Journal of Marine Science and Technology

Volume 23 | Issue 1 Article 6

NEW INTELLIGENT PARTICLE SWARM OPTIMIZATION ALGORITHM FOR SOLVING ECONOMIC DISPATCH WITH VALVE-POINT EFFECTS

Jie Lin
School of Energy and Power Engineering, Wuhan University of Technology, Wuhan, P.R. China.

Chun-Lung Chen
Department of Marine Engineering, National Taiwan Ocean University, Keelung, Taiwan, R.O.C., cclung@mail.ntou.edu.tw

Shun-Feng Tsai
Department of Marine Engineering, National Taiwan Ocean University, Keelung, Taiwan, R.O.C.

Chengqing Yuan
School of Energy and Power Engineering, Wuhan University of Technology, Wuhan, P.R. China.

Recommended Citation
Lin, Jie; Chen, Chun-Lung; Tsai, Shun-Feng; and Yuan, Chengqing (2015) "NEW INTELLIGENT PARTICLE SWARM OPTIMIZATION ALGORITHM FOR SOLVING ECONOMIC DISPATCH WITH VALVE-POINT EFFECTS," *Journal of Marine Science and Technology*: Vol. 23: Iss. 1, Article 6.
DOI: 10.6119/JMST-013-1231-2
Available at: https://jmstt.ntou.edu.tw/journal/vol23/iss1/6

NEW INTELLIGENT PARTICLE SWARM OPTIMIZATION ALGORITHM FOR SOLVING ECONOMIC DISPATCH WITH VALVE-POINT EFFECTS

Acknowledgements

This work was supported by the National Science Council, Taiwan, ROC, under Grant NSC99-2221-E-019-044.

Journal of Marine Science and Technology, Vol. 23, No. 1, pp. 44-53 (2015)
DOI: 10.6119/JMST-013-1231-2

NEW INTELLIGENT PARTICLE SWARM OPTIMIZATION ALGORITHM FOR SOLVING ECONOMIC DISPATCH WITH VALVE-POINT EFFECTS

Jie Lin[1], Chun-Lung Chen[2], Shun-Feng Tsai[2], and Chengqing Yuan[1]

Key words: nonconvex economic dispatch, valve-point effect, particle swarm optimization, intelligent particle swarm optimization, direct search method.

ABSTRACT

With increasing fuel prices and restructuring in the power industry, nonconvex economic dispatch (NED) is expected to become crucial because of nonsmooth cost functions. This paper presents an intelligent particle swarm optimization for economic dispatch with valve-point effect. A new index, another particle best ($Pbest_{ap}$), is incorporated into the particle swarm optimization to further improve social behavior. Moreover, a novel diversity-based judgment mechanism for evaluating $Pbest_{ap}$ behavior is proposed for maintaining population diversity, which facilitates identification of the near-global region. The direct search algorithm is used to fine-tune and determine the eventual global optimal solution at low computational expense. Numerical experiments demonstrate that the proposed approach offers higher quality solutions than do several existing techniques.

INTRODUCTION

Maintaining an economic, secure, and reliable generation schedule is critical in modern energy management systems. Rising fuel prices and the progressive exhaustion of traditional fossil energy sources have intensified interest in economic dispatch (ED). Optimizing unit output scheduling can save millions of U.S. dollars in production costs annually. ED is aimed at determining the power output combination of online generating units that minimizes fuel costs while simultaneously satisfying all unit and system equality and inequality constraints. Generally, the fuel cost function for generation units has been approximately represented as a quadratic function and solved using classical optimization techniques, such as the lambda approach, the gradient method, linear programming, and Newton's methods (Wood and Wollenberg, 1996). The lambda iteration is a widely used approach that uses marginal cost information to determine the optimal solution. Unfortunately, the generating units vary greatly in fuel cost functions because of the physical limitations of power plant components such as valve points and combined cycle units. In practice, these valve points generate several prohibited operating zones, and additional constraints affect the operating ranges of units that have prohibited operating zones. Even in a competitive electrical market, generator characteristics can change with commercial interests rather than only physical reality. In other words, generator operators may change their bid prices to increase profit. Classical calculus-based techniques, such as the lambda iteration dispatch, cannot be directly applied in this scenario because of the nonsmooth fuel cost function of the problem. The importance of nonconvex economic dispatch (NED) is thus likely to increase, and developing advanced NED algorithms is essential for optimizing dispatch results.

Fuel cost functions considering valve-point effects in ED further complicate the solution methodology. Dynamic programming can be used to solve ED without restricting the shape of the cost function, but this method suffers from dimensionality (Wood and Wollenberg, 1996) and local optimality (Liang and Glover, 1992). Therefore, several optimization algorithms based on stochastic searching techniques, including simulated annealing (Wong and Fung, 1993), genetic algorithms (GAs) (Walters and Sheble, 1993; Orero and Irving, 1996; Kim et al., 2002), tabu search algorithms (TSAs) (Khamsawang et al., 2002; Lin et al., 2002), evolutionary programming (EP) (Yang et al., 1996; Sinha et al., 2003; Venkatesh et al., 2003), and particle swarm optimization (PSO) (Gaing, 2003), are used to solve highly nonlinear NED problems. Among these, PSO, a new population-based evolu-

Paper submitted 05/30/12; revised 10/17/13; accepted 12/31/13. Author for correspondence: Chun-Lung Chen (e-mail: cclung@mail.ntou.edu.tw).
[1] School of Energy and Power Engineering, Wuhan University of Technology, Wuhan, P.R. China.
[2] Department of Marine Engineering, National Taiwan Ocean University, Keelung, Taiwan, R.O.C.

