



Sharif University of Technology

Scientia Iranica

Transactions D: Computer Science & Engineering and Electrical Engineering

www.scientiairanica.com



# Improved particle swarm optimization for combined heat and power economic dispatch

T.T. Nguyen and D.N. Vo\*

Power System Optimization Research Group, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, No. 19, Nguyen Huu Tho str., 7th dist., Ho Chi Minh City, Vietnam.

Received 3 March 2015; received in revised form 3 December 2015; accepted 8 February 2016

## KEYWORDS

Improved particle swarm optimization;  
Combined heat and power economic dispatch;  
Pure power units;  
Combined heat and power units;  
Pure heat units;  
Quadratic fuel cost function;  
Nonconvex fuel cost function.

**Abstract.** This paper presents eight Improved Particle Swarm Optimization (IPSO) algorithms for solving Combined Heat and Power Economic Dispatch (CHPED) problem with valve point loading effects on fuel cost function and transmission losses. The main objective of the problem is to find the power output and the heat output of available units so that the total fuel cost is minimized while satisfying power and heat demands and power and heat limits. The proposed IPSO algorithms are based on some modifications on the parameters and the use of Cauchy distribution. The effectiveness of the IPSO methods has been validated through six different test systems including three systems with quadratic fuel cost function of pure power units neglecting transmission losses, two systems with nonconvex fuel cost function of pure power units considering transmission losses, and one large-scale system with nonconvex fuel cost function of pure units and without transmission losses. The result comparisons between the IPSO methods and other methods in the literature have indicated that the proposed methods can obtain higher solution quality with faster computational time than many other methods. Therefore, the proposed IPSO methods would be very efficient methods for solving the nonconvex CHPED problem.

© 2016 Sharif University of Technology. All rights reserved.

## 1. Introduction

Economic load dispatch has become one of the most important problems in electrical power system operation as it can enable the system consisting of thermal units to produce electricity possibly with the minimum fuel cost of generation. However, there is a fact that the benefit can be higher as heat from the electrical generation process released into the air is utilized to supply to industrial zones or manufacturers [1]. The best way to minimize the operating cost of the thermal units is to use both heat and electricity under working condition. Consequently, the optimal operation of combined heat and power units has played a very important role in enhancement of the efficiency of

power system operation. The generation process of both electricity and heat, known as combined heat and power generation, can reduce the emission release into the air, avoiding greenhouse effect [2].

Several methods have been applied for solving the Combined Heat and Power Economic (CHPED) problem, including conventional methods and meta-heuristic methods such as Newton Method [3], Lagrange Relaxation (LR) [4], Genetic Algorithm (GA) [5], Improved Ant Colony Search (IACSA) [6], Evolutionary Programming (EP) [7], Improved Genetic Algorithm with Multiplier Updating (IGA-MU) [8], Lagrange Relaxation and Sequential Quadratic Programming (LR-SQP) method [9], Self-Adaptive Real-coded Genetic Algorithm (SARGA) [10], Bee Colony Optimization (BCO) [11], Harmony Search (HS) [1,12-13], Mesh Adaptive Direct Search Algorithm (MADSA) [14], Novel Direct Search (NDS) [15], Artifi-

\*. Corresponding author. Tel: +84-8-3864 7256  
E-mail address: vongocdieu@tdt.edu.vn (D.N. Vo)

cial Immune System (AIS) [16], Lagrangian Relaxation with Surrogate Subgradient Multiplier Updates (LR-SSMU) [17], Selective Particle Swarm Optimization (SPSO) [18], Particle Swarm Optimization with Time Varying Acceleration Coefficients (TVAC-PSO) [19], Improved Group Search Optimization (IGSO) [20], and Oppositional Teaching Learning Based Method (OTLBO) [2]. Generally, the conventional methods such as Newton Method [3] and LR [4] suffer difficulty when dealing with the problems with nonconvex functions. To overcome the disadvantage, the meta-heuristic algorithms have been proposed based on random search for properly dealing with nonconvex problem with near optimal solution. In [5], the proposed GA method can enhance the quality of optimal solution; however, the GA may suffer from local optimal solution although the penalty factors are set from small to large values. This drawback is tackled by a combination of the augmented Lagrange function with the Lagrange function and penalty terms to update the multiplier (IGA-MU) [8]. As a result, the solution has been significantly improved compared to that from GA in [5]. Nevertheless, the IGA-MU method is still slow for obtaining the optimal solution. IACSA is a quick search algorithm; however, it tends to obtain near optimal solution for small-scale and simple problems and suffers the same drawback as that of the IGA-MU method. Similar to conventional EP, the EP in [7] may still cope with slow convergence and local optimum. On the contrary to LR, the LR-SQP can solve nonlinear and more complicated problems. In SARGA, the combination of tournament selection and simulated binary crossover performed on real-coded GA enables this method to quickly obtain optimal solution with low computational burden. Moreover, there also has been a penalty approach without parameters used to properly handle equality and inequality constraints. BCO and AIS have shown to be superior to EP, PSO, and Real Coded GA (RCGA) in terms of high solution quality and execution time. However, effectiveness of the two methods has not been validated in large-scale systems. In addition, AIS can suffer the premature convergence if the application of the aging operator to eliminate the bad antibodies is not successful. The HSA has been considered as a strong search algorithm since it can deal with several discrete optimization problems [21]. In [14], the combination of each of the three search techniques including Latin Hypercube Sampling (LHS), PSO, and Design and Analysis of Computer Experiments (DACE) with MADS algorithm forms MADS-LHS, MADS-PSO, and MADS-DACE methods, where MADS-DACE is superior to the others. The NDS along with successive refinement search techniques have been employed to speed up the computation with low number of iterations and short computational time. In [17], the two proposed rules, i.e.

Constant Step Size (CSS) rule and Square Summable But not Summable (SSBS) rule, have successfully been applied to update Lagrange multiplier. The obtained results from LR-SSMU-CSS and LR-SSMU-SSBS in terms of cost and characteristic rate have indicated that there is no method superior to another. SPSO and TVAC-PSO are two improved versions and have obtained promising results. In [20], Conventional GSO (CGSO) and its improved version have been implemented for a large system with 48 units. As stated in the study, the GSO methods have been superior to the PSO method; however, the actual value of cost and the verification of the operating zone of combined heat and power units based on the reported solution have shown that the evaluation is not reliable. Teaching Learning Based Optimization (TLBO) is also a population-based simple search algorithm with less control parameters [22], high convergence rate, and less execution time [23].

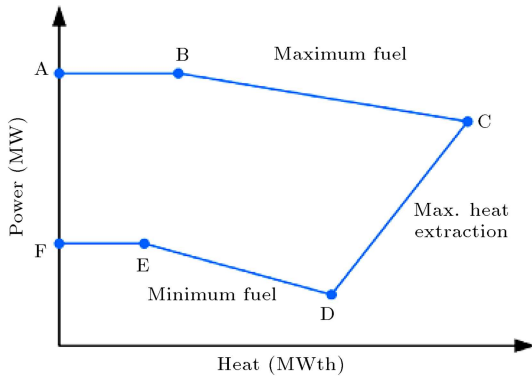
PSO is a population-based optimal search algorithm developed by Kennedy and Eberhart in 1995 [24]. Each individual particle searches in space by adjusting its velocity based on both its own previous best location and its neighbors' best location at each time step. The effectiveness and robustness of the PSO have been demonstrated through several optimization problems in power systems such as reactive power and voltage control [25], power system stabilizer design [26], optimal power flow [27], and short-term hydrothermal scheduling [28,29].

In this paper, eight different improvements in the PSO method are proposed to solve the CHPED problem with valve point loading effects on pure power units and power losses in transmission lines. The proposed improved PSO (IPSO) methods are tested on six systems and the obtained results are compared with those from other methods such as LR [4], GA [5], IACSA [6], EP [7], IGA-MU [8], LR-SQP [9], SARGA [10], BCO [11], HSA [1,12-13], MADS-LHS [14], MADS-PSO [14], MADS-DACE [14], NDS [15], AIS [16], LR-SSMU-CSS [17], LR-SSMU-SSBS [17], TVAC-PSO [19], CGSO [20], and IGSO [20] in terms of cost and execution time.

## 2. Problem formulation

The main task of the problem is to determine the heat and power generations for each unit so that the total cost of the heat and power generations is minimized while the heat demand, power demand, and capacity of each unit are satisfied. In the CHPED, there is a mutual dependency between the power and heat production of combined heat and power units. A typical heat-power feasible region for a combined heat and power unit is given in Figure 1.

