This article has been peer reviewed and accepted for publication in JMST but has not yet been copyediting, typesetting, pagination and proofreading process. Please note that the publication version of this article may be different from this version. This article now can be cited as doi: 10.6119/JMST-015-1215-1.

OPTIMIZATION OF LARGE-SCALE ECONOMIC DISPATCH WITH VALVE-POINT EFFECTS USING A MODIFIED HYBRID PSO-DSM APPROACH

Chun-Lung Chen[1,*], Yu-Liang Lin[2], and Yen-Chang Feng[1]

Key words: non-convex economic dispatch, valve-point effects, particle swarm optimization, direct search method.

ABSTRACT

This paper proposes a modified hybrid particle swarm optimization (PSO) and the direct search method (DSM) for the solution of large-scale non-convex economic dispatch (NED) problem with valve-point effects. A novel diversity based particle swarm optimization (DPSO) with a fewer iterations required is developed to increase the possibility of generating high quality initial solutions for the DSM. The enhanced direct search method (EDSM) incorporates the parallel nature of evaluation programming into the direct search procedure to enhance its search capacity about global exploration and local optimization using the answer from DPSO as starting points. Many inequality and equality constraints can be handled properly in the direct search procedure. Appropriate setting of control parameters of the proposed hybrid DPSO-EDSM algorithm is also recommended to increase the possibility of occurrence of escaping from local optimal solution. Numerical experiments are included to demonstrate that the proposed hybrid approach can obtain a higher quality solution with better performance than many existing techniques for the large-scale NED application.

I. INTRODUCTION

With increasing of the progressive exhaustion of traditional fossil energy sources and restructuring of the power industry, the non-convex economic dispatch (NED) problem may become a more important issue for achieving the optimal utilization of energy sources in a power system. It is widely recognized that a proper schedule of available generating units may save utilities millions of dollars per year in production costs. The main objective of solving the NED problem is to minimize the total production cost of power plants subjected to the operating constraints of a power system. For simplicity, the fuel cost function for each generation unit in the NED problems has been approximately represented by a quadratic function and is solved using classical calculus-based techniques, such as the lambda dispatch approach, the gradient method and the Newton's method (Wood and Wollenberg, 1996). Unfortunately, the generating units exhibit a greater variation in the fuel cost functions due to the physical operation limitations of power plant components, such as valve-point loading, prohibited operating zones and combined cycle units (Walters and Sheble, 1993). Even in a competitive electrical market environment, generator characteristics can also change with commercial interest, not just physical reality. The classical calculus-based techniques, such as lambda-iteration dispatch method, cannot be directly applied to solve this complicated problem due to its non-smooth fuel cost function. The importance of the NED problem is, thus, likely to increase, and more advanced algorithms for the NED problem are worth developing to obtain accurate dispatch results.

Dynamic programming (DP) is a widely used algorithm which has been proved effective in solving complex NED optimization problems. However, the main problem of the DP methods is the curse of dimensionality (Wood and Wollenberg, 1996) and may lead to sub-optimal solutions (Liang and Glover, 1992). Over the past decade, several optimization algorithms based on stochastic searching techniques, including simulated annealing (SA) (Wong and Fung, 1993), genetic algorithm (GA) (Walters and Sheble, 1993; Lee et al., 2011), evolutionary programming (EP) (Yang et al., 1996; Sinha et al., 2003), particle swarm optimization (PSO) (Gaing, 2003; Lu et al., 2010) and direct search method (DSM) (Chen and Chen, 2001; Chen, 2006) could be used to solve the highly nonlinear NED problem without any restrictions on the shape of the cost functions. Among them, the PSO algorithm has received great attention in solving the NED problem due to its simple concept and easy implementation. With a parallel searching mechanism, the PSO has high probabilities of determining the global or near-global optimal solution for the NED problem. However,

Paper submitted 04/10/14; revised 10/20/15; accepted 12/15/15. Author for correspondence: Chun-Lung Chen (e-mail:cclung@mail.ntou.edu.tw).
[1] *Department of Marine Engineering, National Taiwan Ocean University, Keelung, Taiwan, R.O.C.*
[2] *Engineering Division, Taipei Broadcasting Station, Taipei City Government*

DOI: 10.6119/JMST-015-1215-1

one of the main drawbacks of the PSO is attributed to provide a near-global optimal solution with long computing time for convergence. Recently, the DSM has also received great attention in solving the NED problem due to its flexibility and efficiency. However, the standard DSM has premature convergence problem and easy to be trapped in local optima, especially while handling large-scale NED problems with more local optima and heavier constraints. The degree of complexity of the NED problem is related to the system-size. The larger system-size increases the non-linearity as well as the number of equality and inequality constraints in the NED problem. Therefore, development of better hybrid algorithms is necessary to improve the solution quality and performance for the large-scale NED problem.

Several hybrid optimization methods combining stochastic search techniques and deterministic techniques may prove to be very effective in solving the NED problem (Wong and Wong, 1994; Bhagwan Das and Patvardhan, 1999; Victoire and Jeyakumar, 2004; Lu et al., 2008; Alsumait et al., 2010; Subathra et al., 2015), such as the hybrid evolutionary programming-sequential quadratic programming (EP-SQP), the hybrid particle swarm optimization-sequential quadratic programming (PSO-SQP), the hybrid simulated annealing-direct search method (SA-DSM) and the hybrid cross-entropy method-sequential quadratic programming (CEM-SQP). In general, the stochastic search technique was responsible for "global exploration" and the deterministic technique was used to "local optimization" with the current solutions of the stochastic search technique as the starting points. In this study, an alternative approach is proposed for the solution of large-scale NED problem with valve-point effects using a hybrid particle swarm optimization and direct search method. A novel diversity based particle swarm optimization (DPSO) with a fewer iterations required is developed to increase the possibility of generating high quality initial solutions for the DSM. The enhanced direct search method (EDSM) incorporates the parallel nature of evaluation programming into the direct search procedure to enhance its search capacity using the answer from DPSO as starting points. A comparative analysis with other existing techniques demonstrates the superior performance of the proposed hybrid DPSO-EDSM algorithm in terms of both solution accuracy and convergence performances. Numerical experiments are also included to demonstrate that the proposed hybrid DPSO-EDSM approach can obtain a higher quality solution than the PSO or DSM for the large-scale NED application.

II. FORMULATION OF NON-CONVEX ECONOMIC DISPATCH PROBLEM

The main objective of the NED problems is to determine an optimal combination of power outputs of the online generating units so that the fuel cost of generation can be minimized, while simultaneously satisfying all unit and system equality and inequality constraints. Fig. 1 shows the configuration that will be studied in this paper. This system consists of N thermal generating units connected to a single bus-bar serving a received electrical load P_D. The objective function can be formulated as follows:

$$Minimize\ \ F_T = \sum_{i=1}^{N} F_i(P_i) \tag{1}$$

where F_T is the total fuel cost. N is the number of units in the system. $F_i(P_i)$ is the fuel cost function of unit i, and P_i is the power output of unit i. Generally, fuel cost of generation unit will be in second-order polynomial function (Wood and Wollenberg, 1996).

$$F_i(P_i) = a_i + b_i P_i + c_i P_i^2 \tag{2}$$

where a_i, b_i and c_i are the cost coefficients of unit i.

However, the thermal units with multi-valve steam turbines exhibit a greater variation in the fuel cost functions. Reference (Walters and Sheble, 1993) has shown the input-output performance curve for a typical thermal unit with many valve points. The cost curve function of units with valve point effects is depicted in Fig. 2. The fuel cost functions should be replaced by the following to take into account the valve-point effects.

$$F_i(P_i) = a_i + b_i P_i + c_i P_i^2 + \left| e_i \sin(f_i (P_i^{\min} - P_i)) \right| \tag{3}$$

where e_i and f_i are the cost coefficients of generator i reflecting valve-point effects.
Subject to following constraints:

- Power balance constraint

$$\sum_{i=1}^{N} P_i = P_D + P_{Loss} \tag{4}$$

- Unit capacity constraints

$$P_i^{\min} \le P_i \le P_i^{\max} \tag{5}$$

where P_D is the total load demand; P_{Loss} is the transmission loss; $P_i^{\min}$ and $P_i^{\max}$ are minimum and maximum generation limits of unit i, respectively. The transmission losses are traditionally represented by

$$P_{Loss} = \sum_{i=1}^{N}\sum_{j=1}^{N} P_i B_{ij} P_j + \sum_{i=1}^{N} B_{0i} P_i + B_{00} \tag{6}$$

where B_{ij} is the coefficient of transmission losses.