tionary computation technique, has received considerable attention because of its flexibility and efficiency. PSO was developed by Kennedy and Eberhart in 1995 (Kennedy and Eberhart, 1995) and was inspired by the flocking and schooling behaviors of birds and fish. The approach is one among the latest nature-inspired algorithms and is characterized by high performance and easy implementation. The PSO algorithm uses a parallel searching mechanism and provides a high probability of determining the global or near-global optimal NED solution. However, conventional PSO entails several problems. Like other stochastic techniques, the main drawback of PSO is its tendency to be easily trapped in a local optimal solution, particularly when handling NED problems with more local optima and heavier constraints. Conventional PSO requires further research to improve its performance and robustness.

The inclusion of a nonsmooth cost function increases the nonlinearity and number of local optima in the solution space. Usually, stochastic search techniques identify a near-global region but are slow in finely tuned local searches. By contrast, local searching techniques climb hills rapidly but are easily trapped in local minima. Several effective hybrid optimization methods combining stochastic and deterministic techniques have been proposed (Bhagwan Das and Patvardhan, 1999; Lin et al., 2001; Victoire and Jeyakumar, 2004; Niknam, 2010). In this study, to further increase the possibility of exploring the search space where the global optimal solution exists, an intelligent PSO (INPSO) combined with a direct search method (DSM) is developed for ED with valve-point effect. The INPSO algorithm is responsible for "global exploitation" and the DSM algorithm for "local optimization," with the current INPSO solutions used as the starting points. A simple procedure based on a repairing strategy involves determining the system solution for initialization. A new index, namely another particle best (Pbest_{ap}), is incorporated into the PSO to provide some of the information guiding to the global solution and provides additional exploration capacity for swarming. Moreover, a novel diversity-based judgment mechanism for evaluating Pbest_{ap} behavior is proposed for enhancing search capacity, increasing the probability of obtaining the global optimal solution. A local optimization technique that utilizes DSM (Chen and Chen, 2001) is used to fine-tune the final optimal solution exploration. Finally, numerical results illustrate the merits of the proposed hybrid INPSO-DSM algorithm.

II. FORMULATION OF ED WITH VALVE-POINT EFFECTS

ED solutions are aimed at minimizing the total fuel cost of power plants subject to the operating constraints of a power system. The objective function is formulated as follows:

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

where F_T is the total fuel cost and 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, the fuel cost of the generation unit is a 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, thermal units with multivalve steam turbines vary more in fuel cost functions. (Walters and Sheble, 1993) presented the input-output performance curve for typical multivalve thermal units. The fuel cost function is replaced by the following function, which considers 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.

The generating unit is 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, and $P_i^{\min}$ and $P_i^{\max}$ are the minimal and maximal generation limits, respectively, of unit i. The transmission losses are conventionally represented as

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

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

III. OVERVIEW OF CONVENTIONAL PSO

PSO is a population-based optimization approach. In conventional PSO, the positions of Q particles are candidate solutions to the N-dimensional problem, and the moves of the particles are considered the search process for improved solutions. In a physical N-dimensional search space, the position and velocity of particle q are represented as vectors $X_q = \{x_{q1}, x_{q2}, ..., x_{qN}\}$ and $V_q = \{v_{q1}, v_{q2}, ..., v_{qN}\}$ in the PSO algorithm. During the search process, the particle successively adjusts its position according to two best values: the best position of particle q, represented as $Pbest_q = \{x_{q1}^{Pbest}, x_{q2}^{Pbest}, ..., x_{qN}^{Pbest}\}$ and the best position achieved so far by any particle, represented as $Gbest = \{x_{1,Gbest}, x_{2,Gbest}, ..., x_{N,Gbest}\}$. By tracking $Pbest_q$ and *Gbest*, the global optimal may be obtained. Like other evolutionary algorithms, PSO involves numerous parameters that must be predefined. Acceleration constants $c1$ and $c2$, which control the maximal step size, are predetermined. Inertia weight ω controls the impact of the previous velocity of the particle on its current velocity. The benefits of appropriately selecting these parameters justify the effort involved in experimentally determination them. The modified velocity and position of each particle is calculated using the current velocity and distance from $Pbest_q$ to *Gbest*, as shown in the following formulae:

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

$$X_q^{k+1} = X_q^k + V_q^{k+1}, \quad q = 1, 2, ... Q \quad (8)$$

where V_q^k and X_q^k are the velocity and position of particle q in iteration k, respectively. $Pbest_q^k$ is the best value of fitness function achieved by particle q before iteration k and $Gbest^k$ is the best fitness function value achieved so far by any particle. $c1$ and $c2$ weight the stochastic acceleration terms that pull each particle toward $Pbest_q$ and *Gbest*, rand represents a random variable between 0.0 and 1.0, and ω is the inertia weight factor. ω is clearly an influencing factor that provides a well-balanced mechanism between global and local exploration abilities. Usually, ω decreases linearly during iterations and is calculated using the following expression (Shi and Eberhart, 1998).

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

where $\omega_{\max}$ and $\omega_{\min}$are the initial and final weights, respectively, $iter_{\max}$ is the maximal iteration count, and *iter* is the current iteration number. PSO is summarized as follows:

Step 1: Randomly generate an initial population of particles.
Step 2: Evaluate the fitness function value of each particle.
Step 3: Record and update *Pbest* and *Gbest*.
Step 4: Update the velocity and position of all particles according to (7) and (8).
Step 5: Repeat steps 2-4 until a criterion is satisfied.