The objective of the problem is to minimize



**Figure 1.** Typical heat-power feasible region for combined heat and power units.

the total fuel cost for heat and power production formulated as follows:

$$\text{Min} \left\{ \sum_{i=1}^{N_{PP}} F_{pi}(P_{pi}) + \sum_{j=1}^{N_c} F_{cj}(P_{cj}, H_{cj}) + \sum_{k=1}^{N_{ph}} F_{hk}(H_{hk}) \right\}, \quad (1)$$

where:

$$F_{pi}(P_{pi}) = a_{pi} + b_{pi}P_{pi} + c_{pi}P_{pi}^2, \quad (2)$$

or:

$$F_{pi}(P_{pi}) = a_{pi} + b_{pi}P_{pi} + c_{pi}P_{pi}^2 + \left| e_{pi} \times \sin(f_{pi} \times (P_{pi, \min} - P_{pi})) \right|, \quad (3)$$

and:

$$F_{cj}(P_{cj}, H_{cj}) = a_{cj} + b_{cj}P_{cj} + c_{cj}P_{cj}^2 + k_{cj}H_{cj} + l_{cj}H_{cj}^2 + m_{cj}H_{cj}P_{cj}, \quad (4)$$

$$F_{hk}(H_{hk}) = a_{hk} + b_{hk}H_{hk} + c_{hk}H_{hk}^2, \quad (5)$$

subject to:

- (a) Power balance constraint: The total power output generated by pure power and combined heat and power units must satisfy the power demand:

$$P_D + P_L - \sum_{i=1}^{N_{PP}} P_{pi} - \sum_{j=1}^{N_c} P_{cj} = 0, \quad (6)$$

where the power losses in transmission line are calculated by:

$$P_L = \sum_{i=1}^{N_{PP}+N_c} \sum_{j=1}^{N_{PP}+N_c} P_i B_{ij} P_j + \sum_{i=1}^{N_{PP}+N_c} B_{0i} P_i + B_{00}. \quad (7)$$

- (b) Heat balance constraint: The total heat produced

by pure heat and combined heat and power units must satisfy the heat demand neglecting heat loss:

$$H_D - \sum_{j=1}^{N_c} H_{cj} - \sum_{k=1}^{N_{ph}} H_{hk} = 0. \quad (8)$$

- (c) Generation and heat limits constraints: Each unit must operate within their upper and lower bounds:

$$P_{pi, \min} \leq P_{pi} \leq P_{pi, \max}, \quad (9)$$

$$P_{cj, \min}(H_{cj}) \leq P_{cj} \leq P_{cj, \max}(H_{cj}) \quad (10)$$

$$H_{cj, \min}(P_{cj}) \leq H_{cj} \leq H_{cj, \max}(P_{cj}), \quad (11)$$

$$H_{hk, \min} \leq H_{hk} \leq H_{hk, \max}. \quad (12)$$

Based on Figure 1, the dependent upper and lower power outputs and upper and lower heat outputs of the combined heat and power units are determined as follows:

$$P_{cj}^{\max}(H_{cj}) = \min\{P_{cj}(H_{cj})|_{AB}, P_{cj}(H_{cj})|_{BC}\}, \quad (13)$$

$$P_{cj}^{\min}(H_{cj}) = \max\{P_{cj}(H_{cj})|_{CD}, P_{cj}(H_{cj})|_{DE}\},$$

$$P_{cj}(H_{cj})|_{EF}, \quad (14)$$

$$H_{cj}^{\max}(P_{cj}) = \min\{H_{cj}(P_{cj})|_{BC}, H_{cj}(P_{cj})|_{CD}\}, \quad (15)$$

$$H_{cj}^{\min}(P_{cj}) = 0. \quad (16)$$

### 3. Improved particle swarm optimization methods for CHPED problem

#### 3.1. The proposed improved PSO algorithms

Similar to other meta-heuristic algorithms, PSO algorithm consists of  $N_p$  particles with their position  $X_d$  and velocity  $V_d$ ,  $d = 1, \dots, N_p$ , where each particle  $d$  contains a solution for the problem. The velocity of each particle  $d$  is updated by using the exchange information among its current position and its best previous position as well as its neighbor's best previous position. The best previous position of particle  $d$  and the best previous position of the  $d$ th particle's neighbor are respectively represented by  $Pbest_d$  and  $Nbest_d$ ,  $d = 1, \dots, N_p$ . The new velocity and position of the particle  $d$  are updated as follows:

$$V_d^{\text{new}} = V_d + c_1 \cdot \text{rand} \cdot (Pbest_d - X_d) + c_2 \cdot \text{rand} \cdot (Nbest_d - X_d), \quad (17)$$

$$X_d^{\text{new}} = X_d + V_d, \quad (18)$$

where  $c_1$  and  $c_2$  are acceleration constants;  $\text{rand}$  is random number with uniform distribution between 0 and 1.

The conventional PSO is simple for application on optimization problems. However, the conventional PSO still suffers from some disadvantages such as local optimum and slow convergence for large-scale problems with complex constraints. Therefore, we propose eight different PSO algorithms to improve the solution quality and convergence speed. The improved PSO algorithms are described in details as follow.

### 3.1.1. Global vision of PSO with inertia weight (GWPSO)

By inserting an inertia weight factor into the original PSO, an improved version of PSO is developed by Shi and Eberhart [30]. The suggestion aims to improve the new velocity of each particle meanwhile the way to obtain the new position is retained as in (18). In the paper, the improvement is named global vision of PSO with inertia weight (GWPSO) and the velocity for the GWPSO is as follows [30]:

$$V_i^{\text{new}} = \omega V_i + c_1 \cdot \text{rand.}(Pbest_d - X_d) + c_2 \cdot \text{rand.}(Gbest - X_d), \quad (19)$$

where weight factor,  $\omega$ , is determined as follows [31]:

$$\omega = \omega_{\max} \cdot \frac{\omega_{\max} - \omega_{\min}}{\text{Iter}_{\max}} \cdot \text{Iter}. \quad (20)$$

Obviously, the value of  $\omega$  becomes smaller as the number of iterations increases. This manner enables the IPSO to find out better solution by searching in a smaller local zone.

### 3.1.2. Local vision of PSO with inertia weight (LWPSO)

In this improvement, the velocity of the  $d$ th particle's current iteration is obtained as in Eq. (21). There is a difference between LWPSO and GWPSO as LWPSO uses the best previous position between its two other neighbors  $Lbest_{d-1}$  and  $Lbest_{d+1}$ , and itself  $Lbest_d$ .  $Lbest$  is the best one among  $Lbest_{d-1}$ ,  $Lbest_d$ , and  $Lbest_{d+1}$  of the  $(d-1)$ th particle,  $d$ th particle, and  $(d+1)$ th particle.

$$V_i^{\text{new}} = \omega V_i + c_1 \cdot \text{rand.}(Pbest_d - X_d) + c_2 \cdot \text{rand.}(Lbest_d - X_d). \quad (21)$$

### 3.1.3. Global vision of PSO with constriction factor (GCP SO)

Clerc [32] indicates the usefulness of constriction factor for improving the convergence characteristic of PSO; Eberhart and Shi have applied this parameter to PSO to improve the velocity of particles as follows [31]:

$$V_i^{\text{new}} = K \left( V_i + c_1 \cdot \text{rand.}(Pbest_d - X_d) + c_2 \cdot \text{rand.}(Gbest - X_d) \right), \quad (22)$$

$$K = 2 / \left| 2 - \varphi - \sqrt{\varphi^2 - 4\varphi} \right|, \quad (23)$$

$$\varphi = c_1 + c_2. \quad (24)$$

This improvement with the constriction factor is called global vision of PSO with constriction factor (GCP SO).

### 3.1.4. Local vision of PSO with constriction factor (LCPSO)

By using the information from the best previous position of the two other neighbors and constriction factor, the velocity of the LCPSO is updated as follows:

$$V_i^{\text{new}} = K \left( V_i + c_1 \cdot \text{rand.}(Pbest_d - X_d) + c_2 \cdot \text{rand.}(Lbest_d - X_d) \right). \quad (25)$$

### 3.1.5. Global vision of PSO with inertia weight and Cauchy distribution (GWPSO-CD)

In this version, the new IPSO is proposed by modifying a random number of the above GWPSOs. The random number is replaced with another number ranging in  $[0,1]$  generated by Cauchy distribution as follows:

$$dc = |\tan((\pi/4 \times (\text{rand} - 0.5)))|, \quad (26)$$

and the velocity is updated by:

$$V_i^{\text{new}} = \omega V_i + c_1 \cdot dc \cdot (Pbest_d - X_d) + c_2 \cdot \text{rand.}(Gbest - X_d). \quad (27)$$

### 3.1.6. Local vision of PSO with inertia weight and Cauchy distribution (LWPSO-CD)

By combining LWPSO and Cauchy distribution, the velocity of this version is updated by:

$$V_i^{\text{new}} = \omega V_i + c_1 \cdot dc \cdot (Pbest_d - X_d) + c_2 \cdot \text{rand.}(Lbest_d - X_d). \quad (28)$$

### 3.1.7. Global vision of PSO with constriction factor and Cauchy distribution (GCP SO-CD)

This version also uses Cauchy distribution to generate random number for GCP SO. The velocity is updated as follows:

$$V_i^{\text{new}} = K \left( V_i + c_1 \cdot dc \cdot (Pbest_d - X_d) + c_2 \cdot \text{rand.}(Gbest - X_d) \right). \quad (29)$$

### 3.1.8. Local vision of PSO with constriction factor and Cauchy distribution (LCPSO-CD)

Similar to GCP SO-CD, the Cauchy distribution is also used and the velocity is updated as follows:

$$V_i^{\text{new}} = K \left( V_i + c_1 \cdot dc \cdot (Pbest_d - X_d) + c_2 \cdot \text{rand.}(Lbest_d - X_d) \right). \quad (30)$$

### 3.2. Calculation of slack variables

In this research, two slack variables are used to satisfy the power balance constraint (6) and heat balance constraint (8), including the slack pure power unit 1,  $P_{p1}$ , and the slack pure heat unit 1,  $H_{h1}$ .