III. DEVELOPMENT OF PROPOSED HYBRID DPSO-EDSM ALGORITHM

1. Traditional PSO algorithm and its improvement

PSO was original presented by Kennedy and Eberhart (Kennedy and Eberhart, 1995; Shi and Eberhart, 1998). It was inspired by observation of the behaviors in bird flocks and fish schools. While searching for food, the birds are either scattered or go together before they locate the place where they can find food. While the birds are searching for food from one place to another, there is always a bird that can smell the food very well, that is, this bird is perceptible of the place where the food can be found, having the better food resource information. Because they are transmitting the information, the birds will eventually flock to the place where food can be found. Therefore, the most optimist solution can be worked out in PSO algorithm by the cooperation of each individual.

In the traditional PSO, the movement of a particle (bird) is governed by three behaviors which are inertia, cognitive and social. The inertia behavior simulates the particle to swarm in the previous direction (its present velocity). The cognitive behavior helps the particle to remember its previously visited best position (its previous experience; Pbest). The social behavior models the memory of the particle about the best position among the particles (the experience of its neighbors; Gbest). The position of each particle is updated using its velocity vector as shown in Fig. 3. The modified velocity and position of each particle can be calculated using the current velocity and the distance from $Pbest_q$ to $Gbest$ as shown in the following formulas:

$$V_q^{k+1} = \omega \times V_q^k + c1 \times rand \times (Pbest_q^k - X_q^k) + c2 \times rand \times (Gbest^k - X_q^k) \tag{7}$$

$$X_q^{k+1} = X_q^k + V_q^{k+1}\ , \quad q=1,2,\ldots,NP \tag{8}$$

$$\omega = \omega_{\max} - (\omega_{\max} - \omega_{\min}) \times \frac{iter}{iter_{\max}} \tag{9}$$

where NP is the population size; V_q^k is the velocity of particle q in iteration k; X_q^k is the position of particle q in iteration k; $Pbest_q^k$ is the best value of fitness function that has been achieved by particle q before iteration k; $Gbest^k$ is the best value of fitness function that has been achieved so far by any particle; $c1$ and $c2$ represent the weighting of the stochastic acceleration terms that pull each particle toward $Pbest_q$ and $Gbest$ positions; rand means a random variable between 0.0 to 1.0; ω is the inertia weight factor; $\omega_{\max}$ and $\omega_{\min}$ are the initial and final weight respectively; $iter_{\max}$ is the maximum iteration count, and $iter$ is the current number of iterations.

Similar to other evolutionary algorithms, the PSO has a number of parameters that must be selected. The acceleration constants c1 and c2 should be determined in advance that control the maximum step size. The inertia weight ω controls the impact of the previous velocity of the particle on its current one. The appropriate selection of these parameters justifies the preliminary efforts required for their experimental determination. It is obvious that the Gbest is also an important factor to provide the information guiding to the global solution. However, it is not reasonable for social behavior to only employ the Gbest which is not normally the global optimal solution, containing parts of non-optimal information. The influence of social behavior to the next movement of the bird (particle) often is affected not only by the location of the bird (particle) which is in the best position of all, but also by the location of the bird (particle) which it randomly looked at when bird flocks start looking for food. Therefore, the traditional PSO has premature convergence problem and easy to be trapped in local optima if a promising area where the global optimum is residing is not identified at the end of the optimization process.

To increase the possibility of exploring the search space where the global optimal solution exists, we follow a slightly different approach about the social behavior to further provide a selection of the global best guide of the particle swarm. The social behavior consists of two phases, the best particle position ever obtained (Gbest) and the random another particle best position ($Pbest_{ap}$), namely, another behavior. Fig. 4 presents the seeking algorithm of the proposed novel strategy. After increasing another behavior to the social behavior, the $Pbest_{ap}$ provides parts of information guiding to the global solution and gives additional exploration capacity to swarm. However, the information guiding to the global solution from the $Pbest_{ap}$ may contain in the best particle position ever obtained, Gbest. The $Pbest_{ap}$ cannot normally present a positive guidance. For maintaining population diversity, an intelligent judgment mechanism for the evaluation of the $Pbest_{ap}$ behavior is developed to give a good direction to identify the near global region. The new velocity of each particle can be calculated as shown in the following formulas.

$$\begin{aligned} V_q^{k+1} &= c0 \times V_q^{k} + c1 \times rand \times (Pbest_q^{k} - X_q^{k}) + c2 \\ &\times rand \times (Gbest^k - X_q^{k}) - c3_q^{k} \times rand \times (Pbest_{ap}^{k} - X_q^{k}),\ if\ (x_{Gbest}^k - x_q^k) \times (x_{ap}^k - x_q^k) \ge 0 \end{aligned} \tag{10}$$

$$\begin{aligned} V_q^{k+1} &= c0 \times V_q^{k} + c1 \times rand \times (Pbest_q^{k} - X_q^{k}) + c2 \\ &\times rand \times (Gbest^k - X_q^{k}) + c3_q^{k} \times rand \times (Pbest_{ap}^{k} - X_q^{k}),\ if\ (x_{Gbest}^k - x_q^k) \times (x_{ap}^k - x_q^k) < 0 \end{aligned} \tag{11}$$

$$c3_q = c3_{\max} - (c3_{\max} - c3_{\min}) \times \frac{\mathrm{iter}}{iter_{\max}}\ , \quad q = 1,2,\ldots,NP \quad ; \quad ap \ne q \tag{12}$$

where c0 is the inertia weight factor; $Pbest_{ap} = \left\{x_{ap1}^{Pbest}, x_{ap2}^{Pbest}, \ldots, x_{apN}^{Pbest}\right\}$ is the best position of a random another particle, called particle ap; $c3_q = \left\{c3_{q1}, c3_{q2}, \ldots, c3_{qN}\right\}$ is the weight factor of another behavior; $c3_{\max}$ and $c3_{\min}$ are the initial and final weight respectively.

The weight factor $c3_q$ plays the role of maintaining a good spread of non-dominated solutions. From (10), if the $(x^k{}_{ap} - x^k{}_q)$ and $(x^k{}_{Gbest} - x^k{}_q)$ move at the same direction, the information guiding to the global solution from Pbest_{ap} and Gbest is too similar. Compared with the Gbest, $x^k{}_{ap}$ is a bad position and the influence of particle ap to the movement of particle q is negative guidance. Otherwise, the information guiding to the global solution from Pbest_{ap} and Gbest is much more different if the $(x^k{}_{Gbest} - x^k{}_q)$ and $(x^k{}_{ap} - x^k{}_q)$ do not move at the same direction. As shown in Eq. (11), the influence of particle ap to the movement of particle q is positive guidance. The main attractive feature of intelligent judgment mechanism for the evaluation of the Pbest_{ap} behavior described above is to maintain the population diversity, which increases the possibility of occurrence of escaping from local optimal solutions.

2. Standard DSM algorithm and its improvement

DSM, first introduced by Chen and Chen, has been successfully applied to economic dispatch problem considering transmission capacity constraints (Chen and Chen, 2001). A salient feature of the DSM is to begin with an initial feasible solution and search for the optimal solution along a trajectory that maintains a feasible solution at all time. The advantage of direct search procedure is to handle several inequality constraints without introducing any multipliers. Furthermore, it can solve problems with derivatives unavailable or the fuel cost functions being much more complicated. Results show that the algorithm is an efficient approach for determining the optimal generation schedules. However, there are many problems in the solution process by the standard DSM for solving the NED problem. Like many local search techniques, the standard DSM is more sensitive to the initial starting points and has a number of parameters that must be selected carefully. Like other stochastic searching techniques, the main problem of the DSM is that it gets easily trapped in a local optimal solution, especially while handling large-scale NED problems with more local optima and heavier constraints. Therefore, the standard DSM still need further research and development to improve its performance and to obtain the robustness.