IV. INPSO WITH LOCAL OPTIMIZATION

In conventional PSO, the movement of a particle (fish) is governed by three behaviors: inertia, cognitive, and social. Inertia behavior causes the particle to swarm in the previous direction (at its present velocity). Cognitive behavior enables the particle remember its previously visited best position (its previous experience; *Pbest*). Social behavior models the memory of the particle regarding the best position among the particles (the experience of its neighbors; *Gbest*). However, for social behavior to employ only *Gbest*, which is generally not the global optimal solution, containing parts of nonoptimal information, is unreasonable. The subsequent movement of the fish is often affected by the location of the fish that is in the best position and by the location of the other fish that it randomly observes when fish schools begin feeding. Therefore, conventional PSO experiences premature convergence and is easily trapped in local optima if a promising area where the global optimum is residing is not identified at the end of optimization.

1. Improving PSO by Adding the $Pbest_{ap}$ Item

To increase the possibility of exploring the search space where the global optimal solution exists, we follow a slightly different social behavior approach to further select the global best guide of the particle swarm. Social behavior comprises two phases: best particle position ever obtained (*Gbest*) and random another particle best position ($Pbest_{ap}$), namely, another behavior. After increasing another behavior to the social behavior, $Pbest_{ap}$ provides some of the information guiding to the global solution and affords additional exploration capacity for swarming. The new velocity update equation is given by:

$$V_q^{k+1} = c0 \times V_q^k + c1 \times rand \times (Pbest_q^k - X_q^k) + c2 \times rand \times (Gbest^k - X_q^k) + c3_q^{\mathrm{k}} \times rand \times (Pbest_{ap}^k - X_q^k) \quad (10)$$

$$X_q^{k+1} = X_q^k + V_q^{k+1} \quad q = 1, 2, ..., Q; \quad ap \neq q \quad (11)$$

where $c0$ is the inertia weight factor. $Pbest_{ap} = \{x_{ap1}^{Pbest}, x_{ap2}^{Pbest}, ..., x_{apN}^{Pbest}\}$ is the best position of a random another particle, called particle *ap*. $c3_q = \{c3_{q1}, c3_{q2}, ..., c3_{qN}\}$ is the weight factor of another behavior. Initial candidate solutions are usually far from the global optimum and hence a larger $c3_q$ may benefit global exploration. However, the difference in global best guides between *Gbest* and $Pbest_{ap}$ gradually decreases with successive iterations. Therefore, $c3_q$ decreases linearly and is calculated using the following expression.

$$c3_q = c3_{\max} - (c3_{\max} - c3_{\min}) \times \frac{\mathrm{iter}}{iter_{\max}}$$
$$q = 1, 2, \ldots Q; i = 1, 2, \ldots N \quad (12)$$

where $c3_{\max}$ and $c3_{\min}$ are the initial and final weights, respectively, $iter_{max}$ is the maximal iteration count, and $iter$ is the current iteration number.

2. Application of $Pbest_{ap}$ with a Diversity-Based Judgment Mechanism in PSO

Adding the $Pbest_{ap}$ item increases PSO search space and robustness. However, the information guiding to the global solution from $Pbest_{ap}$ may contain the best particle position ever obtained, *Gbest*. The random another particle best position cannot usually provide positive guidance. For maintaining population diversity, an intelligent judgment mechanism for evaluating the $Pbest_{ap}$ behavior is developed to provide facilitate identification of the near-global region. The new velocity of each particle is calculated using the following formulae.

$$V_q^{k+1} = c0 \times V_q^k + c1 \times rand \times (Pbest_q^k - X_q^k) + c2 \times rand \times (Gbest^k - X_q^k) - c3_q^k \times rand \times (Pbest_{ap}^k - X_q^{k'}),$$
$$if\,(x^k_{Gbest} - x^k_q) \times (x^k_{ap} - x^k_q) \geq 0 \quad (13)$$

$$V_q^{k+1} = c0 \times V_q^k + c1 \times rand \times (Pbest_q^k - X_q^k) + c2 \times rand \times (Gbest^k - X_q^k) + c3_q^k \times rand \times (Pbest_{ap}^k - X_q^{k'}),$$
$$if\,(x^k_{Gbest} - x^k_q) \times (x^k_{ap} - x^k_q) < 0 \quad (14)$$

The weight factor $c3_q$ maintains a wide spread of nondominated solutions. From (13), if $(x^k_{ap} - x^k_q)$ and $(x^k_{Gbest} - x^k_q)$ move in the same direction, the information guiding to the global solution from $Pbest_{ap}$ and *Gbest* is similar. Compared with *Gbest*, x^k_{ap} is a bad position, and the influence of particle *ap* to the movement of particle *q* is negative. Conversely, the information guiding to the global solution from $Pbest_{ap}$ and *Gbest* differs largely if $(x^k_{Gbest} - x^k_q)$ and $(x^k_{ap} - x^k_q)$ do not move in the same direction. As shown in (14), the influence of particle *ap* on the movement of particle *q* is positive. The most attractive feature of the intelligent judgment mechanism for evaluating the aforementioned $Pbest_{ap}$ behavior is its ability to maintain population diversity, which increases the possibility of escaping local optimal solution traps.