#### 3.2.1. Calculation of the slack pure power unit

Suppose that the power output of all units is known excluding  $P_{p1}$ ; therefore, the power output of the unit is calculated using Eq. (6) as follows:

$$P_{p1} = P_D + P_L - \sum_{i=2}^{N_{PP}} P_{pi} - \sum_{j=1}^{N_c} P_{cj}. \quad (31)$$

#### 3.2.2. Calculation of the slack pure heat unit

By using Eq. (8), the slack pure heat unit 1 is obtained as follows:

$$H_{h1} = H_D - \sum_{j=1}^{N_c} H_{cj} - \sum_{k=2}^{N_{ph}} H_{hk}. \quad (32)$$

### 3.3. Implementation of the proposed IPSO algorithms

#### 3.3.1. Initialization

Similar to other meta-heuristic algorithms, the proposed IPSO methods have  $N_P$  particles, where each particle consists of velocity and position. The position of the  $d$ th particle  $X_d$  ( $d = 1, \dots, N_P$ ) contains the power outputs of  $(N_{pp} - 1)$  pure power units and  $N_c$  combined heat and power units and the heat outputs of  $N_c$  combined heat and power units and  $(N_{ph} - 1)$  pure heat units. The details of the initialization are as follows:

$$P_{pi,d} = P_{pi,\min} + \text{rand}^*(P_{pi,\max} - P_{pi,\min}), \quad i = 2, \dots, N_{pp}, \quad (33)$$

$$P_{cj,d} = P_{cj,\min} + \text{rand}^*(P_{cj,\max} - P_{cj,\min}), \quad j = 1, \dots, N_c, \quad (34)$$

$$H_{cj,d} = H_{cj,\min} + \text{rand}^*(H_{cj,\max} - H_{cj,\min}), \quad j = 1, \dots, N_c, \quad (35)$$

$$H_{hk,d} = H_{hk,\min} + \text{rand}^*(H_{hk,\max} - H_{hk,\min}), \quad k = 2, \dots, N_{ph}. \quad (36)$$

The velocity of each particle is also initialized as follows:

$$V_d = V_{d,\min} + \text{rand}^*(V_{d,\max} - V_{d,\min}), \quad d = 1, \dots, N_P, \quad (37)$$

where the maximum and minimum velocities are se-

lected by:

$$V_{d,\max} = 0.15^*(X_{d,\max} - X_{d,\min}), \quad (38)$$

$$V_{d,\min} = -V_{d,\max}. \quad (39)$$

Based on the initialized population of the particles, the fitness function to be minimized corresponding to each nest for the considered problem is calculated:

$$\begin{aligned} FT(X_d) = & \sum_{i=1}^{N_{pp}} F_{Pi}(P_{pi,d}) + \sum_{j=1}^{N_c} F_{cj}(P_{cj,d}, H_{cj,d}) \\ & + \sum_{k=1}^{N_{ph}} F_{hk}(H_{hk,d}) + K_P(P_{p1,d} - P_{p1}^{\lim})^2 \\ & + K_h(H_{h1,d} - H_{h1}^{\lim})^2, \end{aligned} \quad (40)$$

where  $K_P$  and  $K_h$  are penalty factors;  $P_{p1,d}$  and  $H_{h1,d}$  are the power output and heat output of the slack pure power unit and the slack pure heat unit, respectively.

The limitations in Eq. (40) need to be determined as follows:

$$P_{p1}^{\lim} = \begin{cases} P_{p1,\max} & \text{if } P_{p1,d} > P_{p1,\max} \\ P_{p1,\min} & \text{if } P_{p1,d} < P_{p1,\min} \\ P_{p1,d} & \text{otherwise} \end{cases} \quad (41)$$

$$H_{h1}^{\lim} = \begin{cases} H_{h1,\max} & \text{if } H_{h1,d} > H_{h1,\max} \\ H_{h1,\min} & \text{if } H_{h1,d} < H_{h1,\min} \\ H_{h1,d} & \text{otherwise} \end{cases} \quad (42)$$

Each initialized position of particles is set to the best particle  $Pbest_d$  ( $d = 1, \dots, N_P$ ) while the particle with the lowest value of fitness function is set to the best global particle  $Gbest$ . In addition, each particle  $d$  with the best local particle  $Lbest_d$  needs to be found as described in Section 3.1.2.

#### 3.3.2. Updating new velocity and new position for each particle $d$

In the IPSO, the velocity is first updated by eight ways as described in Section 3.1 and the new position is then determined using Eq. (18). During the process, each new velocity and position cannot always satisfy their limits and the following definitions are useful to fix them:

$$V_d = \begin{cases} V_{d,\max} & \text{if } V_d > V_{d,\max} \\ V_{d,\min} & \text{if } V_d < V_{d,\min}; \quad d = 1, \dots, N_P \\ V_d & \text{otherwise} \end{cases} \quad (43)$$

$$P_{pi,d} = \begin{cases} P_{pi,\max} & \text{if } P_{pi,d} > P_{pi,\max} \\ P_{pi,\min} & \text{if } P_{pi,d} < P_{pi,\min}; \quad i = 2, \dots, N_{pp} \\ P_{pi,d} & \text{otherwise} \end{cases} \quad (44)$$

$$P_{cj,d}(H_{cj,d}) = \begin{cases} P_{cj,\max}(H_{cj,d}) & \text{if} \\ P_{cj,d}(H_{cj,d}) > P_{cj,\max}(H_{cj,d}) \\ P_{cj,\min}(H_{cj,d}) & \text{if} \\ P_{cj,d}(H_{cj,d}) < P_{cj,\min}(H_{cj,d}); & (45) \\ j = 1, \dots, N_c \\ P_{cj,d}(H_{cj,d}) & \text{otherwise} \end{cases}$$

$$H_{cj,d}(P_{cj,d}) = \begin{cases} H_{cj,\max}(P_{cj,d}) & \text{if} \\ H_{cj,d}(P_{cj,d}) > H_{cj,\max}(P_{cj,d}) \\ H_{cj,\min}(P_{cj,d}) & \text{if} \\ H_{cj,d}(P_{cj,d}) < H_{cj,\min}(P_{cj,d}); & (46) \\ j = 1, \dots, N_c \\ H_{cj,d}(P_{cj,d}) & \text{otherwise} \end{cases}$$

$$H_{hk,d} = \begin{cases} H_{hk,\max} & \text{if } H_{hk,d} > H_{hk,\max} \\ H_{hk,\min} & \text{if } H_{hk,d} < H_{hk,\min}; \\ k = 2, \dots, N_{ph} \\ H_{hk,d} & \text{otherwise} \end{cases} \quad (47)$$

The power output of the slack pure power unit and heat output of the slack pure heat unit are then obtained as in Section 3.2.

**3.3.3. Overall Procedure of IPSO for CHPED problem**  
The overall procedure of IPSO for solving the CHPED problem is described as follows:

**Step 1:** Select parameters for IPSO, including the number of particle,  $N_P$ , the maximum number of iterations,  $Iter_{\max}$ , and other ones such as  $\omega_{\min}$ ,  $\omega_{\max}$ ,  $c_1$ , and  $c_2$  for each version of IPSO;

**Step 2:** Initialize  $N_P$  particles randomly for power output and heat output by using Eqs. (33)-(36);

**Step 3:** Calculate the slack pure power unit and the slack pure heat unit using Eqs. (31) and (32). Set the current iteration  $Iter = 1$ ;

**Step 4:** Evaluate fitness function for each particle using Eq. (40) to determine  $Pbest_d$ ,  $Gbest$ , and  $Lbest$ ;

**Step 5:** Update new velocity and new position for each particle as in Section 3.1;

**Step 6:**

- Define the new velocity and position for each particle in case of violating their limits using Eqs. (43)-(47);
- Calculate the slack pure power unit and the slack pure heat unit;

**Step 7:** If  $Iter < Iter_{\max}$ , set  $Iter = Iter + 1$  and return to Step 4;

**Step 8:** Calculate the fitness function for each particle using Eq. (40) to determine the best particle,  $Gbest$ , and stop.

## 4. Numerical results

The proposed IPSO algorithms have been tested on six systems consisting of three systems with quadratic fuel cost function of pure power units and three systems with valve point effects on pure power units. Each proposed IPSO algorithm is coded in MATLAB platform and twenty independent trials for each case of convex systems and fifty independent trials for each case of nonconvex systems are run on a 1.8 GHz PC with 4GB of RAM.