A good initial solution could enhance the possibility to obtain a better solution. However, it is easily trapped in local minima since, with a single initial solution, it is hardly to explore the search space where the global optimal solution exists. To enhance the solution quality of DSM, the stochastic technique is applied for the standard DSM to generate a population of NP initial candidate solutions at random and finds solution in parallel using direct search procedure. To further weaken the dependence of finding the global optimal solution on the initial starting solutions, the selection of calculation step S in the direct search procedure is also vital to the success of DSM to find the global optimal solution. In the previous work (Chen, 2006), the EDSM with large initial calculation step S_1 and small reduced factor K is usually commended to enhance its search capacity. The attractive feature of the EDSM is to reduce the step size gradually by using the multi-level convergence strategy to increase the possibility of occurrence of escaping from local optimal solution. The numerical results show the EDSM can identify a near global region and perform a local search rapidly. The efficient approach makes it an attractive method, and this methodology is very suitable for assessing costs of NED problem.

3. Conventional hybrid DPSO-EDSM algorithm and its improvement

Usually, the stochastic search technique can identify a near global region but slows in a finely tuning local search. In contrast, the local searching technique can climb hills rapidly but is easily trapped in local minima. Development of hybrid DPSO-EDSM algorithm is necessary to improve the solution quality and performance for the large-scale NED problem. In general, the DPSO algorithm was responsible for “global exploration” and the EDSM algorithm was used to “local optimization” with the current solutions of the DPSO as the starting points. The outline of the conventional DPSO-EDSM algorithm is shown in the flowchart in Fig. 5. However, the conventional hybrid DPSO-EDSM has premature convergence problem if a promising area where the global optimum is residing is not identified at the end of the optimization process. Like DPSO algorithm, the EDSM may also get easily trapped in a local optimal solution because the initial starting points obtained by the DPSO are too similar. Enhancement of solution quality becomes major concern to solve the large-scale NED problem. Besides, it is obvious that the major portion of computing time is spent in performing the DPSO technique to explore the search space where the global optimal solution exists. Improvement of solution performance becomes another concern to solve the large-scale NED problem. Therefore, the conventional DPSO-EDSM algorithm still needs further research and development to improve its performance and to obtain the robustness.

To enhance its search capacity, a modified DPSO-EDSM algorithm is developed to improve the solution quality and performance for the large-scale NED problem. In this study, the DPSO algorithm is only used to generate high quality initial starting points and the EDSM algorithm is responsible for both global exploration and local optimization. It should be noted that the advantage of the EDSM algorithm is to begin with a coarse convergence step to enhance the global exploration ability and end with a refined convergence step to enable quick convergence. To enhance the solution quality of EDSM, a larger population size NP is desired to increase the possibility of finding the global optimal solution for the large-scale NED problem. However, the main problem of the EDSM is more sensitive to the initial random starting points and it is hardly to explore the search space where the global optimal solution exists. To identify a near global region, the DPSO with a fewer iterations required is used to increase the possibility of generating high quality initial solutions for the EDSM. Like meta-heuristic approaches, the parallel searching mechanism incorporated in DSM algorithm is also used to enhance its search capacity, leads to a higher probability of obtaining the global optimal solution. The outline of the proposed modified DPSO*-EDSM algorithm is shown in the flowchart in Fig. 6.

IV. IMPLEMENTATION OF MODIFIED DPSO-EDSM ALGORITHM FOR NED PROBLEMS

1. Improved DSM with a parallel searching mechanism

Like meta-heuristic approaches, the parallel searching mechanism incorporated in standard DSM algorithm is used to enhance its search capacity, leads to a higher probability of obtaining the global optimal solution. Let rand be uniform random value in the range [0,1]. The initial power outputs of $N-1$ generating units without violating (5) are generated randomly by

$$P_i = P_i^{\min} + rand \times (P_i^{\max} - P_i^{\min}) \tag{13}$$

To satisfy the power balance equation, a dependent generating unit is arbitrarily selected among the committed N units and the output of the dependent generating unit P_d is determined by

$$P_d = P_D + P_{Loss} - \sum_{\substack{i=1 \\ i \neq d}}^{N} P_i \tag{14}$$

Whereas P_d can be calculated directly from the quadratic equation as shown in below.

$$AP_d^2 + (B-1)P_d + \left[C + P_D - \sum_{\substack{i=1 \\ i \neq d}}^{N} P_i \right] = 0 \tag{15}$$

where,

$$A = B_{dd}$$

$$B = \sum_{\substack{j=1 \\ j \neq d}}^{N} B_{dj} P_j + \sum_{\substack{i=1 \\ i \neq d}}^{N} P_i B_{id} + B_{d0}$$

$$C = \sum_{\substack{i=1 \\ i \neq d}}^{N} \sum_{\substack{j=1 \\ j \neq d}}^{N} P_i B_{ij} P_j + \sum_{\substack{i=1 \\ i \neq d}}^{N} B_{i0} P_i + B_{00}$$

If P_d with violating (5), a repairing strategy is applied to pick one unit at random to increase (or decrease) its output by the random or predefined step (e.g., 10 MW), one by one, until it can satisfy all the constraints.

2. Improved DSM with a high quality initial solutions mechanism

To enhance the solution quality of DSM, a larger population size of NP initial candidate solutions is desired to increase the possibility of finding the global optimal solution for the large-scale NED problem. To further explore the search space where the global optimal solution exists, the DPSO with a fewer iterations required, described in Section 3.2, is applied to generate high quality initial solutions for the EDSM. The process of the DPSO* can be summarized as follows:

Step 1: Establish the DPSO* parameters.
Set up the set of parameters of DPSO*, such as number of particles NP, weighting factors c0, c1, c2, $c3_{max}$, $c3_{min}$, and predefined number of iterations (i.e. iter0=10~300).

Step 2: Create an initial population of particles randomly.
The stochastic technique, described in Section 4.1, is applied to generate an initial population of particles randomly.

Step 3: Evaluate the value of the fitness function for each particle.
Calculate the value of fitness function for each particle. The fitness function is an index to evaluate the fitness of particles. Equation (1) shows the fitness function of the NED problem.

Step 4: Record and update the Pbest and Gbest.
The two best values are recorded in the searching process. Each particle keeps track of its coordinate in the solution space that is associated with the best solution it has reached so far. This value is recorded as Pbest. Another best value to be recorded is Gbest, which is the overall best value obtained so far by any particle.

Step 5: Update the velocity and position of the particles.
Eqs. (8), (10)-(12) are applied to update the velocity and position of particles. The velocity of a particle represents a movement of the generation of the generators. The position of a particle is the generation of the generators. It represents a movement of a particle. The new positions of the particles are forced to satisfy the unit's generation limit constraint given by (5) and other constraints if they exist.

Step 6: Check the end condition.
If the predefined number of iterations (iter0) is reached, invoke the EDSM algorithm with the current solutions of the DPSO* as the starting points to further explore the final optimal solution, otherwise, repeat steps 3-5 until the end conditions are satisfied.

3. Direct search procedure for candidates

Exploration on initialization begins with finding the best direction for improvement. One-at-a-time search is an effective strategy of direct search procedure for handling coupling constraints effectively without introducing any multipliers. At each step of the searching process, only a particular pair of units (assume unit x and unit y, y≠x) is selected to achieve the most reduction in the total fuel cost F_T. Once all units are examined and no improvement in the total operating cost is found, the search process is terminated. The computation steps of the enhanced direct search procedure are shown as follows:

Step 1: Units, without violating the maximum or minimum generation limits, are to increase or decrease their outputs by the predefined step S for calculating their incremental costs (IC) and decrement costs (DC). This is shown as follows:

$$IC_i = \frac{F_i(P_i + S) - F_i(P_i)}{S} \quad \text{for} \quad i=1, 2, \ldots, N \tag{16}$$

$$DC_i = \frac{F_i(P_i) - F_i(P_i - S)}{S} \quad \text{for} \quad i=1, 2, \ldots, N \tag{17}$$

subject to

$$P_i + S \le P_i^{\max} \quad \text{and} \quad P_i - S \ge P_i^{\min} \tag{18}$$

Step 2: All units are examined to check if there is any improvement. If no more improvement can be achieved, then stop; otherwise, go to step 3.
Step 3: An independent unit with minimum incremental cost ICx (assume unit x) is chosen to increase its output by the predefined step S, and then, only a dependent unit DCy (assume unit y, y≠x) while gaining the most reduction in the total operating cost F_T, should be selected to reduce its output to satisfy the power balance equation.
Step4: The outputs of this particular pair of units will be adjusted again by the predetermined step S if they do not violate the generation limits, and only the incremental cost of unit x and the decrement cost of unit y need to be recalculated.
Step 6: Go to step 2.

