152	92.4835	114.8251	77.1772	61.3068	68.8396	67.8669	113.1989
U_{15}	123.6486	76.8173	74.6512	118.4038	226.0671	212.2472	180.3214	123.0396
U_{16}	131.1538	160	122.9518	121.8622	160	122.5299	126.7631	128.3935
U_{17}	130	130	130	130	130	130	130	130
U_{18}	49.9019	47.0916	120	48.1175	83.8864	83.3179	53.7866	120
U_{19}	25.6158	22.3027	21.7725	21.7898	52.4353	20.2422	21.5846	21.3338
U_{20}	55	55	55	55	55	55	55	55
U_{21}	153.6568	155.195	229.3618	210.9768	310.5565	226.3367	305.6144	297.1359
U_{22}	136.1982	136.6017	308.8311	397.4244	145.145	408.2237	314.1424	401.4987
U_{23}	175.8338	96.2927	313.3874	300.1877	325.1337	290.9892	292.0067	292.3057
U_{24}	121.9591	60.9382	66.9515	60.7523	66.3776	70.3569	84.2282	116.8452
U_{25}	122.6434	175.122	173.2034	130.4837	73.722	122.5648	175.1315	223.0581
U_{26}	114.4669	124.0518	124.7801	124.9592	124.8547	121.8398	123.1107	120.0564
U_{27}	130	130	130	130	130	130	130	130
U_{28}	56.7627	82.4834	49.5629	47.4806	48.9207	88.7115	120	55.3063
U_{29}	21.3812	21.61	21.026	20.7623	22.7356	21.6768	20.9844	27.4997
U_{30}	55	55	55	55	55	55	55	55
Cost per unit ($/day)	86,712.26	91,749.84	100,913.2	101,210.8	114,233.9	123,976.6	129,921.8	135,035.5
Hour	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Demand (MW)	**5772**	**6216**	**6438**	**6660**	**6216**	**5772**	**5328**	**4662**
Unit								
U_1	455.4156	454.461	468.1167	456.6197	457.9405	459.2393	379.9946	227.0437
U_2	395.4997	389.5283	389.5237	395.1748	393.8168	395.647	460	396.5805
U_3	340	310.1666	340	304.6925	297.1566	307.9964	294.5595	308.1237
U_4	236.9211	124.4537	183.2899	300	300	119.3569	67.6225	63.5592
U_5	122.9109	221.9668	219.9002	243	220.2901	229.1454	230.3565	77.5625
U_6	160	160	160	122.5459	160	119.7974	160	120.076
U_7	130	130	130	130	130	130	130	130
U_8	87.3227	120	120	120	84.5263	84.5242	81.2069	48.4701
U_9	48.8748	47.7972	22.5313	20.3959	22.1823	24.5704	24.2413	20.5996
U_{10}	55	55	55	55	55	55	55	55
U_{11}	470	456.9818	470	470	470	384.6041	451.6924	384.368
U_{12}	314.5853	395.3526	404.0876	460	460	394.5032	219.1056	306.9112
U_{13}	330.2565	340	304.0497	340	340	340	294.2354	313.2232

Table 15 (continued)