### 4.1. Selection of control parameters

Each meta-heuristic algorithm possesses a set of control parameters which can be classified into two control parameter groups including basic control parameter group and advanced control parameter group. The former often includes the number of population and the maximum number of iterations. These parameters can easily be found based on the fact that the higher is set the number, the better solution and the longer simulation time are obtained and vice versa. The latter depends on each individual algorithm and section of these parameters is difficult since there is no rule to choose them.

In the eight proposed IPSO algorithms, there are two parameters in the basic group consisting of the number of particles and the maximum number of iterations and other ones in the advanced group comprising inertia weight and constriction factor. The first group has a significant impact on the solution quality and execution time. Obviously, a large maximum number of iterations will certainly lead to time consuming and better optimal solution whereas the lower value will achieve fast simulation time and worse solution. On the other hand, the population is also dependent on the scale of considered systems, such as the complexity of constraints and objective function. Therefore, the number of individuals in population is selected based on experience. In order to implement the IPSO methods for the CHPED problem, each particle contains pure power units and pure heat units except pure power unit 1 and pure heat unit 1, and all combined heat and power units. For initialization, each particle is randomly generated within its limitations and then the slack pure power unit 1 and slack pure heat unit 1 are determined based on the power balance equation and heat balance equation. If the power losses and heat losses are neglected, the power slack unit and heat slack unit of the population are determined at once. On the contrary, the two slack units of each particle are calculated one by one since the power losses and heat losses are taken into account. This manner leads to time consumption for systems with power losses. Consequently, the population should be set to high value for systems neglecting power losses meanwhile

fewer particles of population should be chosen for systems considering power losses. By experiment, the population is set to 50 for all test systems neglecting transmission losses in Section 4.3, 10 for other systems in Section 4.4, and 20 for 48-unit system 6.

In contrast to the former, the latter influences optimal solution quality, but not the execution time. Furthermore, the selection of these parameters is difficult since there is no rule to choose them. By experiment, the inertia weight and constriction factor are determined based on  $\omega_{\max} = 0.9$ ,  $\omega_{\min} = 0.1$ ,  $c_1 = 2.05$ , and  $c_2 = 2.05$  for all test systems except system 6.

#### 4.2. Benchmark optimization function

In this section, the proposed IPSO methods are tested on several benchmark functions [33] and given in Table 1. For implementation of the IPSO methods,

$N_P$  is set to 20 whereas  $Iter_{\max}$  is set to 2,000 for all functions, except function 1 with 150,000. Note that all methods in [34] including GA, PSO, and GSO have set the population and the maximum iterations to 48 individuals and 150,000 iterations, respectively, for all functions. There is no information about the population and iteration reported in [33]. For each case, each IPSO method is run in 100 independent

trials. The comparisons between IPSO methods and others including RCGA-RTVM (RCGA with random transfer vectors-based mutation) [33], GA, PSO, and GSO [34] are shown in Tables 2 to 6.

As observed from Tables 4–6, the IPSO methods obtain better minimum and average values than RCGA-RTVM [33] for functions F3, F4, and F5 except for the average value of function F5. As compared to GA, PSO, and GSO, the IPSO methods are more effective and robust since the average values and standard deviation values from the IPSO methods are less than those from these methods for functions F1 and F2. Furthermore, like the functions F1 and F2, the IPSO methods are also more efficient than RCGA-RTVM [33] due to less average and standard deviation values. Consequently, it can be concluded that the IPSO methods are very efficient for the benchmark functions as compared to other methods.

#### 4.3. Systems with quadratic fuel cost function of pure power units

In this section, three systems with quadratic fuel cost function of pure power units neglecting power losses in transmission lines are considered, including one system with four units and two other systems with five units.

Table 1. Five benchmark functions.

Function	Definition	Domain
F1 (Rosenbrock)	$\sum_{i=1}^N [(100(x_{i+1} - x_i^2))^2 + (x_i - 1)^2]$	$x_i \in [-30, 30]; N = 30$
F2	$4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	$x_i \in [-5, 5]; N = 2$
F3 (Sphere)	$\sum_{i=1}^n x_i^2$	$x_i \in [-5.12, 5.12]; N = 25$
F4 (Rosenbrock)	$\sum_{i=1}^{N-1} [(100(x_{i+1} - x_i^2))^2 + (x_i - 1)^2]$	$x_i \in [-5.12, 5.12]; N = 25$
F5 (Griewangk)	$1 + \frac{1}{4000} \sum_{i=1}^N x_i^2 - \prod_{i=1}^N \cos(\frac{x_i}{\sqrt{i}})$	$x_i \in [-600, 600]; N = 25$

Table 2. Comparisons for benchmark function F1.

Method	Min.	Average	Max.	Std. dev.	CPU time (s)
RCGA-RTVM [33]	-	34.4281	-	24.5366	-
GA [34]	-	338.5516	-	361.497	11.3
PSO [34]	-	37.3582	-	32.1436	37.6
GSO [34]	-	49.8359	-	30.1771	27.8
LCPSO	0.5674	22.0832	28.9418	6.8405	13.93
LCPSO-CD	9.8838	22.4809	28.4788	5.1825	13.56
LWPSO	6.4779	20.8742	28.8812	6.322	14.13
LWPSO-CD	13.4237	21.5025	28.2256	4.6819	13.07
GCPSO	4.8124	21.3941	28.0665	5.9035	13.04
GCPSO-CD	0.8947	21.2693	24.9435	5.2577	13.19
GWPSO	2.8233	21.9946	28.4215	6.9495	13.69
GWPSO-CD	0.8113	23.3905	29.391	6.0809	13.00

**Table 3.** Comparisons for benchmark function F2.

Method	Min.	Average	Max.	Std. dev.	CPU time (s)
RCGA-RTVM [33]	-	-1.03162845	-	2.8796 E-11	-
GA [34]	-	-1.0298	-	3.1314E-3	0.5
PSO [34]	-	-1.0160	-	1.2786 E-2	0.3
GSO [34]	-	-1.031628	-	0	0.2
LCPSO	-1.031628453	-1.03162845	-1.03162845	0	0.15
LCPSO-CD	-1.031628453	-1.03162845	-1.03162845	0	0.16
LWPSO	-1.031628453	-1.03162845	-1.03162845	0	0.16
LWPSO-CD	-1.031628453	-1.03162845	-1.03162845	0	0.17
GCPSO	-1.031628453	-1.03162845	-1.03162845	0	0.15
GCPSO-CD	-1.031628453	-1.03162845	-1.03162845	0	0.15
GWPSO	-1.031628453	-1.03162845	-1.03162845	0	0.16
GWPSO-CD	-1.031628453	-1.03162845	-1.03162845	0	0.15

**Table 4.** Comparisons for benchmark function F3.

Method	Min.	Average	Max.	Std. dev.	CPU time (s)
RCGA-RTVM [33]	1.4038E-24	4.0738E-22	-	1.1958 E-21	-
LCPSO	2.76005E-32	1.444E-29	1.58113E-28	3.48271E-29	0.27
LCPSO-CD	1.65718E-32	2.896E-28	5.65971E-27	1.23E-27	0.31
LWPSO	7.53991E-28	1.511E-25	1.72898E-24	3.83299E-25	0.28
LWPSO-CD	1.57528E-36	1.985E-33	1.15549E-32	3.24783E-33	0.32
GCPSO	3.09053E-37	5.683E-30	8.36926E-29	1.82112E-29	0.13
GCPSO-CD	6.14382E-35	7.052E-25	1.4097E-23	3.0723E-24	0.16
GWPSO	1.78587E-33	3.545E-24	6.1817E-23	1.34102E-23	0.13
GWPSO-CD	3.45608E-33	1.478E-19	1.15129E-18	3.4534E-19	0.17

**Table 5.** Comparisons for benchmark function F4.

Method	Min.	Average	Max.	Std. dev.	CPU time (s)
RCGA-RTVM [33]	22.504	24.094	-	0.6804	-
LCPSO	7.9294	22.1923	24.6959	2.4808	0.29
LCPSO-CD	4.0231	22.2021	24.6947	3.065	0.32
LWPSO	7.3	46.8	1237.9	170.2	0.3
LWPSO-CD	6.4955	22.1	24.6957	3.3924	0.29
GCPSO	9.7564	21.7	24.7	2.66	0.13
GCPSO-CD	5.8902	22	24.69	2.89	0.16
GWPSO	6.4	17.187	27.988	6.233	0.17
GWPSO-CD	9.695	17.902	26.496	4.932	0.18

#### 4.3.1. System 1 with 4 units

The test system from [4] includes one pure power unit, two combined heat and power units, and one pure heat unit. There are three load demands for this system including: 1) power demand of 200 MW and heat demand of 115 MWth, 2) power demand of 175 MW and heat demand of 110 MWth, and 3) power demand of 225 MW and heat demand of 125 MWth.