4. Overall hybrid DPSO*-EDSM solution procedure

The overall procedure of the proposed DPSO*-EDSM algorithm can be stated as follows:

Step 1: Read system data.
Step 2: Set the proper values of initial step size S_1 and reduced factor K.
Step 3: Initialize a population of candidate solutions at random.
Step 4: Re-initialize this population of candidate solutions by using DPSO* with a fewer iterations required (iter0).
Step 5: $S = S_1$
Step 6: Perform direct search procedure for candidates.
Step 7: Is S greater than predefined resolution ε ?
Yes, S= S/K, go to step 6; otherwise, go to step 8.
Step 8: Print results.

V. NUMERICAL EXPERIMENTS

To verify the feasibility and effectiveness of the proposed hybrid algorithm, numerical studies have been performed for the several test systems, where valve-point effects are considered. All the computation is performed on a PC Pentium (R) Dual CPU 2.00 GHz computer with 1.0GRAM size, and several computer programs were developed in FORTRAN:

PSOIW: Particle swarm optimization with inertia weight
DPSO: Diversity based particle swarm optimization
EDSM: Enhanced direct search method
DPSO-EDSM: DPSO with local optimization using the EDSM
DPSO*-EDSM: EDSM with high quality initial solutions obtained by the DPSO*

After testing and evaluating different parameter combinations, parameters of the PSO-IW, DPSO, EDSM, DPSO-EDSM and DPSO*-EDSM algorithms used in the three test systems are listed in Table 1 for clarity. The studied cases are stated in detail as follows:

1. Example 1: Test for a 13-unit system

In the first example, a system with thirteen generating units considering the valve-point effects is studied. The system unit data is given in Ref. (Victoire and Jeyakumar, 2004) and the total load demand is 2520MW. Network losses are neglected in the tests for comparison. Table 2 depicts the numerical results of the various methods. The best result obtained by the proposed DPSO*-EDSM is compared with those of the HSS in (Bhagwan Das and Patvardhan,1999), the ESA in (Lu et al., 2008), the EP-SQP in (Victoire and Jeyakumar, 2004) and the PSO-SQP in (Victoire and Jeyakumar, 2004). This table reveals that the proposed approach can obtain a higher quality solution than many existing techniques. It shows that the best cost of the PSO-SQP is \$24261.05 and that of proposed DPSO*-EDSM algorithm is \$24169.92. Details of the best solutions obtained by the proposed DPSO*-EDSM algorithm is shown in the sixth column of Table 2. To further examine the merits of the DPSO*-EDSM algorithm, Table 3 shows the dispatch results of the PSOIW, DPSO, EDSM, DPSO-EDSM and DPSO*-EDSM algorithms for 30 trial runs. The simulation results reveal that the DPSO*-EDSM has provided better solution than the other approaches. Also, the efficiency of the proposed hybrid algorithm has been demonstrated in the test case. To investigate effects of initial trail solutions on the final results, different initial solutions obtained by DPSO* were given to the EDSM approach for comparison. Fig. 7 shows the variation of the average cost of 30 runs versus a series of different iter0 ranging from 0 to 60 in steps of 5 iterations. Although the average cost is oscillated as iter0 increases, the quality of the solution is improved with various iter0. Fig. 8 shows the solution obtained from EDSM, DPSO-EDSM and DPSO*-EDSM depends on the population size. This figure reveals that the results obtained by the proposed DPSO*-EDSM is very close to that of DPSO-EDSM and finds a better solution than EDSM in the studied case. The results show that the proposed DPSO*-EDSM provides an accurate algorithm to tackle efficiently the difficult NED problem.

2. Example 2: Test for a 40-unit system

In the second example, a system with forty generating units is studied to test the solution quality and performance of the proposed hybrid algorithm. The system unit data is shown in Ref. (Sinha et al., 2003) and the total load demand is 10500 MW. The corresponding costs of the obtained best solution from DPSO*-EDSM are compared with those of the previous researches in Table 4, such as MFEP (Sinha et al., 2003), IFEP (Sinha et al., 2003), PSO-SQP (Victoire and Jeyakumar, 2004), GA-PS-SQP (Alsumait et al., 2010), HCPSO (Cai et al., 2012), HCPSO-SQP (Cai et al., 2012), SOMA (Coelho and Mariani, 2010) and CE-SQP (Subathra et al., 2015). From these results, the proposed hybrid algorithm can find a better solution ($121412.6) than many existing techniques, and has clearly shown the superiority to the previous researches in terms of minimum cost as well as average cost. To illustrate the good convergence property of the proposed algorithm, Table 5 gives a comparison of operation costs during each convergence level. Details of the best solutions obtained by the proposed DPSO*-EDSM algorithm is shown in the Table 6. To demonstrate the need for integrating the EDSM with the DPSO*, Table 7 shows the best cost, average cost, and worst cost achieved for 30 trial runs using various heuristic algorithms. From the results, the basic PSOIW has premature convergence problem and easy to be trapped in local optima (average cost: $121885.6). Using an intelligent judgment mechanism, the proposed DPSO can find a better solution (average cost: $121485.8) than the basic PSOIW technique. However, the DPSO makes no guarantee that the solutions are optimal or even close to the optimal solution. Similar to conventional PSO algorithm in optimization, the main problem of the DPSO-EDSM is that it also gets trapped in a local optimal solution (average cost: $121431.6) since a promising area where the global optimal is residing is not identified at the end of the optimization process. It is seen that the satisfactory solution (average cost: $121418.0) achieved by EDSM with better performance. However, only the near global optimal solution can be obtained by the EDSM approach. As shown in the sixth lows of Table 7, the final results (average cost: $121412.8) of DPSO*-EDSM with high quality initial starting points are better than that of EDSM. This test case study converges within 1 sec for each run when the value of iter0 is chosen to be 20.

To investigate effects of initial trail solutions on the final results, different initial solutions obtained by DPSO* and PSOIW* were given to the EDSM approach for comparison. Fig. 9 shows the variation of the average cost of 30 runs versus a series of different iter0 ranging from 0 to 300 iterations. The results show that the DPSO* performs much better than PSOIW* as an optimizer for initialization and the superiority of the DPSO*-EDSM algorithm over PSOIW*-EDSM can also be noticed. Although multiple local minimum solutions exist in this studied case, the proposed DPSO*-EDSM can still find a better solution than EDSM when the value of iter0 is less than 150. It can also be seen that the average fuel cost of 30 runs is lowest one in this figure when the value of iter0 is chosen to be 20. But in certain cases, the average cost may be oscillated as iter0 increases. To improve the final solution, an iterative process with different iter0 ranging from 0 to 300 in steps of 10 iterations can be placed outside the DPSO*-EDSM loop. In this study, the proposed hybrid algorithm is terminated if the best cost is unchanged within three consecutive iterations. The quality of the solution is found with various iter0 as illustrated in Table 8 when the value of NP is chosen to be 50 in a typical run. Note that the best solution is always saved among the obtained solutions during iterative process. Fig. 10 shows the solution obtained from iterative DPSO*-EDSM depends on the population size. Increasing of population size will provide a better solution but takes longer computing time. Note that the DPSO*-EDSM method still finds a satisfactory solution (average cost: $121413.6) even with a very small population size (NP=40). This test case study converges within 1.67 sec for each run when the value of NP is chosen to be 100.

3. Example 3: Test for a 80-unit system

In the last example, the simulation includes test runs for the large-scale system to demonstrate the robustness and effectiveness of the proposed DPSO*-EDSM algorithm. The 80-unit system is created simply by expanding example 2. The degree of complexity of the NED problem is related to the system-size. The larger system-size increases the non-linearity as well as the number of equality and inequality constraints in the NED problem. There are many local optimal solutions for the dispatch problem and the problem is well suitable for testing and validating the developed hybrid algorithm. The results obtained by the proposed DPSO*-EDSM are compared with those obtained by using previously published methods, such as CSO (Selvakumar and Thanushkodi, 2009), PSO (Selvakumar and Thanushkodi, 2009), CSE (Selvakumar and Thanushkodi, 2009) and CE-SQP (Subathra et al., 2015). Table 9 depicts the numerical results of various methods. This table reveals that the proposed hybrid algorithm outperforms other existing methods. It shows that the best cost of the CE-SQP is $242883.04 and that of proposed DPSO*-EDSM algorithm is $242794.7, which is the minimum cost found so far. Details of the best solutions obtained by the proposed DPSO*-EDSM algorithm is shown in the Table 10. Fig. 11 shows the variation of the average cost of 30 runs versus a series of different iter0 ranging from 0 to 300 iterations. It can also be seen that the average fuel cost of 30 runs is lowest one in this figure when the value of iter0 is chosen to be 100. To further examine the merits of the DPSO*-EDSM algorithm, Table 11 depicts the numerical results of various methods. From the results, the superiority of the DPSO*-EDSM algorithm over basic PSOIW, DPSO, EDSM and DPSO-EDSM can be noticed. From these results, although multiple local minimum solutions exist in this studied case, the proposed DPSO*-EDSM can still find a better solution than EDSM, by