3. Local Optimization Using DSM

The DSM algorithm appears to be the optimal choice for local optimization because of its simplicity, computational efficiency, and easy implementation. A salient feature of DSM is that it begins with an initial feasible solution and searches for the optimal solution along a trajectory that continually maintains a feasible solution. DSM advantageously handles several inequality constraints without introducing multipliers. Furthermore, DSM solves problems involving unavailable derivatives and complex fuel cost functions. Moreover, DSM gradually reduces the step size by using the multilevel convergence strategy to increase the possibility of escaping local optimal solution traps. Numerical results demonstrate that DSM rapidly identifies a near-global region and performs a local search. This efficient approach is appropriate for assessing NED costs. However, like many local search techniques, DSM is sensitive to the starting points. In addition, selecting calculation step S is vital for successfully obtaining the global optimal solution. Empirically, an appropriate initial calculation step S_1 is 10%-20% of the largest generation unit in the power system. Depending on the number of local minimal points in the cost functions, the recommended value of the reduced factor K is 1.1-3.0. The extended economic dispatch problem is solved as described in (Chen and Chen, 2001). The proposed INPSO-DSM algorithm is outlined in Fig. 1.

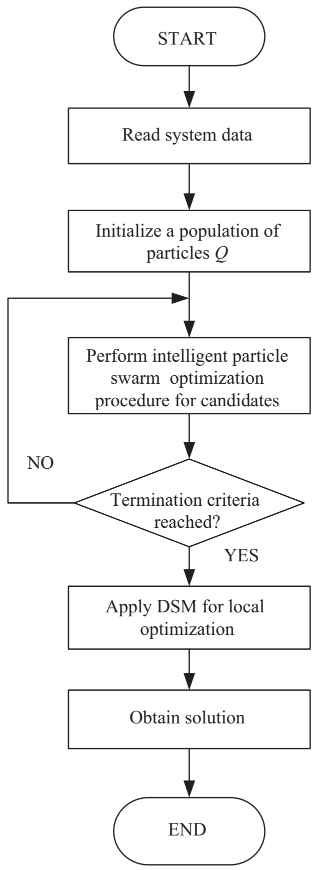

Fig. 1. Flow chart for the proposed INPSO-DSM algorithm.

V. SOLVING AND IMPLEMENTING INPSO-DSM

The primary computational processes of the algorithm presented in this paper for solving ED with valve-point power systems are detailed here. This algorithm is an implementation of INPSO-DSM.

Step 1: Establish the INPSO-DSM parameters.

Establish the set of INPSO-DSM parameters, such as the number of particles Q; weighting factors $c0$, $c1$, $c2$, $c3_{\max}$, and $c3_{\min}$; the maximal number of iterations $iter_{\max}$; initial calculation step S_1; and reduced factor K.

Step 2: Define the particle elements.
A particle is a solution in INPSO-DSM. Each particle contains elements that represent the actual power generation of the generators. Eq. (15) shows a particle q:

$$X_q^k = [P_1^k, P_2^k, ..., P_i^k, ..., P_N^k], \quad q = 1, 2, ..., Q \tag{15}$$

Step 3: Randomly generate an initial population of particles.
Let rand be a uniform random value in the range [0,1]. The initial power outputs of $N - 1$ thermal generating units without violating (5) are generated randomly by using

$$P_i = P_i^{\min} + rand \times (P_i^{\max} - P_i^{\min}) \tag{16}$$

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

$$P_d = P_D + P_{Loss} - \sum_{\substack{i=1 \\ i \neq d}}^{N} P_i \tag{17}$$

P_d can be calculated directly by using a quadratic equation, as shown in (Wong and Fung, 1993). If P_d violates (5), a repairing strategy is applied to randomly select a unit to increase (or reduce) its output by the random or predefined step (e.g., 10 MW) sequentially until all constraints are satisfied.

Step 4: Evaluate the fitness of each particle.
Calculate the fitness function value of each particle. The fitness function is an index for evaluating the fitness of particles. Eq. (1) is the fitness function of the ED problem.

Step 5: Record and update *Pbest* and *Gbest*.
The two best values are recorded in the searching process. Each particle tracks coordinates in the solution space associated with the best solution reached so far, which is recorded as *Pbest*. The overall best value obtained by any particle is recorded as *Gbest*.

Step 6: Update the velocity and position of the particles.
Eqs. (18)-(20) update the velocity and position of the 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 and represents a movement of a particle.

$$\begin{aligned} V_{qi}^{k+1} &= c0 \times V_{qi}^{k} + c1 \times rand \times (Pbest_{qi}^{k} - X_{qi}^{k}) + c2 \\ &\times rand \times (Gbest_{i}^{k} - X_{qi}^{k}) - c3_{qi} \times rand \\ &\times (Pbest_{api}^{k} - X_{qi}^{k}), \\ &if\, (x^k_{i,Gbest} - x^k_{qi}) \times (x^k_{api} - x^k_{qi}) \geq 0 \end{aligned} \tag{18}$$

$$\begin{aligned} V_{qi}^{k+1} &= c0 \times V_{qi}^{k} + c1 \times rand \times (Pbest_{qi}^{k} - X_{qi}^{k}) + c2 \\ &\times rand \times (Gbest_{i}^{k} - X_{qi}^{k}) + c3_{qi} \times rand \\ &\times (Pbest_{api}^{k} - X_{qi}^{k}), \\ &if\, (x^k_{i,Gbest} - x^k_{qi}) \times (x^k_{api} - x^k_{qi}) < 0 \end{aligned} \tag{19}$$

$$\begin{aligned} &X_{qi}^{k+1} = X_{qi}^{k} + V_{qi}^{k+1} \\ &q = 1, 2, ..., Q;\ i = 1, 2, ..., N;\ ap \neq q \end{aligned} \tag{20}$$

The new positions of the particles are forced to satisfy the unit's generation limit constraint yielded by (5) and any other existing constraints.