Hour	9	10	11	12	13	14	15	16
Demand (MW)	**5772**	**6216**	**6438**	**6660**	**6216**	**5772**	**5328**	**4662**
U_{14}	173.7543	300	300	300	123.5559	63.2327	61.1228	122.652
U_{15}	174.5582	243	224.3059	243	172.4131	228.2889	127.9415	221.4074
U_{16}	124.267	160	160	160	160	160	121.8913	116.3018
U_{17}	130	130	130	130	130	130	130	130
U_{18}	120	120	82.8362	83.8627	120	50.8432	120	47.224
U_{19}	20.5503	20.5962	50.4151	26.8082	23.4197	28.5902	45.0589	23.0396
U_{20}	55	55	55	55	55	55	55	55
U_{21}	454.5898	470	470	455.0604	470	462.2676	456.3738	302.14
U_{22}	389.9642	460	398.0501	460	308.5902	460	398.653	309.5274
U_{23}	321.3959	306.968	340	300.6481	293.9653	303.3105	312.6154	312.887
U_{24}	121.2386	62.9746	300	300	300	118.3226	60.4335	69.8464
U_{25}	164.1937	240.9904	173.8243	243	227.3967	218.3714	172.7586	118.3036
U_{26}	120.0645	117.9701	160	160	160	160	129.4894	127.6315
U_{27}	130	130	130	130	130	130	130	130
U_{28}	49.7019	85.6586	120	120	47.3427	84.21	49.3565	49.1868
U_{29}	20.9349	52.1342	22.0693	20.1918	48.4037	20.1785	54.2906	20.3349
U_{30}	55	55	55	55	55	55	55	55
Cost per unit ($/day)	143,473.9	155,980	161,285.4	162,735.1	154,869.6	142,933.5	134,027.4	114,389.1
Hour	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**
Demand (MW)	**4440**	**4884**	**5328**	**6216**	**5772**	**4884**	**3996**	**3552**
Unit								
U_1	227.1256	154.7243	460.3588	451.4051	458.4054	462.8513	377.7089	380.4584
U_2	141.7164	306.3977	398.8181	396.6633	460	399.1278	142.5676	309.7039
U_3	308.8095	309.1892	290.493	295.4164	340	298.7534	294.1235	199.6546
U_4	65.5958	61.7759	123.4598	300	122.1345	119.4114	66.4421	69.2744
U_5	216.1024	232.908	174.3895	183.0822	178.7955	75.9503	120.1514	74.7298
U_6	135.3044	123.6005	129.0219	160	160	160	125.6952	123.2682
U_7	130	130	130	130	130	130	130	130
U_8	47.5604	47.8949	82.6273	49.0022	120	48.3291	54.128	47.3901
U_9	52.2878	20.5378	23.1839	24.1483	32.3267	21.307	20.5975	21.4514
U_{10}	55	55	55	55	55	55	55	55
U_{11}	306.0758	381.7282	299.7631	470	378.9605	370.6101	296.1123	153.4444
U_{12}	392.9375	397.905	401.1924	460	398.7344	396.6708	394.3747	136.7718
U_{13}	294.2377	304.3464	303.4469	289.4879	298.3372	308.4771	185.7545	78.6511
U_{14}	101.2546	73.6006	182.9236	122.8943	120.8767	66.1119	63.8072	131.6552
U_{15}	74.0031	123.2715	73.1885	219.0339	243	116.8524	169.4518	119.991
U_{16}	127.8285	124.1132	131.9019	120.4227	134.9719	117.4291	129.3315	122.4987
U_{17}	130	130	130	130	130	130	130	130
U_{18}	47.8673	85.1634	119.2559	86.6209	120	85.4857	50.3169	49.0406
U_{19}	22.8533	53.8519	22.8715	26.6395	23.2127	46.1556	50.992	51.7326
U_{20}	55	55	55	55	55	55	55	55
U_{21}	227.3557	456.9452	376.3048	470	380.5054	231.8649	151.9123	230.8197
U_{22}	388.5731	315.1246	397.0103	398.8798	387.1446	310.4801	215.9083	221.6757
U_{23}	198.615	185.1819	308.4753	308.0651	292.2205	281.615	195.0372	73.2621
U_{24}	73.8779	73.1462	69.9627	300	115.5469	106.3535	65.2368	66.9052
U_{25}	224.3901	223.42	172.1393	228.3715	221.6636	74.7023	73.0682	121.313
U_{26}	137.1451	160	160	160	160	121.6104	127.5902	119.6289
U_{27}	130	130	130	130	130	130	130	130

Table 15 (continued)

Hour	17	18	19	20	21	22	23	24
Demand (MW)	4440	4884	5328	6216	5772	4884	3996	3552
U_{28}	49.8953	86.5096	50.5047	120	47.2221	83.3086	49.1244	47.7603
U_{29}	23.5878	27.6636	21.707	20.867	22.9416	25.5422	21.5677	45.9188
U_{30}	55	55	55	55	55	55	55	55
Cost per unit ($/day)	114,449.5	124,822	132,574.6	154,810.8	144,475.6	125,095.4	105,455.3	95,974.6
Total best cost (for 30 units) = 3,051,105.8134 ($/day)								
Total mean cost (for 30 units) = 3,053,546.734 ($/day)								
Total worst cost (for 30 units) = 3,055,657.243 ($/day)								