0.02 percent equivalent to 49.8 (refer to Table 11). Furthermore, the solution reached by the proposed DPSO*-EDSM is also better than DPSO-EDSM, by 0.006 percent equivalent to 14.8. Table 12 shows the solution of DPSO*-EDSM after thirty runs under different particle numbers. From this result, the average cost of thirty runs decreased when the particle number increased. It is also observed that the total operation cost is not sensitive to the particle number. In fact, several different cases were studied and the results show that the final results of DPSO*-EDSM are better than those of PSOIW, DPSO, EDSM and DPSO-EDSM. The encouraging simulation results clearly show that the proposed DPSO*-EDSM is capable of obtaining higher quality solutions to tackle the difficult NED problems. The efficient approach also makes it an attractive method for the solution of the large-scale NED problem in these test cases. The suitableness of the algorithm presented in this paper to the solution of the optimal NED problem is, thus, confirmed.

VI. CONCLUSIONS

This paper presents a modified hybrid algorithm based on a combination of DPSO and EDSM to solve the NED problems with valve-point effects. Adding the $Pbest_{ap}$ item with a diversity based judgment mechanism, the proposed DPSO algorithm can give a good direction to generate high quality initial solutions for the EDSM. The EDSM incorporates the parallel searching mechanism of evaluation programming into the direct search procedure to enhance its search capacity about global exploration and local optimization. The global searching capability has been improved significantly by the proposed heuristic mechanism in the three test systems. It is observed that obtaining the global optimal solution for the NED problem is possible by using the proposed hybrid DPSO*-EDSM algorithm. Numerical experiments also demonstrate that the proposed algorithm is more practical and valid than many existing techniques for the solution of the large-scale NED problem.

ACKNOWLEDGEMENTS

Financial supports under the Grant NO. NSC102-2221-E-019-025 from the National Science Council, Taiwan, ROC are acknowledged.

REFERENCES

Alsumait, J. S., J. K. Sykulski and A. K. Al-Othman (2010). A hybrid GA–PS–SQP method to solve power system valve-point economic dispatch problems. Applied Energy 87, 1773-1781.

Bhagwan Das, D. and C. Patvardhan (1999). Solution of economic load dispatch using real coded hybrid stochastic search. Electric Power and Energy System 21, 165-170.

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

Chen, C.L. (2006). Non-convex economic dispatch: A direct search approach. Energy Convers and Manage. 48, 219-225.

Chen, C. L. and N. Chen (2001). Direct search method for solving economic dispatch problem considering transmission capacity constraints. IEEE Trans. on Power Systems 16, 764-769.

Coelho, L.S. and V.C. Mariani (2010). An efficient cultural self-organizing migrating strategy for economic dispatch optimization with valve-point effect. Energy Convers and Manage. 51, 2580-2587.

Gaing, Z. L. (2003). Particle swarm optimization to solving the economic dispatch considering the generator constraints. IEEE Trans. on Power Systems. 18, 1187-1195.

Kennedy, J. and R. Eberhart (1995). Particle swarm optimization. Proc. IEEE International Conference Neural Networks 4, 1942-1948.

Lee, J.C., W.M. Lin, G.C. Liao and T.P. Tsao (2011). Quantum genetic algorithm for dynamic economic dispatch with valve-point effects and including wind power system. International Journal of Electrical Power & Energy Systems 33, 189-197.

Liang, Z. X. and J. D. Glover (1992). A zoom feature for a dynamic programming solution to economic dispatch including transmission losses. IEEE Trans. on Power Systems 7, 544-550.

Lu, C. L., C.L. Chen, D.S. Hwang and Y.T. Cheng (2008). Effects of wind energy supplied by independent power producers on the generation dispatch of electric power utilities. International Journal of Electrical Power & Energy Systems 30, 553-561.

Lu, H., P. Sriyanyong, Y. H. Song and T. Dillon (2010). Experimental study of a new hybrid PSO with mutation for economic dispatch with non-smooth cost function. International Journal of Electrical Power & Energy Systems 32, pp. 921-935.

Selvakumar, A.I., K. Thanushkodi (2009). Optimization using civilized swarm: Solution to economic dispatch with multiple minima. Elect. Power Syst. Res. 79, 8–16.

Sinha, N., R. Chakrabarti and P. K. Chattopadhyay (2003). Evolutionary programming for economic load dispatch. IEEE Trans. on Evolutionary Computation 7, 83-94.

Shi, Y. and R. Eberhart (1998). A modified particle swarm optimizer. Proc. IEEE International Conference Evol. Comput., Anchorage, Alaska, 69-73.

Subathra, M.S.P., S.E. Selvan, T.A.A. Victoire, A.H. Christinal and U. Amato (2015). A hybrid with cross-entropy method and sequential quadratic programming to solve economic load dispatch problem. IEEE Trans. on System Journal 9, 1031-1044.

Victoire, T. A. A. and A. E. Jeyakumar (2004). Hybrid PSO-SQP for economic dispatch with valve-point effect. Electric Power Systems Research 71, 51-59.

Walters, D. C. and G. B. Sheble (1993). Genetic algorithm solution of economic dispatch with valve point loading. IEEE Trans. on Power Systems 8, 1325-1331.

Wong, K. P. and C. C. Fung (1993). Simulated annealing based economic dispatch algorithm. IEE Proc. Gen., Transm., Distrib. 140, 509-515.

Wong, K.P. and Y.W. Wong (1994). Genetic and genetic/simulated-annealing approach to economic dispatch. IEE Proc. 141, 507-513.

Wood, J. and B. F. Wollenberg (1996). Power Generation Operation and Control, 2^{nd} ed, New York: Wiley, 29-57.

Yang, H. T., P. C. Yang and C. L. Huang (1996). Evolutionary programming based economic dispatch for units with non-smooth fuel cost functions. IEEE Trans. on Power Systems 11, 112-118.

Table 1. Best parameter setting of various methods in the three test systems.

DOI: 10.6119/JMST-015-1215-1

Parameter	Example 1: 13-unit system	Example 2: 40-unit system	Example 3: 80-unit system
PSOIW	NP=100; c1=2.0; c2=2.0; $\omega_{max}=0.9$; $\omega_{min}=0.4$; $iter_{max}$=1000	NP=300; c1=2.0; c2=2.0; $\omega_{max}=0.9$; $\omega_{min}=0.4$; $iter_{max}$=2000	NP=600; c1=2.0; c2=2.0; $\omega_{max}=0.9$; $\omega_{min}=0.4$; $iter_{max}$=2000
DPSO	NP=100; c0=0.3; c1=2.5; c2=0.8; $c3_{max}$=0.4; $c3_{min}$=0.01; $iter_{max}$=1000	NP=300; c0=0.3; c1=2.5; c2=0.8; $c3_{max}$=0.4; $c3_{min}$=0.01; $iter_{max}$=2000	NP=600; c0=0.3; c1=2.5; c2=0.8; $c3_{max}$=0.4; $c3_{min}$=0.01; $iter_{max}$=2000
EDSM	NP=100; S_1=200; K=1.2; ε =0.001	NP=300; S_1=200; K=1.2; ε =0.001	NP=600; S_1=200; K=1.2; ε =0.001
DPSO-EDSM	NP=100; c0=0.3; c1=2.5; c2=0.8; $c3_{max}$=0.4; $c3_{min}$=0.01; $iter_{max}$=1000; S_1=200; K=1.2; ε =0.001	NP=300; c0=0.3; c1=2.5; c2=0.8; $c3_{max}$=0.4; $c3_{min}$=0.01; $iter_{max}$=2000; S_1=200; K=1.2; ε =0.001	NP=600; c0=0.3; c1=2.5; c2=0.8; $c3_{max}$=0.4; $c3_{min}$=0.01; $iter_{max}$=2000 S_1=200; K=1.2; ε =0.001
DPSO*-EDSM	NP=100; c0=0.3; c1=2.5; c2=0.8; iter0=10; $c3_{max}$=0.4; $c3_{min}$=0.01; S_1=200; K=1.2; ε =0.001	NP=300; c0=0.3; c1=2.5; c2=0.8; iter0=20; $c3_{max}$=0.4; $c3_{min}$=0.01; S_1=200; K=1.2; ε =0.001	NP=600; c0=0.3; c1=2.5; c2=0.8; iter0=100; $c3_{max}$=0.4; $c3_{min}$=0.01; S_1=200; K=1.2; ε =0.001

Table 2. Comparison of dispatch results of each method for the system Example 1.