Step 7: Verify the end condition.
Upon reaching the maximal number of iterations, invoke the DSM algorithm, using the current solutions of the INPSO as the starting points for further exploring the final optimal solution; repeat steps 4-6 until the end conditions are satisfied.

VI. NUMERICAL EXPERIMENTS

To verify the feasibility and effectiveness of the proposed algorithm, numerical studies are performed for the two test systems where valve-point effects are considered, one with 13 generators and another with 40 generators. All computations are performed on a Pentium (R) Dual-Core 3.0 GHz PC, and the following computer programs are developed in FORTRAN:

PSO: Basic PSO
PSO-IW: PSO with inertia weight
CNPSO: PSO using common another particle behavior
INPSO: CNPSO with a diversity-based judgment mechanism
INPSO*: INPSO with local optimization

Table 1 lists the optimal parameter settings determined after testing and evaluating different parameter combinations for the PSO, PSO-IW, CNPSO, INPSO, and INPSO* algorithms. The studied cases are detailed herein:

1. Example 1: 13-Unit System

A system with 13 generating units that considers valve-point effects is studied to test the solution quality and performance of the proposed algorithm. The system unit data is shown in Table 2 (Sinha et al., 2003); the total load demand is 2520 MW. To facilitate comparison, the network losses of the

Table 1. Best parameter setting of the five PSO strategies.

Parameter	PSO	PSO-IW	CNPSO/INPSO	INPSO*
Example 1 and Example 2	$Q = 300$; $iter_{max} = 2000$; $c1 = 2.0$; $c2 = 2.0$	$Q = 300$; $iter_{\max} = 2000$; $c1 = 2.0$; $c2 = 2.0$; $\omega_{\max} = 0.9$; $\omega_{\min} = 0.4$	$Q = 300$; $iter_{\max} = 2000$; $c0 = 0.3$; $c1 = 2.5$; $c2 = 0.8$; $c3_{\max} = 0.4$; $c3_{\min} = 0.01$	$Q = 300$; $iter_{\max} = 2000$; $c0 = 0.3$; $c1 = 2.5$; $c2 = 0.8$; $c3_{\max} = 0.4$; $c3_{\min} = 0.01$; S1 = 120, K = 1.2

Table 2. Parameters for the 13-unit system.

Unit No.	$P_i^{\max}$	$P_i^{\min}$	a_i	b_i	c_i	e_i	f_i
1	680	0	550	8.1	0.00028	300	0.035
2	360	0	309	8.1	0.00056	200	0.042
3	360	0	307	8.1	0.00056	200	0.042
4	180	60	240	7.74	0.00324	150	0.063
5	180	60	240	7.74	0.00324	150	0.063
6	180	60	240	7.74	0.00324	150	0.063
7	180	60	240	7.74	0.00324	150	0.063
8	180	60	240	7.74	0.00324	150	0.063
9	180	60	240	7.74	0.00324	150	0.063
10	120	40	126	8.6	0.00284	100	0.084
11	120	40	126	8.6	0.00284	100	0.084
12	120	55	126	8.6	0.00284	100	0.084
13	120	55	126	8.6	0.00284	100	0.084

Table 3. Comparison of dispatch results for the load of 2520 MW in the system Example 1.

Unit	HSS	TSA	EP-SQP	PSO-SQP	INPSO*
1	628.23	628.319	628.3136	628.3205	628.3185
2	299.22	299.1993	299.1715	299.0524	299.1990
3	299.17	331.8975	299.0474	298.9681	299.1990
4	159.12	159.7305	159.6399	159.4680	159.7330
5	159.95	159.7331	159.6560	159.1429	159.7330
6	158.85	159.7306	158.4831	159.2724	159.7328
7	157.26	159.7334	159.6749	159.5371	159.7328
8	159.93	159.7308	159.7265	158.8522	159.7329
9	159.86	159.7316	159.6653	159.7845	159.7329
10	110.78	40.0028	114.0334	110.9618	77.3996
11	75.00	77.3994	75.0000	75.0000	77.3996
12	60.00	92.3932	60.0000	60.0000	92.3998
13	92.62	92.3986	87.5884	91.6401	87.6868
Cost ($/h)	24275.71	24313	24266.44	24261.05	24169.92

system are ignored. Traditional approaches, such as the lambda iteration, cannot be used to solve this problem because of their nonsmooth fuel cost functions. The obtained optimal result using the proposed INPSO* is compared with those of the following earlier studies: HSS (Bhagwan Das and Patvardhan, 1999), TSA (Khamsawang et al., 2002), EP-SQP (Victoire and Jeyakumar, 2004), and PSO-SQP (Victoire and Jeyakumar, 2004) (Table 3). From these results, although multiple local minimal solutions exist in the studied case, the proposed INPSO* obtains a solution ($24169.92) superior to those obtained using the other approaches. Thus, the appropriateness of the algorithm presented in this paper for obtaining optimal NED is confirmed.