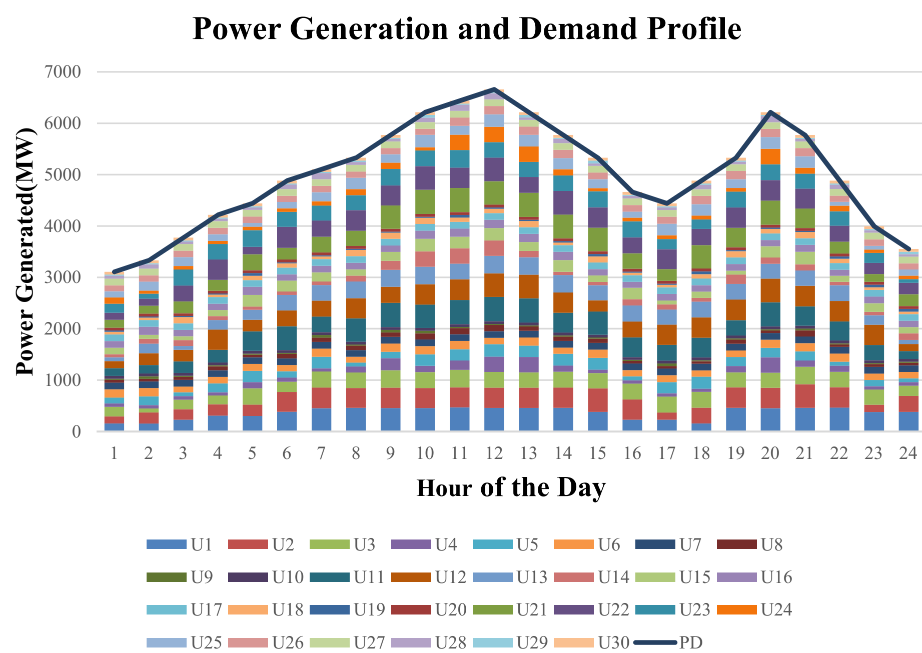

Fig. 8 Power generation and power demand profile over 24-h period in a day for test case 4

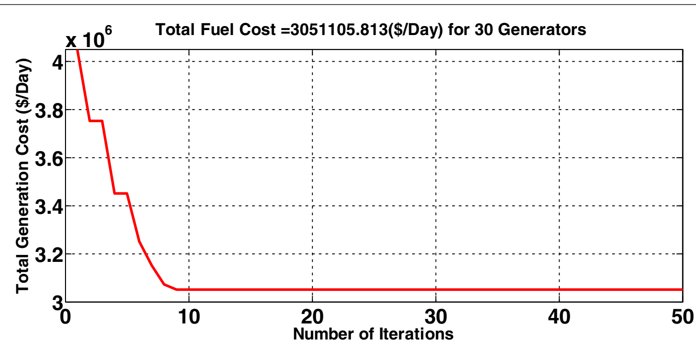

Fig. 9 Convergence graphs of test case 4

Table 16 Cost comparison of optimization results for 30-unit test system

Method	Pop size	Max iteration	Minimum cost ($)	Mean cost ($)	Maximum cost ($)	Time (min)
Proposed method	100	50	3,051,105.813	3,053,546.734	3,055,657.243	4.25
BBOSB [29]	–	–	3,054,190.6032	3,055,431.4392	3,057,772.7211	–
BBO–PSOTVAC [27]	50	1000	3,105,700	–	3,122,200	3.4018
BBPSO [26]	–	–	3,062,144	3,067,277	–	–
HIGA [25]	–	–	3,055,435.068	3,058,126.233	3,066,754.92	–
LDISS-2 [24]	100	50	3,051,259.9486	3,054,149.2516	3,056,051.0341	25.46
HHS [15]	–	25	3,057,313.39	–	–	27.65
Hybrid EP–SQP [17]	–	–	3,159,204	3169,093	–	–

Abbreviations

Formulation of dynamic economic dispatch (DED)

a_i (in $/h), b_i (in $/MWh) and c_i (in $/MW^2 h): fuel cost coefficients of *i*th unit; e_i (in $/h) and f_i (in 1/MW): valve point loading coefficients of the *i*th unit; $C_{i,t}$ ($P_{i,t}$): cost of producing real power output $P_{i,t}$ at time *t*; *n*: number of dispatch-able power generating units; P_D: total load demand; $P_{i,t}$: real power output of *i*th unit during time interval *t*; UR_i, DR_i: upper ramp and down ramp rate limits of the *i*th generator; P_i^{lower}, P_i^{upper}: lower and upper boundary generation limits of *i*th unit; P_L (*t*): transmission line losses at time *t*; *T*: total number of hours of operation; C_T: total number of hours of operation; n_i: No. of POZ of unit *i*; $B_{i,j}$, $B_{0,i}$, $B_{0,0}$: B-loss coefficient.