Unit	HSS	ESA	EP-SQP	PSO-SQP	DPSO*-EDSM
1	628.23	628.3068	628.3136	628.3205	628.3185
2	299.22	298.8529	299.1715	299.0524	299.1990
3	299.17	298.7195	299.0474	298.9681	299.1990
4	159.12	159.7211	159.6399	159.4680	159.7330
5	159.95	159.5390	159.6560	159.1429	159.7330
6	158.85	159.6340	158.4831	159.2724	159.7328
7	157.26	159.0156	159.6749	159.5371	159.7328
8	159.93	159.6087	159.7265	158.8522	159.7329
9	159.86	159.0345	159.6653	159.7845	159.7329
10	110.78	76.3879	114.0334	110.9618	77.3996
11	75.00	77.1473	75.0000	75.0000	77.3996
12	60.00	92.2443	60.0000	60.0000	92.3998
13	92.62	91.7883	87.5884	91.6401	87.6868
Cost ($)	24275.71	24174.17	24266.44	24261.05	24169.92

Table 3. Comparison of results after 30 trials for the system Example 1.

Methods	Best cost ($)	Average cost ($)	Worse cost ($)	Avg. Time (s)
PSOIW	24287.91	24451.40	24643.13	0.458
DPSO	24171.29	24195.21	24242.83	0.605
EDSM	24169.92	24175.83	24216.21	0.092
DPSO-EDSM	24169.92	24170.62	24174.09	0.687
DPSO*-EDSM	**24169.92**	**24170.49**	**24174.09**	**0.107**

Table 4. Comparison of dispatch results of each method for the system Example 2.

Methods	Best cost ($)	Average cost ($)	Worse cost ($)
MFEP	122647.57	123489.74	---
IFEP	122624.35	123382.00	125740.63
PSO-SQP	122094.67	122245.25	---
GA-PS-SQP	121458	122039	---
HCPSO	121865.23	122100.74	---
HCPSO-SQP	121458.54	122028.16	---
SOMA	121418.79	121449.88	---
CE-SQP	121412.88	121423.65	---
DPSO*-EDSM	**121412.6**	**121412.8**	**121414.7**

Table 5. Comparison of costs under various S in the 40-unit system.

Convergence	Cost ($)	Convergence	Cost ($)	Convergence	Cost ($)
Initialization	147941.1	S_{22}=4.347 MW	121839.4	S_{45}=0.065 MW	121422.0
Re-initialization	135214.2	S_{23}=3.622 MW	121786.8	S_{46}=0.054 MW	121420.2
S_1=200.000 MW	124180.8	S_{24}=3.018 MW	121724.4	S_{47}=0.045 MW	121418.6
S_2=166.666 MW	124180.8	S_{25}=2.515 MW	121647.6	S_{48}=0.037MW	121418.1
S_3=138.888 MW	124180.8	S_{26}=2.096 MW	121639.9	S_{49}=0.031 MW	121417.0
S_4=115.740 MW	124180.8	S_{27}=1.747 MW	121618.3	S_{50}=0.026 MW	121416.2
S_5=96.450 MW	124176.0	S_{28}=1.455 MW	121601.0	S_{51}=0.021 MW	121415.8
S_6=80.375 MW	123729.2	S_{29}=1.213 MW	121569.5	S_{52}=0.018 MW	121415.0
S_7=66.979 MW	123729.2	S_{30}=1.011 MW	121542.9	S_{53}=0.015 MW	121414.7
S_8=55.816 MW	123409.5	S_{31}=0.842 MW	121522.8	S_{54}=0.012 MW	121414.4
S_9=46.513 MW	123032.8	S_{32}=0.702 MW	121491.2	S_{55}=0.010 MW	121414.0
S_{10}=38.761 MW	123032.8	S_{33}=0.585 MW	121476.0	S_{56}=0.0088 MW	121413.8
S_{11}=32.301 MW	122801.5	S_{34}=0.487MW	121468.2	S_{57}=0.0073 MW	121413.6
S_{12}=26.917 MW	122801.5	S_{35}=0.406 MW	121461.5	S_{58}=0.0061 MW	121413.4
S_{13}=22.431 MW	122639.4	S_{36}=0.338 MW	121453.9	S_{59}=0.0051MW	121413.2
S_{14}=18.692 MW	122625.9	S_{37}=0.282 MW	121449.5	S_{60}=0.0042 MW	121413.1
S_{15}=15.577 MW	122435.4	S_{38}=0.235 MW	121444.7	S_{61}=0.0035 MW	121413.0
S_{16}=12.981 MW	122373.4	S_{39}=0.195 MW	121435.0	S_{62}=0.0029 MW	121413.0
S_{17}=10.817 MW	122323.1	S_{40}=0.163 MW	121433.1	S_{63}=0.0024 MW	121412.9
S_{18}=9.014 MW	122296.0	S_{41}=0.136 MW	121429.9	S_{64}=0.0020 MW	121412.8
S_{19}=7.512 MW	122106.6	S_{42}=0.113 MW	121427.9	S_{65}=0.0017 MW	121412.8
S_{20}=6.260 MW	121976.2	S_{43}=0.094 MW	121425.1	S_{66}=0.0014 MW	121412.7
S_{21}=5.216 MW	121924.6	S_{44}=0.078 MW	121423.6	S_{67}=0.0011 MW	121412.6

Table 6. Best dispatch results for the 40-unit system.

Unit No.	P_i (MW)	Unit No.	P_i (MW)	Unit No.	P_i (MW)	Unit No.	P_i (MW)
1	110.799600	11	94.000210	21	523.279900	31	189.999900
2	110.799600	12	94.000120	22	523.279800	32	189.999800
3	97.400350	13	214.759200	23	523.279100	33	189.999200
4	179.733600	14	394.279700	24	523.280000	34	164.799500
5	87.799680	15	394.278700	25	523.279000	35	199.999800
6	139.999200	16	394.279600	26	523.279100	36	194.396800
7	259.600200	17	489.278900	27	10.000210	37	109.999700
8	284.599300	18	489.278900	28	10.000630	38	110.000000
9	284.599300	19	511.279800	29	10.000220	39	109.999800
10	130.000600	20	511.278900	30	87.800590	40	511.278900

Table 7. Comparison of results after 30 trials for the system Example 2.

Methods	Best cost ($)	Average cost ($)	Worse cost ($)	Avg. Time (s)
PSOIW	121745.5	121885.6	122213.5	8.13
DPSO	121417.6	121485.8	121694.6	10.91
EDSM	121412.6	121418.0	121440.1	0.80
DPSO-EDSM	121412.6	121431.6	121502.9	11.67
DPSO*-EDSM	**121412.6**	**121412.8**	**121414.7**	**1.03**

Table 8. Comparison of results with various iter0 in the 40-unit system .

Convergence	Initialization	iter0=0	iter0=10	iter0=20	iter0=30	iter0=40	iter0=50
Best cost ($)	134956.1	121467.3	121461.8	121414.9	121412.6	121412.6	121412.6

Table 9. Comparison of dispatch results of each method for the system Example 3.

Methods	Best cost ($)	Average cost ($)	Worse cost ($)
CSO	243195.38	243546.63	---
PSO	244188.35	246375.87	---
SCA	250864.05	254579.79	---
CE-SQP	242883.04	242945.25	---
DPSO*-EDSM	**242794.7**	**242813.9**	**242864.9**

Table 10. Best dispatch results for the 80-unit system.