The INPSO optimization procedure is compared with typical PSO, PSO-IW, and CNPSO runs in Fig. 2, illustrating the convergence property of the proposed algorithm. In this test, the same initial random solution ($25293.98) is used in the four PSO strategies. The results reveal that INPSO offers an excellent convergence property for determining the optimal solution ($24169.92). Moreover, the results demonstrate that the total number of iterations required for achieving the optimal solution is approximately 60 using INPSO and 220 using

Table 4. Comparison of results after 100 trials for the system Example 1.

Methods	Minimum cost ($)	Average cost ($)	Maximum cost ($)	NTO	ACT (s)
PSO	24392.76	24565.24	24714.27	0	1.9448
PSO-IW	24287.9	24413.04	24695.04	0	2.2325
CNPSO	24169.95	24266.33	24444.13	0	1.8999
INPSO	24169.92	24182.91	24223.88	20	2.0675
INPSO*	24169.92	24169.92	24169.92	100	2.086

NTO: number of times to reach optimal solution ($ 24169.92).
ACT: average computation time for 100 trail tests.

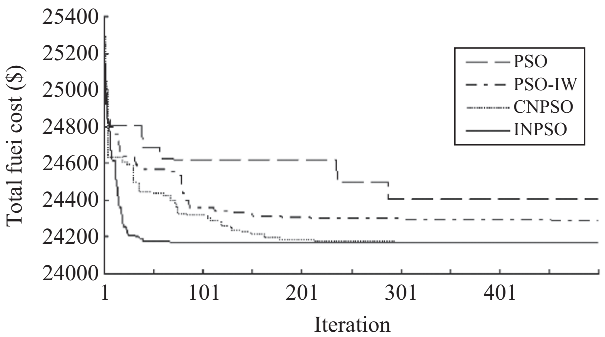

Fig. 2. Comparative convergence behaviors of the four PSO strategies for the 13-unit example system.

CNPSO. PSO-IW exhibits premature convergence and is easily trapped in local optima ($24392.76). The simulation results clearly demonstrate that the proposed diversity-based judgment mechanism for evaluating $Pbest_{ap}$ provides an improved property during optimization.

Because of the randomness of heuristic algorithms, their performance cannot be judged from a single run. Many trials with different initial conditions are necessary to reach a fair conclusion. Table 4 reports the lowest, average, and highest costs achieved for 100 trial runs. The results highlight the superiority of the INPSO and INPSO* algorithms over the basic PSO. The proposed INPSO* algorithm reaches the optimal NED solution ($24169.92) with a high probability, demonstrating the benefit of integrating DSM into INPSO. In these test cases, the proposed INPSO* easily obtains satisfactory solutions, and the average computing time is approximately 2 s. The encouraging simulation results reveal that INPSO* can obtain higher-quality NED solutions.

2. Example 2: 40-Unit System

In this example, to demonstrate the robustness and effectiveness of the proposed INPSO* algorithm, the simulation includes test runs for the large-scale 40-unit system used in (Sinha et al., 2003). Many local optimal solutions exist for the dispatch problem, which is thus well-suited for testing and validating the developed algorithm. The system unit data is listed in Table 5; the load demand is 10500 MW. This example problem has previously been solved using IFEEP (Sinha et al., 2003), MPSO (Park et al., 2005), PSO-SQP (Victoire and

Table 5. Parameters for the 40-unit system.

Unit No.	P_i^{max}	P_i^{min}	a_i	b_i	c_i	e_i	f_i
1	114.0	36.0	94.700	6.73	0.00690	100.0	0.084
2	114.0	36.0	94.705	6.73	0.00690	100.0	0.084
3	120.0	60.0	309.54	7.07	0.02028	100.0	0.084
4	190.0	80.0	369.03	8.18	0.00942	150.0	0.063
5	97.0	47.0	148.89	5.35	0.0114	120.0	0.077
6	140.0	68.0	222.33	8.05	0.01142	100.0	0.084
7	300.0	110.0	287.71	8.03	0.00357	200.0	0.042
8	300.0	135.0	391.98	6.99	0.00492	200.0	0.042
9	300.0	135.0	455.76	6.60	0.00573	200.0	0.042
10	300.0	130.0	722.82	12.9	0.00605	200.0	0.042
11	375.0	94.0	635.20	12.9	0.00515	200.0	0.042
12	375.0	94.0	654.69	12.8	0.00569	200.0	0.042
13	500.0	125.0	913.40	12.5	0.00421	300.0	0.035
14	500.0	125.0	1760.4	8.84	0.00752	300.0	0.035
15	500.0	125.0	1728.3	9.15	0.00708	300.0	0.035
16	500.0	125.0	1728.3	9.15	0.00708	300.0	0.035
17	500.0	220.0	647.85	7.97	0.00313	300.0	0.035
18	500.0	220.0	649.69	7.95	0.00313	300.0	0.035
19	550.0	242.0	647.83	7.97	0.00313	300.0	0.035
20	550.0	242.0	647.81	7.97	0.00313	300.0	0.035
21	550.0	254.0	785.96	6.63	0.00298	300.0	0.035
22	550.0	254.0	785.96	6.63	0.00298	300.0	0.035
23	550.0	254.0	794.53	6.66	0.00284	300.0	0.035
24	550.0	254.0	794.53	6.66	0.00284	300.0	0.035
25	550.0	254.0	801.32	7.10	0.00277	300.0	0.035
26	550.0	254.0	801.32	7.10	0.00277	300.0	0.035
27	150.0	10.0	1055.1	3.33	0.52124	120.0	0.077
28	150.0	10.0	1055.1	3.33	0.52124	120.0	0.077
29	150.0	10.0	1055.1	3.33	0.52124	120.0	0.077
30	97.0	47.0	148.89	5.35	0.01140	120.0	0.077
31	190.0	60.0	222.92	6.43	0.00160	150.0	0.063
32	190.0	60.0	222.92	6.43	0.00160	150.0	0.063
33	190.0	60.0	222.92	6.43	0.00160	150.0	0.063
34	200.0	90.0	107.87	8.95	0.0001	200.0	0.042
35	200.0	90.0	116.58	8.62	0.0001	200.0	0.042
36	200.0	90.0	116.58	8.62	0.0001	200.0	0.042
37	110.0	25.0	307.45	5.88	0.0161	80.0	0.098
38	110.0	25.0	307.45	5.88	0.0161	80.0	0.098
39	110.0	25.0	307.45	5.88	0.0161	80.0	0.098
40	550.0	242.0	647.83	7.97	0.00313	300.0	0.035