Optimization algorithm: particle swarm optimization (PSO)

X_i: position of a particle *i*; $\vec{V}_i k$: velocity vectors for particle *i* in the previous iterations; $\vec{V}_i\{k+1\}$: velocity vectors for particle *i* in the current iterations; *D*: dimension; *S*: d-dimensional search space; c_1, c_2: positive constants; r_1, r_2: random parameters of uniform distribution; X_{pi}: local best (pbest) position of a particle *i*; X_{gi}: global best (gbest) position of a particle *i*; V_i: velocity of particle *i*; *w*: inertia weight.

Termite colony optimization (TCO)

X_i: cost/fitness function value for each position of the termite; *M*: size of the termite population; *D*: dimension; $fit(X_i)$: fitness function value for each position of termite; R_w: random walk function of current position radius of search; B_i: best local position of termite; $\tau_i\{t-1\}$: pheromone level of *i*th termite at the current locations; $\tau_i\{t\}$: pheromone level of *i*th termite at the previous locations; ρ: evaporation rate of pheromone; w_b, r_b: probabilistically controls parameters for attracting the termite toward local best position; X_{max}, X_{min}: maximum and minimum limit of search space along a dimension.

HPSTCO

η: switching time; n_1: iterations of PSO; n_2: iterations of TCO; c_1 & c_{12}: acceleration coefficient; γ: constriction factor; $P = (p_1, p_2, ...pn)$: priority list; *s*: population size; Lb & Ub: lower and upper bound of solution space.

Acknowledgements
Data were taken from the IEEE Congress on Evolutionary Computation 2011 Competition on Testing Evolutionary Algorithms on Real World Optimization Problems; Lab and MATLAB software were used in RCC Institute of Information Technology, Kolkata, India.

Authors' contributions

The novelty of the present study lies in the fact that PSO and TCO together, i.e., their hybrid combination (HPSTCO), have never been tried before to optimize small- to large-scale economic load dispatch problem. The contribution of the paper lies in the fact that it has established by reporting four distinct test cases and comparing the result with other hybrid methods for each of these four test cases that the HPSTCO hybrid is quite effective and advantageous in dealing with small-scale (5- and 10-unit) as well as medium-scale (30-unit) DED problem. In large-scale systems with higher-capacity turbines, the fuel cost function is highly non-smooth and non-convex and contains discontinuous values at each boundary forming multiple local optima. The complexity of the problem also increases significantly with the increase in the number of generating units because of their combinatorial nature. The present work has tackled this challenge nicely having no earlier precedence of application of this particular (HPSTCO) hybrid optimization mechanism. Therefore, it can be said that this paper introduces a new metaheuristics in DED with significant results. DS and AM performed the literature survey and hybridization analysis and selected hybrid metaheuristic components for the present application after considering different feasible combinations of constraints and also reviewed the work finally in view of the reviewer's comments and added necessary changes, graphs, etc.; DS and KS developed the MATLAB coding of the hybrid metaheuristics, executed trials by altering the parameters of the programs and reported results; DS and SM analyzed and interpreted the results and compared performance with similar other methods. All authors have read and approved the revised manuscript.

Funding
This study had no funding from any resource.

Data availability
The data that support the findings of this study are available from the corresponding author upon reasonable request.

Competing interests
The authors declares that they have no competing interests.

Author details
[1] Department of Electrical Engineering, RCC Institute of Information Technology, Canal South Road, Beliaghata, Kolkata, West Bengal 700015, India. [2] RCC Institute of Information Technology, Kolkata, India. [3] Saroj Mohan Institute of Technology, Hooghly, India. [4] Jadavpur University, Jadavpur, Kolkata, India.