Unit No.	P_i (MW)	Unit No.	P_i (MW)	Unit No.	P_i (MW)	Unit No.	P_i (MW)
1	110.799820	21	523.279372	41	110.799830	61	523.279362
2	110.799825	22	523.279363	42	110.799830	62	523.279365
3	97.399915	23	523.279374	43	97.399915	63	523.279372
4	179.733102	24	523.279376	44	179.733100	64	523.279374
5	87.799903	25	523.279363	45	87.799905	65	523.279374
6	140.000000	26	523.279374	46	140.000000	66	523.279365
7	259.599659	27	10.000007	47	259.599659	67	10.000004
8	284.599647	28	10.000005	48	284.599647	68	10.000000
9	284.599647	29	10.000014	49	284.599647	69	10.000002
10	130.000000	30	87.799903	50	130.000000	70	87.799905
11	168.799817	31	189.999986	51	168.799822	71	190.000000
12	94.000002	32	189.999995	52	94.000008	72	189.999999
13	214.759788	33	189.999996	53	214.759787	73	190.000000
14	394.279369	34	164.799820	54	394.279372	74	164.799820
15	394.279370	35	199.356192	55	394.279360	75	199.999992
16	394.279369	36	164.799832	56	304.519569	76	164.799832
17	489.279372	37	109.999996	57	489.279375	77	109.999986
18	489.279373	38	109.999995	58	489.279362	78	110.000000
19	511.279365	39	109.999997	59	511.279361	79	109.999914
20	511.279370	40	511.279373	60	511.279365	80	511.279373

Table 11. Comparison of results after 30 trials for the system Example 3.

DOI: 10.6119/JMST-015-1215-1

Methods	Best cost ($)	Average cost ($)	Worse cost ($)	Avg. Time (s)
PSOIW	243923.1	244206.5	245044.5	32.03
DPSO	242865.6	243171.2	243865.6	44.10
EDSM	242844.5	242926.3	243014.7	4.67
DPSO-EDSM	242809.5	242903.0	243013.9	49.46
DPSO*-EDSM	**242794.7**	**242813.9**	**242864.9**	**7.60**

Table 12.Comparison of results under various NP in the Example 3 by using DPSO*-EDSM algorithm.

Particle numbers (NP)	Best cost ($)	Average cost ($)	Avg. Time (s)
100	242812.6	242836.5	1.31
200	242801.5	242836.2	2.67
300	242798.4	242827.8	4.00
400	242794.7	242819.6	5.27
500	242794.7	242816.5	6.47
600	242794.7	242813.9	7.60
700	242794.7	242813.7	9.10
800	242794.7	242813.5	10.30
900	242794.7	242813.4	11.53

DOI: 10.6119/JMST-015-1215-1

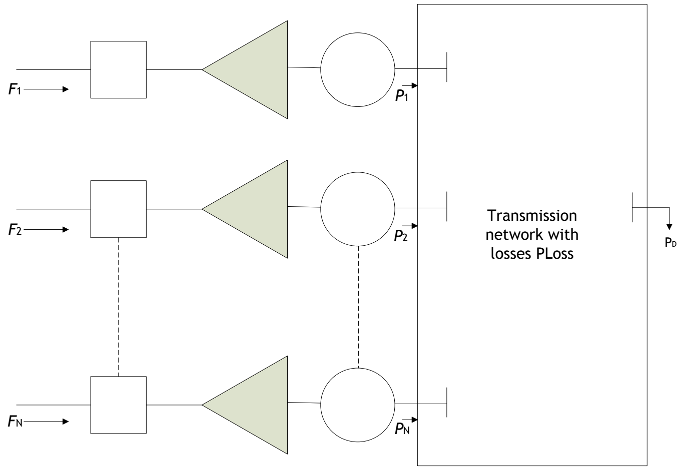

Fig. 1. N thermal units committed to serve a load of P_D

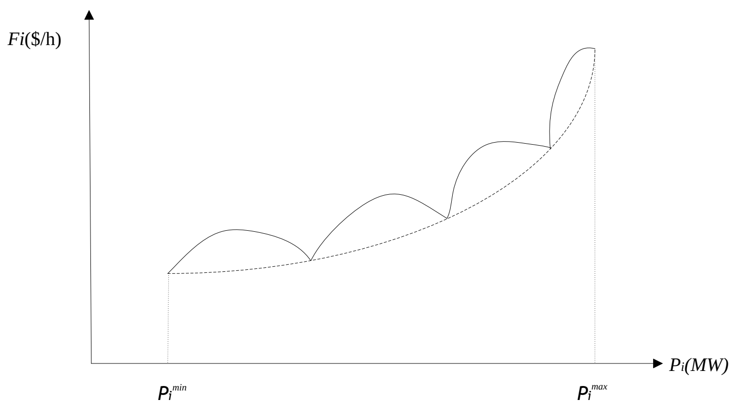

Fig. 2. Fuel cost curve of units with valve-point effects

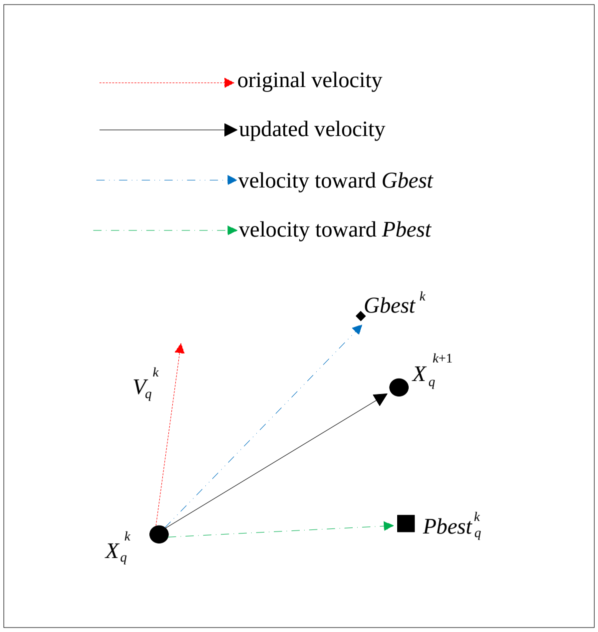

Fig. 3. Depiction of the velocity and position updates in the traditional PSO

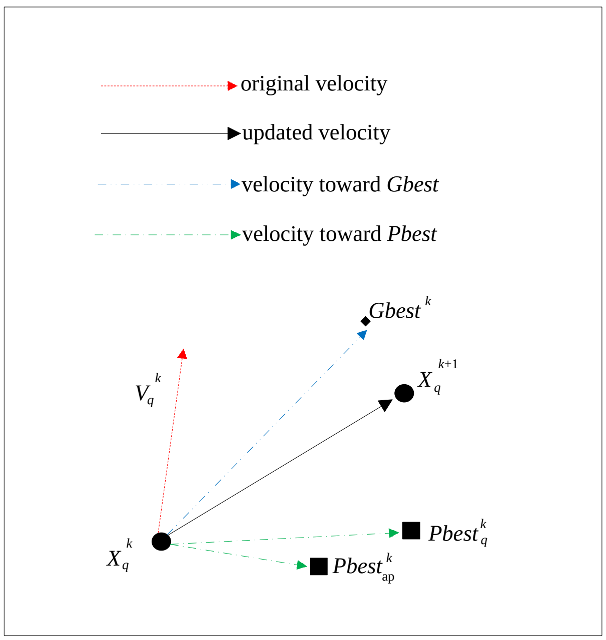

Fig. 4. Depiction of the velocity and position updates in the improved PSO

DOI: 10.6119/JMST-015-1215-1

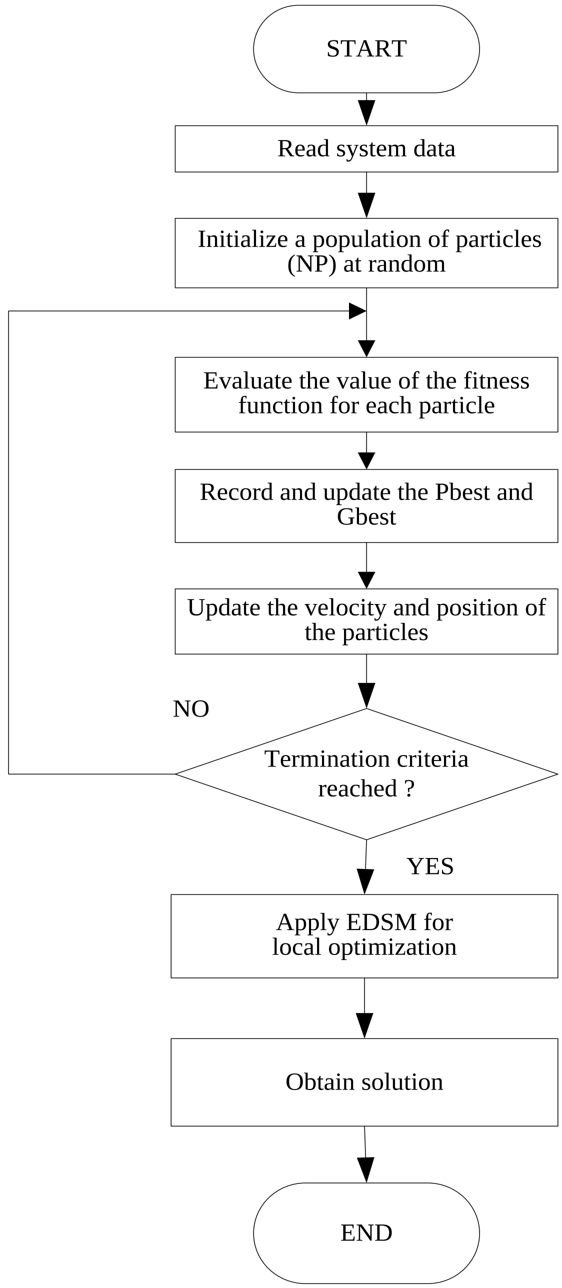

Fig. 5. Simplified flow chart for the conventional hybrid DPSO-EDSM algorithm.