Table 6. Comparison of results of different methods for the system Example 2.

Methods	Minimum cost ($)	Average cost ($)	Maximum cost ($)
IFEP	122624.35	125740.63	123382
PSO-SQP	122094.67	122245.25	---
MPSO	122252.265	---	---
NPSO-LRS	121664.4308	122209.3185	122981.5913
CSO	121461.6707	121936.1926	122844.5391
TSARGA	121463.07	122928.31	124296.54
GA-PS-SQP	121458	122039	---
HMAPSO	121586.90	121586.90	121586.90
SOH-PSO	121501.14	121853.57	122446.30
MTS	121532.10	121798.51	122022.15
PSO-MSAF	121423.23	---	---
θ-PSO	121420.9027	121509.8423	121852.4249
INPSO	121412.6	121481.7	121622.6
INPSO*	121412.6	121437.6	121538.4

Table 7. Best dispatch results for the forty-unit system.

Unit No.	P_i	Unit No.	P_i	Unit No.	P_i	Unit No.	P_i
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

Jeyakumar, 2004), NPSO-LRS (Selvakumar and Thanushkodi, 2007), CSO (Selvakumar and Thanushkodi, 2009), TSARGA (Subbaraj et al., 2011), GA-PS-SQP (Alsumait et al., 2010), HMAPSO (Kumar et al., 2011), SOH-PSO (Chaturvedi et al., 2008), MTS (Sa-ngiamvibool et al., 2011), PSO-MSAF (Subbaraj et al., 2010), and θ-PSO (Hosseinnezhad and Babaei, 2013). In Table 6, the corresponding costs of the optimal solution obtained using INPSO and INPSO* are compared with those of the aforementioned approaches. These results show that the proposed INPSO* yields a solution ($121412.6) superior in minimal and average costs to those obtained in previous research. Table 7 contains details of the best solutions obtained using the proposed INPSO* algorithm. The results show that the proposed INPSO* algorithm is accurate and efficiently solves complex NED problems.

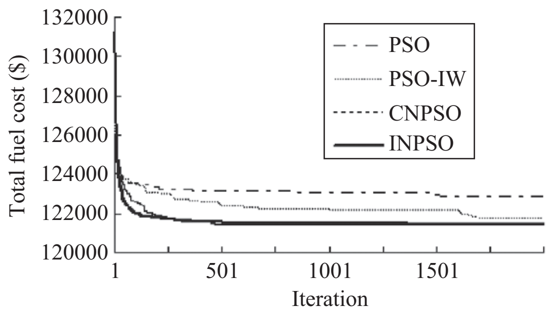

Fig. 3. Comparative convergence behaviors of the four PSO strategies for the 40-unit example system.

The CNPSO and INPSO optimization procedures are compared with typical PSO and PSO-IW runs in Fig. 3, demonstrating the strong convergence property of the proposed algorithm. In the test, the same initial random starting points ($131251.6) are used in the basic PSO, PSO-IW, CNPSO, and INPSO algorithms. As shown in Fig. 3, the basic PSO exhibits premature convergence and is easily trapped in local optima1 ($123516.4) at the 56th iteration. Similarly, PSO-IW is trapped in a local optimal solution ($122206.2) because a promising area where the global optimal exists is unidentified at the end of optimization. The satisfactory solution ($121600.9) achieved by INPSO decreases rapidly before the 300th iteration and the global optimal solution ($121412.6) is obtained in the 881st iteration. The improved social behavior mechanism is effective, and the algorithm converges rapidly. Moreover, the final

Table 8. Comparison of results after 100 trials for the system Example 2.

Methods	Minimum cost ($)	Average cost ($)	Maximum cost ($)	NTO	ACT (s)
PSO	122838.9	123006.4	123198.5	0	6.025
PSO-IW	121745.6	121940.1	122233.7	0	6.5856
CNPSO	121417.6	121685.4	124647.0	0	5.9314
INPSO	121412.6	121481.7	121622.6	13	6.416
INPSO*	121412.6	121437.6	121538.4	33	6.8023

NTO: number of times to reach optimal solution ($121412.6).

ACT: average computation time for 100 trail tests.

results of INPSO are better than those of PSO and PSO-IW. The ability of the proposed INPSO algorithm to escape local optimal traps is thus confirmed.