Received: 4 September 2019 Accepted: 21 January 2020
Published online: 10 February 2020

References

1. Zaman MF, Elsayed SM, Ray T, Sarker RA (2016) Evolutionary algorithms for dynamic economic dispatch problems. IEEE Trans Power Syst 31(2):1486–1495. https://doi.org/10.1109/TPWRS.2015.2428714
2. Chandram K, Subrahmanyam N, Sydulu M (2008) Brent method for dynamic dispatch with transmission losses. Paper presented at the IEEE/PES transmission and distribution conference and exposition. Chicago
3. Granelli GP, Marannino P, Montagna M, Silvestri A (1989) A fast and efficient gradient projection algorithm for dynamic generation dispatching. IEEE Proc C Gener Transm Distrib 136(5):295–302
4. Raj VM, Chanana S (2014) Analysis of unit commitment problem through Lagrange relaxation and priority listing method. Paper presented at the 6th IEEE power India international conference (PIICON). Delhi
5. Somuah CB, Khunaizi N (1990) Application of linear programming re-dispatch technique to dynamic generation allocation. IEEE Trans Power Syst 5:20–26
6. Travers D, Kaye RJ (1998) Dynamic dispatch by constructive dynamic programming. IEEE Trans Power Syst 13(1):72–78
7. Whei-Min L, Shi-Jaw C (2002) Bid-based dynamic economic dispatch with an efficient interior point algorithm. Int J Electr Power Energy Syst 24(1):51–57
8. Secui DC (2015) A method based on the ant colony optimization algorithm for dynamic economic dispatch with valve-point effects. Int Trans Electr Energy Syst 25(2):262–287. https://doi.org/10.1002/etep.1841
9. Yuan X, Su A, Yuan Y, Nie H, Wang L (2009) An improve PSO for dynamic load dispatch of generators with valve-point effects. Energy 34(1):67–74
10. Niknam T, Golestaneh F (2012) Enhanced adaptive particle swarm optimisation algorithm for dynamic economic dispatch of units considering valve-point effects and ramp rates. IET Gener Transm Distrib 6(5):424–435
11. Ghani Ab MR, Hussein ST, Amin NHM, Jano Z, Sutikno T (2017) Dynamic economic dispatch problems: PSO approach. Telkomnika 15(1):48–61
12. Daniela L, Chaturvedia K, Kolhe M L (2018) Dynamic economic load dispatch using levenberg marquardt algorithm. In: Abstracts of the 4th international symposium on hydrogen energy, renewable energy and materials, HEREM, 13–15 June 2018, Bangkok, Thailand, Energy Procedia vol 144, pp 95–103
13. Jebaraj L, Venkatesan C, Soubache I (2017) Application of differential evolution algorithm in static and dynamic economic or emission dispatch problem: a review. Renew Sustain Energy Rev 77:1206–1220
14. Hemamalini S, Simon SP (2011) Dynamic economic dispatch using artificial immune system for units with valve point effect. Int J Electr Power Energy Syst 33(4):868–874. https://doi.org/10.1016/j.ijepes.2010.12.017
15. Pandi VR, Panigrahi BK (2011) Dynamic economic load dispatch using hybrid swarm intelligence based harmony search algorithm. Expert Syst Appl 38:8509–8514
16. Niknam T, Golestaneh F (2013) Enhanced bee swarm optimization algorithm for dynamic economic dispatch. IEEE Syst J 7(4):754–762. https://doi.org/10.1109/JSYST.2012.2191831
17. Victoire TAA, Jeyakumar AE (2005) A modified hybrid EP-SQP approach for dynamic dispatch with valve-point effect. Electr Power Energy Syst 27(8):594–601
18. Victoire TAA, Jeyakumar AE (2005) Reserve constrained dynamic dispatch of units with valve point effects. IEEE Trans Power Syst 20(3):1273–1282. https://doi.org/10.1109/TPWRS.2005.851958
19. Yuan X, Wang L, Yuan Y (2009) A hybrid differential evolution method for dynamic economic dispatch with valve-point effects. Expert Syst Appl 36(2):4042–4048
20. Sivasubramani S, Swarup KS (2010) Hybrid SOA–SQP algorithm for dynamic economic dispatch with valve-point effects. Energy 35(12):5031–5036
21. Cai J, Li Q, Li L, Peng H, Yang Y (2012) A hybrid CPSO–SQP method for economic dispatch considering the valve-point effects. Energy Convers Manag 53(1):175–181. https://doi.org/10.1016/j.enconman.2011.08.023
22. Swain RK, Meher KC, Mishra UC (2012) Dynamic economic dispatch using hybrid gravitational search algorithm. Paper presented at the 2nd international conference on the power, control and embedded systems, Allahabad, India. https://doi.org/10.1109/icpces.2012.6508103
23. Elaiw AM, Xia X, Shehata AM (2013) Hybrid DE–SQP and hybrid PSO–SQP methods for solving dynamic economic emission dispatch problem with valve-point effects. Electr Power Syst Res 103(1):92–200. https://doi.org/10.1016/j.epsr.2013.05.015
24. Chen Y, Wen J, Jiang L, Cheng S (2013) Hybrid algorithm for dynamic economic dispatch with valve-point effects. IET Gener Transm Distrib 7(10):1096–1104. https://doi.org/10.1049/iet-gtd.2012.0726