For implementation, the maximum number of

iterations is set to 300 and the others are mentioned in Section 4.1. The obtained results are compared to those of other methods in Tables 7-9. For load demand 1 in Table 7, the proposed IPSO algorithms can obtain better total cost than LR [4], GA [5], IACSA [6], MADS-LHS and MADS-PSO [14], EP [7], and IGA-MU [8] and the same total cost with other methods except HSA [12,13] and SPSO [18]. Note that the solution obtained by HSA in [12-13] is infeasible since



**Table 6.** Comparisons for benchmark function F5.

Method	Min.	Average	Max.	Std. dev.	CPU time (s)
RCGA-RTVM [33]	1.4353E-8	1.91 E-8	-	3.92E-9	-
LCPSO	0	0.011	0.066	0.016	0.27
LCPSO-CD	0	0.008	0.029	0.009	0.31
LWPSO	0	0.013	0.09	0.018	0.28
LWPSO-CD	0	0.011	0.049	0.013	0.32
GCPSO	0	0.037	0.191	0.045	0.13
GCPSO-CD	0	0.039	0.337	0.059	0.16
GWPSO	0	0.014	0.093	0.021	0.13
GWPSO-CD	0	0.014	0.079	0.019	0.17

**Table 7.** Result comparison for load demand 1 of system 1.

Method	Min. cost (\$/h)	Avg. cost (\$/h)	Max. cost (\$/h)	Std. dev. (\$/h)	CPU time (s)
HSA [1]	9,257.07	-	-	-	-
LR [4]	9,257.10	-	-	-	-
GA [5]	9,267.20	-	-	-	-
IACSA [6]	9,452.20	-	-	-	-
EP [7]	9,257.10	-	-	-	-
IGA-MU [8]	9,257.08	-	-	-	-
HSA [12]	8606.07*	-	-	-	-
HSA [13]	8606.07*	-	-	-	-
MADS-LHS [14]	9277.1311	9547.9151	10005.381	213.3516	7.0422
MADS-PSO [14]	9301.3567	9551.277	9997.6576	251.1569	7.5594
MADS-DACE [14]	9257.0754	9257.5148	9260.4317	1.0743	2.3078
NDS[15]	9,257.07	9,257.07	9,257.07	-	-
LRSS [17]	9,257.07	-	-	-	-
SPSO [18]	9,248.17**	-	-	-	-
TVAC-PSO [19]	9257.075	-	-	-	1.33
LCPSO	9257.075	9257.095	9257.236	0.038373	0.14
LCPSO-CD	9257.075	9257.075	9257.076	0.000169	0.14
LWPSO	9257.075	9257.09	9257.141	0.019792	0.15
LWPSO-CD	9257.075	9257.077	9257.103	0.005975	0.14
GCPSO	9257.075	9257.075	9257.077	0.000718	0.14
GCPSO-CD	9257.075	9257.075	9257.078	0.000396	0.16
GWPSO	9257.075	9257.079	9257.128	0.011484	0.15
GWPSO-CD	9257.075	9257.077	9257.093	0.004995	0.15

\* The solution violated the feasible operating zone of cogeneration unit 3.

\*\* The power generation is less than power load demand.

the power output of combined heat and power unit 3 is outside the feasible operating zone and the total power generation of the SPSO is less than power load. For computational time, the proposed algorithms are faster than the MADS-LHS, MADS-PSO, MADS-DACE, and TVAC-PSO. There is no execution time reported for other methods.

For the result comparison of load demands 2 and 3 in Tables 8 and 9, the proposed IPSO methods can obtain better total cost than MADS-LHS and MADS-PSO in [14] and the same total cost as that obtained from MADS-DACE [14], LR-SSMU-CSS [17], and LR-SSMU-SSBS [17]. For the computational time, the proposed methods are also vastly faster than other

**Table 8.** Result comparison for load demand 2 of system 1.

Method	Min. cost (\$/h)	Avg. cost (\$/h)	Max. cost (\$/h)	Std. dev. (\$/h)	CPU time (s)
MADS-LHS [14]	8622.0748	8921.7626	9442.1843	252.3315	63.6
MADS-PSO [14]	8629.4156	8799.8773	9046.634	136.8504	56.5656
MADS-DACE [14]	8555.9625	8555.9625	8555.9652	0.0009	3.7094
LR-SSMU-CSS [17]	8555.9625	-	-	-	-
LR-SSMU-SSBS [17]	8555.9625	-	-	-	-
LCPSO	8555.9625	8555.96263	8555.9632	0.000173	0.16
LCPSO-CD	8555.9625	8555.962525	8555.9626	0.000043	0.15
LWPSO	8555.9626	8555.98514	8556.1231	0.040325	0.15
LWPSO-CD	8555.9625	8555.962605	8555.9635	0.000237	0.14
GCPSO	8555.9625	8555.962535	8555.9631	0.000131	0.15
GCPSO-CD	8555.9625	8555.9625	8555.9625	0	0.16
GWPSO	8555.9625	8555.96549	8556.0167	0.011761	0.14
GWPSO-CD	8555.9625	8555.962695	8555.9652	0.000583	0.15

**Table 9.** Result comparison for load demand 3 of system 1

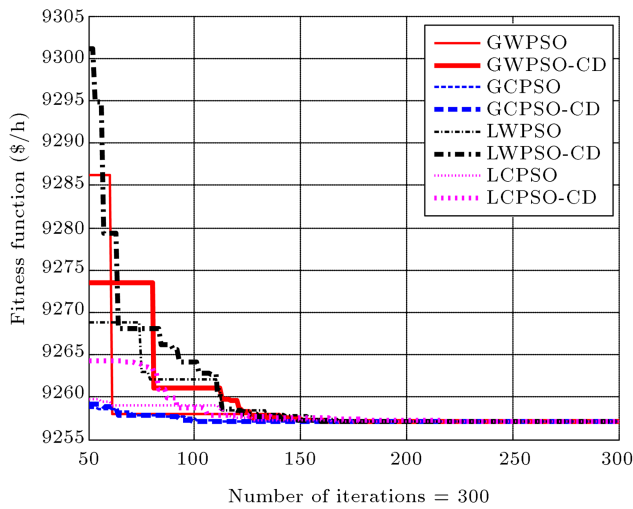
Method	Min. cost (\$/h)	Avg. cost (\$/h)	Max. cost (\$/h)	Std. dev. (\$/h)	CPU time (s)
MADS-LHS [14]	10101.4753	10295.4723	10833.7313	236.1302	8.1516
MADS-PSO [14]	10101.8942	10455.1238	11393.4252	478.2026	7.0563
MADS-DACE [14]	10074.4875	10074.4907	10074.4883	0.0011	5.5281
LR-SSMU-CSS [17]	10,074.49	-	-	-	-
LR-SSMU-SSBS [17]	10,074.49	-	-	-	-
LCPSO	10074.4875	10074.69812	10077.6977	0.69742	0.15
LCPSO-CD	10074.4875	10074.5736	10075.5805	0.23412	0.15
LWPSO	10074.4875	10074.49246	10074.5308	0.01053	0.14
LWPSO-CD	10074.4875	10074.49595	10074.525	0.01261	0.15
GCPSO	10074.4875	10075.73304	10096.1908	4.70334	0.14
GCPSO-CD	10074.4875	10075.03631	10082.3371	1.71600	0.15
GWPSO	10074.4875	10074.58515	10075.7021	0.27089	0.15
GWPSO-CD	10074.4875	10074.57905	10075.2045	0.21077816	0.15

methods. There is no computational time reported for LR-SSMU-CSS and LR-SSMU-SSBS methods.

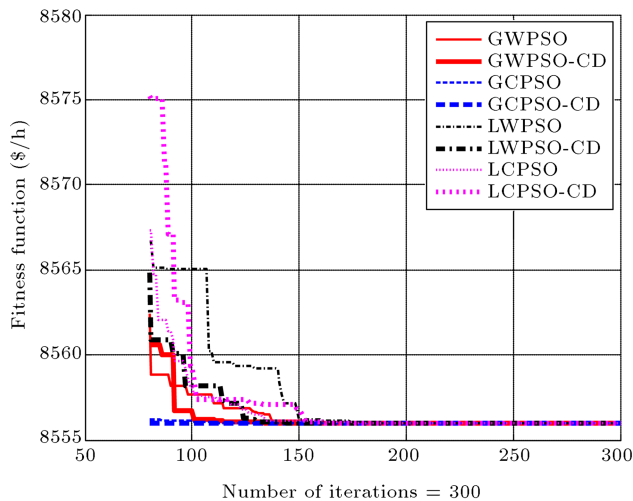
Comparing the obtained results by the proposed IPSO methods, LCPSO-CD and GCPSO-CD are the best two versions for load demand 1 with standard deviation values of 0.000169 and 0.000396 and load demand 2 with the values of 0.000043 and 0, respectively, whereas the best two ones for load demand 3 are LWPSO and LWPSO-CD with the values of 0.01053 and 0.01261, respectively. In addition, as observed from the two different method groups using Cauchy distribution and random distribution, the methods with Cauchy distribution can obtain better standard deviation than those with random distribution, except LWPSO and LWPSO-CD for load demand 2. In terms of convergence speed, the proposed IPSO

can obtain the optimal solution with similar CPU time.