DOI: 10.6119/JMST-015-1215-1

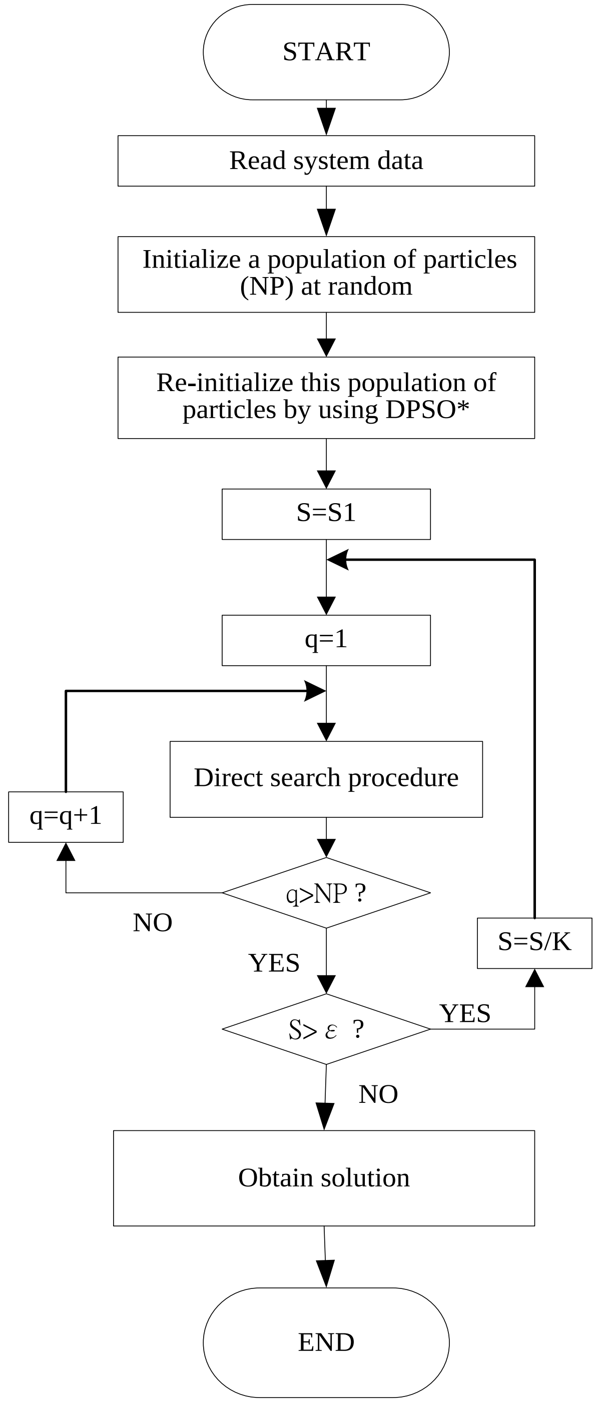

Fig. 6. Simplified flow chart for the proposed hybrid DPSO*-EDSM algorithm.

DOI: 10.6119/JMST-015-1215-1

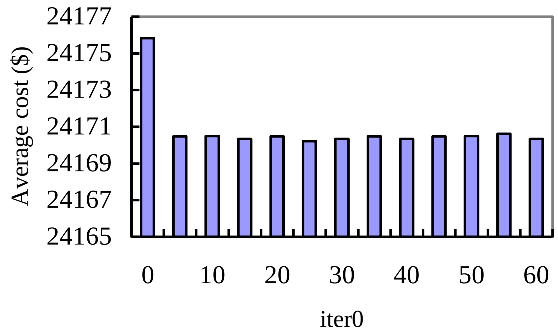

Fig. 7. Comparison of average costs under various iter0 for the system Example 1.

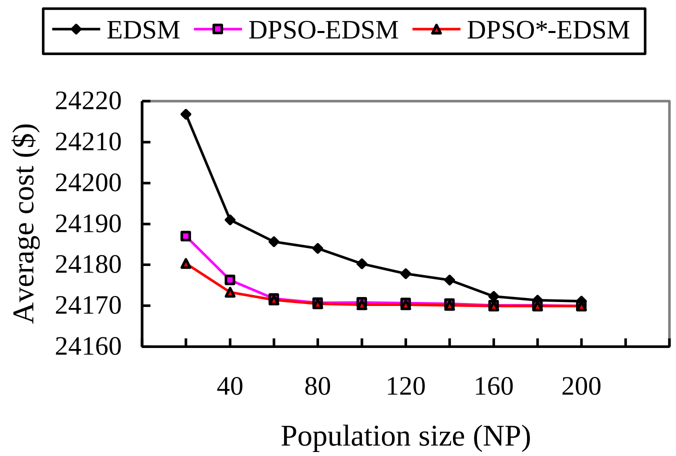

Fig. 8. Comparison of average costs under various NP for the system Example 1.

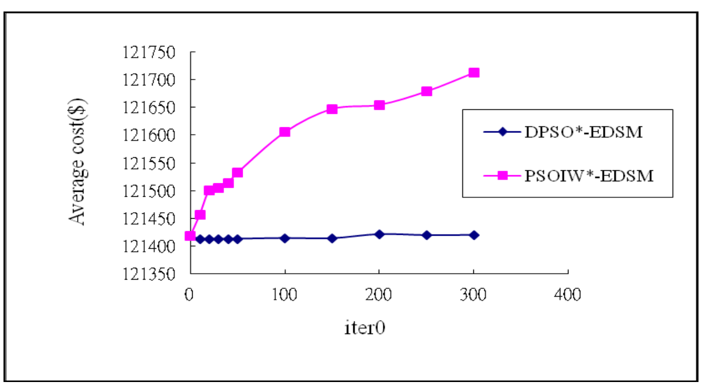

Fig. 9. Comparison of average costs under various iter0 of the two PSO strategies for the system Example 2.

DOI: 10.6119/JMST-015-1215-1

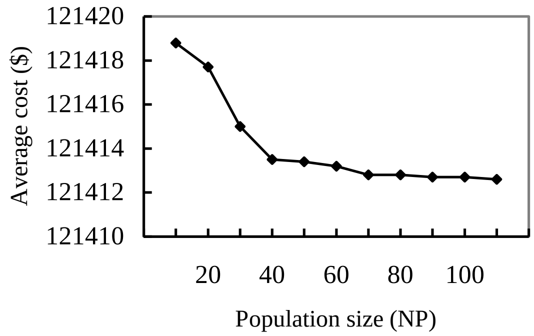

Fig. 10. Comparison of average costs under various NP for the system Example 2.

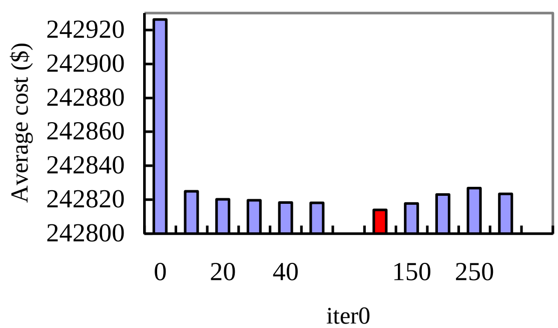

Fig. 11. Comparison of average costs under various iter0 for the system Example 3.