25. Mohammadi-Ivatloo B, Rabiee Aand Soroudi A (2013) Nonconvex dynamic economic power dispatch problems solution using hybrid immune-genetic algorithm. IEEE Syst J 7(4):777–785
26. Zhang Y, Gong D, Geng N, Sun X (2014) Hybrid bare-bones PSO for dynamic economic dispatch with valve-point effects. Appl Soft Comput 18:248–260
27. Hamed M, Mahdad B, Srairi K, Mancer N (2018) Application of a novel modified hybrid algorithm for solving dynamic economic dispatch problem with practical constraints. Adv Electr Electron Eng 16(4):459–470
28. Pan S, Jian J, Yang L (2018) A hybrid MILP and IPM for dynamic economic dispatch with valve point effect. Int J Electr Power Energy Syst 97:290–298
29. Xiong G, Shi D (2018) Hybrid biogeography-based optimization with brain storm optimization for non-convex dynamic economic dispatch with valve-point effects. Energy 157:424–435. https://doi.org/10.1016/j.energy.2018.05.180
30. Hedayatzadeh R, Akhavan SF, Keshtgari M, Akbari R, Ziarati K (2010) Termite colony optimization: a novel approach for optimizing continous problems. Paper presented at the 18th Iraninan international conference on electerical engineering. Isfahan, Iran, pp 1–7. https://doi.org/10.1109/IRANIANCEE.2010.5507009
31. Mohammadi-Ivatloo B, Rabiee A, Soroudi A, Ehsan M (2012) Imperialist competitive algorithm for solving non-convex dynamic economic power dispatch. Energy 44(1):228–240
32. Attaviriyanupap P, Kita H, Tanaka E, Hasegawa J (2002) A hybrid EP and SQP for dynamic economic dispatch with nonsmooth fuel cost function. IEEE Trans Power Syst 17(2):411–416
33. Sriyanyong P (2008) A hybrid particle swarm optimization solution to ramping rate constrained dynamic economic dispatch. World Acad Sci Eng Technol 47:374–379
34. Shouman N, Hegazy YG, Omran WA, Krost G (2018) Hybrid mean variance mapping optimization for dynamic economic dispatch with valve point effects. 19th International scientific conference on electric power engineering (EPE). Brno, Czech Republic, https://doi.org/10.1109/epe.2018.8395997
35. He D, Yang L, Wang Z (2018) Adaptive differential evolution based on simulated annealing for large-scale dynamic economic dispatch with valve-point effects. Math Probl Eng 2018:1–16. https://doi.org/10.1155/2018/4745192
36. Zheng Z, Li J, Han Y (2019) An improved invasive weed optimization algorithm for solving dynamic economic dispatch problems with valve-point effects. J Exp Theor Artif Intell. https://doi.org/10.1080/0952813X.2019.1673488
37. Zou D, Li S, Kong X, Ouyang H, Li Z (2018) Solving the dynamic economic dispatch by a memory-based global differential evolution and a repair technique of constraint handling. Energy 147:59–80
38. Shen X, Wu G, Wang R, Chen H, Li H, Shi J (2018) A self-adapted across neighborhood search algorithm with variable reduction strategy for solving non-convex static and dynamic economic dispatch problems. IEEE Access 6:41314–41324

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.