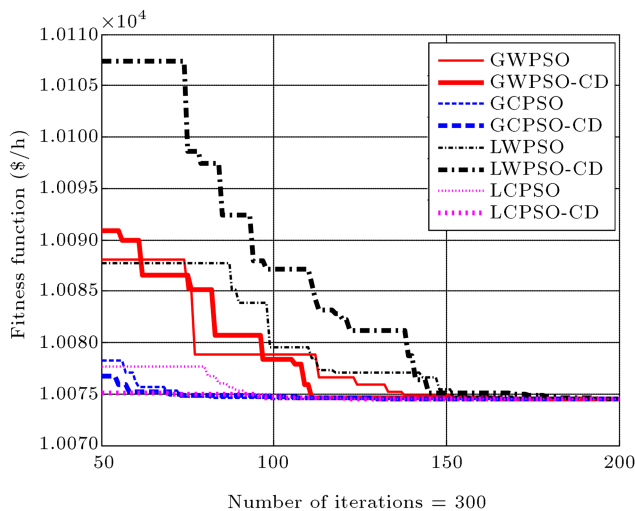
On the other hand, the convergence characteristics of the proposed IPSO methods for test systems have been given for comparison of their performance. Moreover, the proposed IPSO methods are the methods for random search; thus, their convergence characteristic cannot be expressed in mathematics. Therefore, it is useful to include these convergence characteristics with respect to number of iterations. Based on the convergence characteristics of the proposed IPSO methods for each case, the convergence speed of the proposed methods is discussed. For the system, the convergence characteristics of the proposed IPSO methods for three load demands of system 1 are shown in Figures 2-4. As observed from the figures, the



**Figure 2.** Convergence characteristic of the proposed IPSO methods for load demand 1 of system 1.



**Figure 3.** Convergence characteristic of the proposed IPSO methods for load demand 2 of system 1.



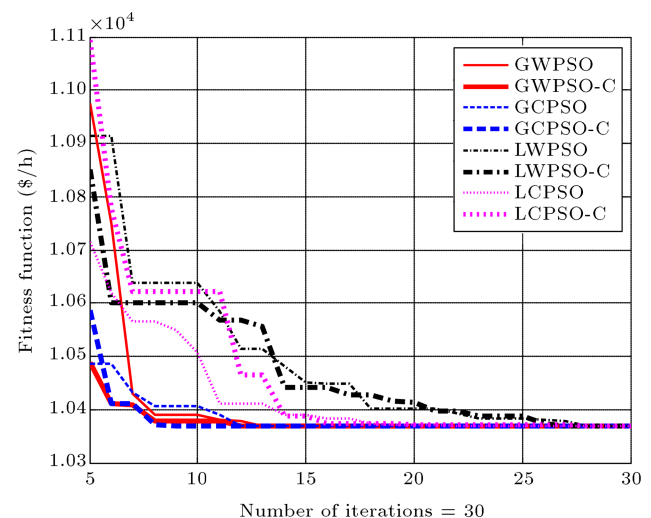
**Figure 4.** Convergence characteristic of the proposed IPSO methods for load demand 3 of system 1.

LWPSO and LWPSO-CD own the worst robustness since the fitness values of them are higher than others at almost all iterations when the search process passes about one fourth of the maximum iterations. On the contrary, the GCP SO and GCP SO-CD are the two best algorithms due to their lowest fitness values at nearly all iterations. In addition, as compared in one algorithm with Cauchy distribution and one algorithm with random distribution, it can be stated that the Cauchy distribution is more useful for the IPSO than random distribution. In fact, almost methods with Cauchy distribution are respectively better than those with random distribution, except for Figure 2 where only GWPSO-CD is worse than GWPSO and Figure 4 where only LWPSO-CD possesses worse convergence characteristic than LWPSO.

#### 4.3.2. Systems 2 and 3 with 5 units

System 2 consists of 4 units of the system 1 in Section 4.3.1 and one additional steam turbine generating unit [9]. The power output of the additional unit is fixed at 80 MW while it always consumes 120 MWth of heat. The power and heat demands are 250 MW and 115 MWth, respectively. For system 3, there are three different load demands including: 1) 160 MW and 220 MWth, 2) 250 MW and 175 MWth, and 3) 300 MW and 150 MWth [1].

For implementation,  $N_P$  is fixed at 30 for system 2, 500 for load demands 1 and 3, and 300 for load demand 2 of system 3. The comparison between the IPSO methods and other methods are reported in Tables 10 and 11. Obviously, the proposed methods can obtain less total cost than LR-SQP for system 2, and GA and HSA for three load demands of system 3. There is no execution time reported for the compared methods in [1,9]. Figure 5 has shown the convergence characteristic by the proposed IPSO



**Figure 5.** Fitness function convergence characteristic of system 2 with 5 units.

**Table 10.** Cost comparison for system 2.

Method	Min. cost (\$/h)	Avg. cost (\$/h)	Max. cost (\$/h)	Std. dev. (\$/h)	CPU time (s)
LR-SQP [9]	10623.4	-	-	-	-
LCPSO	10369.6435	10369.8213	10372.2049	0.556975	0.037
LCPSO-CD	10369.6435	10369.8186	10371.0434	0.332948	0.037
LWPSO	10369.6435	10369.6501	10369.7686	0.027196	0.038
LWPSO-CD	10369.6435	10369.6487	10369.7475	0.022659	0.036
GCPSO	10369.6435	10371.9331	10387.1100	4.639012	0.039
GCPSO-CD	10369.6435	10371.5256	10388.3594	4.712559	0.041
GWPSO	10369.6435	10369.9931	10373.5569	0.901681	0.038
GWPSO-CD	10369.6435	10369.6768	10369.9011	0.067864	0.038

**Table 11.** Cost comparison for system 3.

Method	Load demand 1		Load demand 2		Load demand 3	
	Min. cost (\$/h)	CPU time (s)	Min. cost (\$/h)	CPU time (s)	Min. cost (\$/h)	CPU time (s)
GA [1]	11,837.40	-	12,327.37	-	13,779.50	-
HSA [1]	11,810.88	-	12,284.45	-	13,723.20	-
LCPSO	11758.0692	0.26	12116.6009	0.16	13672.8342	0.25
LCPSO-CD	11758.0629	0.26	12116.6009	0.17	13672.8341	0.26
LWPSO	11758.0717	0.27	12116.6009	0.16	13672.8341	0.26
LWPSO-CD	11758.0672	0.26	12116.6008	0.16	13672.8342	0.26
GCPSO	11758.0702	0.27	12116.6008	0.17	13672.8342	0.25
GCPSO-CD	11758.0702	0.26	12116.6008	0.17	13672.837	0.25
GWPSO	11758.0844	0.27	12116.6008	0.16	13672.8342	0.26
GWPSO-CD	11758.0844	0.28	12116.6008	0.16	13672.8341	0.26

methods for system 2. Clearly, the IPSO method with constriction factor, global vision, and Cauchy distribution is the fastest method obtaining the optimal solution and the IPSO methods with Cauchy distribution are faster than the IPSO methods with random distribution.

#### 4.4. Systems with nonconvex fuel cost function of pure power units

##### 4.4.1. System 4 with 7 units

The test system consisting of four pure power units, two combined heat and power units, and one pure heat unit with nonconvex fuel cost function of pure power units and power losses in transmission lines is considered. The power and heat demands of the system are 600 MW and 150 MWth, respectively. The data for the test system is from [11,16]. For implementation of the proposed IPSO methods for the system, the maximum number of iterations is set to 2000. The minimum total cost and average execution time of the proposed methods are compared with those of BCO and RCGA in [11]; AIS, EP, and PSO in [16]; and TVAC-PSO in [19] as given in Table 12. As seen from the table,

the TVAC-PSO [19] has reported the lowest fuel cost; but the power loss from the method, 0.73922 MW, is much less than the exact value, 7.3922 MW, which is recalculated. Therefore, the TVAC-PSO is not valid to compare the performance of the case. As compared to other valid methods [11,16], the proposed IPSO methods are more efficient and faster due to obtaining less total cost with shorter computational time. Note that all methods in [11,16] were run on a P-IV, 80 GB, and 3.0 GHz personal computer. Therefore, the proposed IPSO methods are very favourable for solving CHPED problem with nonconvex fuel cost function of pure power units.

Comparing the minimum cost and standard deviation of methods with Cauchy distribution and those of methods with random distribution has indicated that the methods which use random distribution can obtain better solution than those which use Cauchy distribution.