To investigate the effects of initial trial solutions on the final results, different initial random solutions are used in PSO, PSO-IW, CNPSO, INPSO, and INPSO*. Table 8 reports the least, average, and highest costs calculated using the five PSO strategies for 100 trial runs. In these test cases, the proposed INPSO easily obtains satisfactory solutions by using the intelligent judgment mechanism. However, only the near-global optimal solution is obtained using the proposed approach. INPSO reached the global optimal solution ($121412.6) 13 times, whereas CNPSO did so 0 times in the test cases. The basic PSO offers no guarantee that the solutions are optimal or even close to the optimal solution. As the data in the sixth row of Table 8 reveals, INPSO* produced the optimal solution ($121412.6) 33 times, demonstrating its effectiveness, reliably, and efficiency. This test case study converges within 6.8 s in each run when Q is 300. Various load demands were studied and the results show that the proposed INPSO* algorithm successfully escapes the local optimal traps. This approach accurately solves NED problems.

VII. CONCLUSIONS

This paper presents a hybrid algorithm that combines INPSO and DSM to solve ED problems with valve-point effects. Using the $Pbest_{ap}$ item with a diversity-based judgment mechanism, the proposed PSO algorithm facilitates identification of the near-global region. Moreover, a local optimization technique that utilizes DSM is used to fine-tune and determine the eventual global optimal solution at low computational expenses. The proposed INPSO-DSM provides the global optimal solution with a high probability for ED problems with valve-point effects. Large-scale NED problems can be solved using the proposed algorithm. Furthermore, numerical experiments demonstrate that the proposed algorithm is more practical and valid than many existing NED solutions.

ACKNOWLEDGMENTS

This work was supported by the National Science Council, Taiwan, ROC, under Grant NSC99-2221-E-019-044.

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.

Chaturvedi, K. T., M. Pandit and L. Srivastava (2008). Self-organizing hierarchical particle swarm optimization for nonconvex economic dispatch. IEEE Transactions on Power Systems 23, 1079-1087.

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

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

Hosseinnezhad, V. and E. Babaei (2013). Economic load dispatch using θ-PSO. International Journal of Electrical Power & Energy Systems 49, 160-169.

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

Khamsawang, S., C. Boonseng and S. Pothiya (2002). Solving the economic dispatch problem with Tabu search algorithm. IEEE International Conference on Industrial Technology 1, 274-278.

Kim, J. O., D. J. Shin, J. N. Park and C. Singh (2002). Atavistic genetic algorithm for economic dispatch with valve point effect. Electric Power Systems Research 62, 201-207.

Kumar, R., D. Sharma and A. Sadu (2011). A hybrid multi-agent based particle swarm optimization algorithm for economic power dispatch. International Journal of Electrical Power & Energy Systems 33, 115-123.

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

Lin, W. M., F. S. Cheng and M. T. Tsay (2001). Nonconvex economic dispatch by integrated artificial intelligence. IEEE Transactions on Power Systems 16, 307-311.

Lin, W. M., F. S. Cheng and M. T. Tsay (2002). An improved Tabu search for economic dispatch with multiple minima. IEEE Transactions on Power Systems 17, 108-112.

Niknam, T. (2010). A new fuzzy adaptive hybrid particle swarm optimization algorithm for non-linear, non-smooth and non-convex economic dispatch problem. Applied Energy 87, 327-339.

Orero, S. O. and M. R. Irving (1996). Economic dispatch of generators with prohibited operating zones: a genetic algorithm approach. IEE Proceedings-Generation, Transmission and Distribution 143, 529-534.

Park, J. B., K. S. Lee, J. R. Shin and K. Y. Lee (2005). A particle swarm optimization for economic dispatch with nonsmooth cost functions. IEEE Transactions Power Syst. 20, 34-42.

Sa-ngiamvibool, W., S. Pothiya and I. Ngamroo (2011). Multiple tabu search algorithm for economic dispatch problem considering valve-point effects.

International Journal of Electrical Power & Energy Systems 33, 846-854.

Selvakumar, A. I. and K. Thanushkodi (2007). A new particle swarm optimization solution to nonconvex economic dispatch problems. IEEE Transactions Power Systems 22, 42-51.

Selvakumar, A. I. and K. Thanushkodi (2009). Optimization using civilized swarm: Solution to economic dispatch with multiple minima. Electric Power System Research 79, 8-16.

Shi, Y. and R. Eberhart (1998). A modified particle swarm optimizer. IEEE International Conference on Evolutionary Computation Proceedings, Anchorage, Alaska, 69-73.

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

Subbaraj, P., R. Rengaraj and S. Salivahanan (2011). Enhancement of self-adaptive real coded genetic algorithm using taguchi method for economic dispatch problem. Applied Soft Computing 11, 83-92.

Subbaraj, P., R. Rengaraj, S. Salivahanan and T. R. Senthilkumar (2010). Parallel particle swarm optimization with modified stochastic acceleration factors for solving large scale economic dispatch problem. International Journal of Electrical Power & Energy Systems 32, 1014-1023.

Venkatesh, P., R. Gnanadass and N. P. Padhy (2003). Comparsion and application of evolutionary programming techniques to combined economic emission dispatch with line flow constraints. IEEE Transactions on Power Systems 18, 688-697.

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 Transactions on Power Systems 8, 1325-1331.

Wong, K. P. and C. C. Fung (1993). Simulated annealing based economic dispatch algorithm. IEE Proceedings-Generation, Transmission and Distribution 140, 509-515.

Wood, J. and B. F. Wollenberg (1996). Power Generation Operation and Control. 2nd ed, Wiley, New York, 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 Transactions on Power Systems 11, 112-118.