##### 4.4.2. System 5 with 11 units

For demonstration of the practical applicability of the proposed IPSO methods, a larger-scale system

**Table 12.** Result comparison for system 4

Method	Min. cost (\$/h)	Avg. cost (\$/h)	Max. cost (\$/h)	Std. dev. (\$/h)	CPU time (s)
AIS [16]	10,355	-	-	-	5.2956
EP [16]	10,390	-	-	-	5.3274
PSO [16]	10,613	-	-	-	5.3944
BCO [11]	10,317	-	-	-	5.1563
RCGA[11]	10,667	-	-	-	6.4723
TVAC-PSO [19]	10,100.32*	-	-	-	3.25
LCPSO	10199.5458	11104.52	13564.26	921.2788	1.02
LCPSO-CD	10279.1023	11366.53	15327.00	1196.669	1.02
LWPSO	10194.0217	11077.21	14304.68	815.3887	1.03
LWPSO-CD	10265.6716	11199.34	15394.56	1219.225	1.02
GCPSO	10143.7898	11605.07	14329.09	1162.164	1.01
GCPSO-CD	10294.2369	11678.77	14808.76	1172.982	1.02
GWPSO	10243.0249	11235.88	15265.87	887.0105	1.03
GWPSO-CD	10281.6835	11476.73	15318.27	1045.916	1.03

\* The reported power loss is much less than the calculated value.

**Table 13.** Result summary of system 5 with nonconvex fuel cost function of pure power units.

Method	Min. cost (\$/h)	Avg. cost (\$/h)	Max. cost (\$/h)	Std. dev. (\$/h)	CPU time (s)
LCPSO	39150.602	45112.8588	53277.5633	3101.651	1.6
LCPSO-CD	39095.767	44333.4244	52017.7678	3582.438	1.7
LWPSO	38800.942	44079.3646	53197.1563	3350.124	1.6
LWPSO-CD	39003.38	44016.9921	49711.6398	2923.853	1.8
GCPSO	39145.78	45069.8953	51697.5972	3115.485	1.8
GCPSO-CD	38924.238	44838.2489	54764.3813	3243.684	1.7
GWPSO	39292.533	44791.2589	53106.0312	3004.722	1.7
GWPSO-CD	39800.802	44367.8227	52079.8046	2786.386	1.8

including eight pure power units with nonconvex fuel cost function, two combined heat and power units, and one pure heat unit where the first four pure power units, two combined heat and power units, and one pure heat unit are from system 4 is considered in this case. The load and heat demands for this system are 800 MW and 1000 MWth, respectively. The data of pure power units 5 to 8 and power loss coefficients for this system are given in Appendix. The obtained result by the proposed methods given in Table 13 has shown that LWPSO obtained the best minimum cost of \$38800.942, whereas GWPSO-CD obtained the highest minimum cost of \$39800.802. Among the methods using random distribution and Cauchy distribution, there is no obvious distinction for which group is more effective, because LCPSO-CD and GCPSO-CD respectively obtain better minimum total costs than those of LCPSO and GCPSO, but LWPSO-CD and GWPSO-CD respectively obtain higher minimum costs than those of LWPSO and GWPSO. On the contrary to

the obtained minimum costs, LCPSO-CD and GCPSO-CD respectively obtain higher standard deviation costs than those of LCPSO and GCPSO, but LWPSO-CD and GWPSO-CD respectively obtain less standard deviation costs than those of LWPSO and GWPSO. Obviously, in the comparison of the proposed methods using random distribution and Cauchy distribution, there is a conflict between minimum cost and standard deviation cost.

#### 4.4.3. System 6 with 48 units

In this section, the proposed PSO methods have also been tested on a large-scale system with 48 units in [20] consisting of 26 pure power units, 12 combined heat and power units, and 10 pure heat units.

For implementation of the proposed methods, the number of particles and the maximum number of iterations are respectively set to 20 and 80,000. In addition, other control parameters for each of the proposed improved PSO methods are listed below:

**Table 14.** The result obtained by the proposed IPSO methods for system 6 with 48 units.

Method	Min. cost (\$/h)	Avg. cost (\$/h)	Max. cost (\$/h)	Std. dev. (\$/h)	CPU time (s)
TVAC-PSO [19]	117824.896*	-	-	-	89.63
CGSO [20]	113131.83**	-	-	-	70.65
IGSO [20]	112320.4159*	-	-	-	70.65
LCPSO	117248.7934	120141.2284	123113.8461	1833.3854	72.4
LCPSO-CD	117164.1170	120461.7189	127241.9529	2951.5578	69.8
LWPSO	117088.3912	118259.5065	119937.0786	592.4360	71.6
LWPSO-CD	116859.8498	118042.9320	120121.4666	897.2295	73.7
GCPSO	117374.0338	121309.6538	127653.3057	2653.4103	74.5
GCPSO-CD	117537.4750	120717.7308	125337.1095	1832.3725	68.7
GWPSO	116572.7024	119151.3075	126173.5174	1961.0783	65.7
GWPSO-CD	116768.1356	120171.6064	122708.8358	1579.9217	73.4

\* The exact cost is much higher than the reported one.

\*\* Combined heat and power unit 36 is outside the feasible operating zone.

- LCPSO, LCPSO-CD, GCPSO, GCPSO-CD:  
 $c_1 = 2.05$ ,  $c_2 = 2.05$
- LWPSO:  $\omega_{\max} = 0.8$ ,  $\omega_{\min} = 0.2$
- LWPSO-CD:  $\omega_{\max} = 0.9$ ,  $\omega_{\min} = 0.2$
- GWPSO:  $\omega_{\max} = 0.9$ ,  $\omega_{\min} = 0.1$
- GWPSO-CD:  $\omega_{\max} = 0.8$ ,  $\omega_{\min} = 0.3$

The detailed results obtained by the improved PSO methods are compared to those from other methods in Table 14. Clearly, the total cost from CGSO and IGSO in [20] is better than that from the other methods. However, the actual total costs calculated based on the provided solutions from TVAC-PSO [19] and IGSO [20] are respectively 118962.5402 \$/h and 117377.8159 \$/h, which are worse than the reported total costs in these papers. The reported cost and recalculated cost from CGSO [20] are identical, but the combined heat and power unit 36 is outside the feasible operating zone. Therefore, the CGSO is not valid for comparison. In terms of the computation time comparison, the proposed IPSO algorithms are faster than TVAC-PSO and approximate to CGSO and IGSO. Note that TVAC-PSO, and GSO and IGSO have been run on computers with 2.0 GHz and 2.5 GHz, respectively.

## 5. Conclusions

In this paper, eight IPSO methods have been proposed for solving the CHPED problem considering nonconvex fuel cost function and power losses in transmission lines. The proposed IPSO methods have been developed based on several factors such as inertia weight and constriction factor and distributions such as random distribution and Cauchy distribution to improve the

search ability and speed up the computation of PSO. The proposed IPSO methods have been tested on three systems with quadratic objective function and three systems with nonconvex objective function of pure power generation units, and the obtained results have indicated that the proposed IPSO methods are more effective and robust than many other methods in the literature. Moreover, the IPSO methods using Cauchy distribution are more efficient than the ones using random distribution for the systems with quadratic function, whereas the proposed IPSO methods with random distribution are more efficient than the ones using Cauchy distribution for the systems with nonconvex fuel cost function. Therefore, the proposed IPSO methods are very favourable for solving the nonconvex CHPED problem.

## Nomenclature

$F_{pi}, F_{hk}$	Cost function of pure power unit $i$ and pure heat unit $k$
$F_{cj}$	Cost function of combined heat and power unit $j$
$a_{pi}, b_{pi}, c_{pi}$	Cost function coefficients of pure power unit $i$
$e_{pi}, f_{pi}$	Nonconvex cost function coefficients of pure power unit $i$
$a_{cj}, b_{cj}, c_{cj}, k_{cj}, l_{cj}, m_{cj}$	Cost function coefficients of combined heat and power unit $j$
$a_{hk}, b_{hk}, c_{hk}$	Cost function coefficients of pure heat unit $k$
$N_{pp}, N_c, N_{ph}$	Number of pure power units, combined heat and power units, and pure heat units

$P_{pi,d}, P_{cj,d}$	Power output of pure power unit $i$ and combined heat and power unit $j$ corresponding to particle $d$
$H_{hk,d}, H_{cj,d}$	Heat output of pure heat unit $k$ and combined heat and power unit $j$ corresponding to particle $d$
$P_{pi,min}, P_{pi,max}$	Lower and upper limits of power output for pure power unit $i$
$P_{cj,min}, P_{cj,max}$	Lower and upper limits of power output for combined heat and power unit $j$
$H_{hk,min}, H_{hk,max}$	Lower and upper limits of heat output for pure heat unit $k$
$H_{cj,min}, H_{cj,max}$	Lower and upper limits of heat output for combined heat and power unit $j$
$P_L$	Power loss on the transmission line
$B_{ij}, B_{0i}, B_{00}$	Power loss coefficients
$P_D, H_D$	Power and heat load demands.

## References

- Vasebi, A., Fesanghary, M. and Bathaee, S.M.T. "Combined heat and power economic dispatch by harmony search algorithm", *Electrical Power and Energy Systems*, **29**, pp. 713-719 (2007).
- Provas, K.R., Chandan, P. and Sneha, S. "Oppositional teaching learning based optimization approach for combined heat and power dispatch", *Electrical Power and Energy Systems*, **57**, pp. 392-403 (2014).
- Rooijers, F.J. and Van Amerongen, R.A.M. "Static economic dispatch for co-generation systems", *IEEE Trans Power Syst*, **3**(9), pp. 1392-8 (1994).
- Tao, G., Henwood, M.I. and Van, O.M. "An algorithm for heat and power dispatch", *IEEE Trans Power Syst.*, **11**(4), pp. 1778-84 (1996).
- Song, Y.H. and Xuan, Y.Q. "Combined heat and power economic dispatch using genetic algorithm based penalty function method", *Electric Mach. Power Systems*, **26**(4), pp. 363-372 (1998).
- Song, Y.H., Chou, C.S. and Stonham, T.J. "Combined heat and power dispatch by improved ant colony search algorithm", *Electric Power Systems Research*, **52**, pp. 115-121 (1999).
- Wong, K.P. and Algie, C. "Evolutionary programming approach for combined heat and power dispatch", *Electric Power Systems Research*, **61**, pp. 227-232 (2002).
- Su, C.T. and Chiang, C.L. "An incorporated algorithm for combined heat and power economic dispatch", *Electric Power Systems Research*, **69**, pp. 187-195 (2004).
- Chapa, G. and Galaz, V. "An economic dispatch algorithm for cogeneration systems", *Proc. IEEE Power Engineering Society General Meeting*, **1**, pp. 989-994 (2004).
- Subbaraj, P., Rengaraj, R. and Salivahanan, S. "Enhancement of combined heat and power economic dispatch using self-adaptive real-coded genetic algorithm", *Applied Energy*, **86**, pp. 915-921 (2009).
- Basu, M. "Bee colony optimization for combined heat and power economic dispatch", *Expert Systems with Applications*, **38**, pp. 13527-13531 (2011).
- Esmale, K. and Majid, J. "Harmony search algorithm for solving combined heat and power economic dispatch problems", *Energy Conversion and Management*, **52**, pp. 1550-1554 (2011).
- Javadi, M.S., Esmael, N.A. and Sabramooz, S. "Economic heat and power dispatch in modern power system harmony search algorithm versus analytical solution", *Scientia Iranica D*, **19**(6), pp. 1820-1828 (2012).
- Hosseini, S.S.S., Jafarnejad, A., Behrooz, A.H. and Gandomi, A.H. "Combined heat and power economic dispatch by mesh adaptive direct search algorithm", *Expert Syst Appl.*, **38**, pp. 6556-64 (2011).
- Chen, C.L., Lee, T.Y., Jan, R.M. and Lu, C.L. "A novel direct search approach for combined heat and power dispatch", *Electrical Power and Energy Systems*, **43**, pp. 766-773 (2012).
- Basu, M. "Artificial immune system for combined heat and power economic dispatch", *Electrical Power and Energy Systems*, **43**, pp. 1-5 (2012).
- Sashirekha, A., Pasupuleti, J., Moin, N.H. and Tan, C.S. "Combined heat and power economic dispatch solved using Lagrangian relaxation with surrogate subgradient multiplier updates", *Electrical Power and Energy Systems*, **44**, pp. 421-430 (2013).
- Ramesh, V., Jayabarathi, T., Shrivastava, N. and Baska, A. "A novel selective particle swarm optimization approach for combined heat and power economic dispatch", *Electric Power Components and Systems*, **37**, pp. 1231-1240 (2009).
- Behnam, M.I., Mohammad, M.D. and Abbas, R. "Combined heat and power economic dispatch problem solution using particle swarm optimization with time varying acceleration coefficients", *Electric Power Systems Research*, **95**, pp. 9-18 (2013).
- Mehrdad, T.H., Saeed, T., Manijeh, A. and Parinaz, A. "Improved group search optimization method for solving CHPED in large scale power systems", *Energy Conversion and Management*, **80**, pp. 446-456 (2014).
- Esmaili, D.A., Esmaili, S. and Goroohi, V. "Multi-objective optimal location of SSSC and STATCOM achieved by fuzzy optimization strategy and harmony search algorithm", *Scientia Iranica D*, **20**(6), pp. 2024-2035 (2013).
- Bhattacharjee, K., Bhattacharya, A. and Halder Nee, D.S. "Teaching-learning-based optimization for different economic dispatch problems", *Scientia Iranica D*, **21**(3), pp. 870-884 (2014).

23. Venkata, R.R. and Kalyankar, V.D. “Multi-pass turning process parameter optimization using teaching-learning-based optimization algorithm”, *Scientia Iranica E*, **20**(3), pp. 967-974 (2013).
24. Kennedy, J. and Eberhart, R. “Particle swarm optimization”, *Proc IEEE Int. Conf. Neural Networks*, pp. 1942-8 (1995).
25. Yoshida, H., Kawata, K., Fukuyama, Y., Takayama, S. and Nakanishi, Y. “A particle swarm optimization for reactive power and voltage control considering voltage security assessment”, *IEEE Trans Power Syst*, **15**, pp. 1232-9 (2000).
26. Abido, M.A. “Optimal design of power system stabilizers using particle swarm optimization”, *IEEE Trans. Energy Conv.*, **17**(3), pp. 723-9 (2002).
27. Abido, M.A. “Optimal power flow using particle swarm optimization”, *Electr. Power Energy Syst.*, **24**, pp. 563-71 (2002).
28. Yu, B., Yuan, X. and Wang, J. “Short-term hydro-thermal scheduling using particle swarm optimization method”, *Energ. Convers. Manag.*, **48**, pp. 1902-1908 (2007).
29. Amjady, N. and Soleymanpour, H.R. “Daily hydrothermal generation scheduling by a new modified adaptive particle swarm optimization technique”, *Electric Power Systems Research*, **80**(6), pp. 723-732 (2010).
30. Shi, Y.H. and Eberhart, R.C. “A modified particle swarm optimizer”, *IEEE Intl. Conf. on Evolutionary Computation*, Anchorage, AK, pp. 69-73 (1998).
31. Eberhart, R.C. and Shi, Y.H. “Comparing inertia weights and constriction factors in particle swarm optimization”, *Proc. of the IEEE Congress on Evolutionary Computation*, USA, **1**, pp. 84-88 (2000).
32. Clerc, M. “The swarm and the queen: Towards a deterministic and adaptive particle swarm optimization”, *Proc. 1999 ICEC*, Washington, DC, pp. 1951-1957 (1999).
33. Haghrah, A., Behnam, M.I. and Seyedmonir, S. “Real coded genetic algorithm approach with random transfer vectors-based mutation for short-term hydro-thermal scheduling”, *IET Generation, Transmission & Distribution*, **9**, pp. 75-89 (2015).
34. He, S., Wu, Q.H. and Saunders, J.R. “Group search optimizer: An optimization algorithm inspired by animal searching behavior”, *IEEE Transactions on Evolutionary Computation*, **13**, pp. 973-989 (2009).

## Appendix

### Data for test system 5

Pure power units from 5 to 8:

$$F_{p5}(P_{p5}) = 25 + 2P_{p5} + 0.008P_{p5}^2 + \left| 100 \right. \\ \left. \times \sin(0.042 \times (P_{p5,\min} - P_{p5})) \right|$$

$$10 \leq P_{p5} \leq 75$$

$$F_{p6}(P_{p6}) = 60 + 1.8P_{p6} + 0.003P_{p6}^2 + \left| 140 \right. \\ \left. \times \sin(0.04 \times (P_{p6,\min} - P_{p6})) \right|$$

$$20 \leq P_{p6} \leq 125$$

$$F_{p7}(P_{p7}) = 100 + 2.1P_{p7} + 0.0012P_{p7}^2 + \left| 160 \right. \\ \left. \times \sin(0.038 \times (P_{p7,\min} - P_{p7})) \right|$$

$$30 \leq P_{p7} \leq 175$$

$$F_{p8}(P_{p8}) = 120 + 2P_{p8} + 0.001P_{p8}^2 + \left| 180 \right. \\ \left. \times \sin(0.037 \times (P_{p8,\min} - P_{p8})) \right|$$

$$40 \leq P_{p8} \leq 250$$

Power losses coefficient:

$$B = 10^{-6}$$

$$\times \begin{bmatrix} 39 & 10 & 12 & 15 & 15 & 16 & 14 & 13 & 15 & 15 \\ 10 & 40 & 14 & 11 & 15 & 20 & 16 & 18 & 19 & 20 \\ 12 & 14 & 35 & 17 & 20 & 18 & 15 & 15 & 14 & 13 \\ 15 & 11 & 17 & 39 & 25 & 19 & 12 & 18 & 16 & 16 \\ 15 & 15 & 20 & 25 & 49 & 14 & 14 & 15 & 16 & 17 \\ 16 & 20 & 18 & 19 & 14 & 45 & 28 & 30 & 30 & 28 \\ 14 & 16 & 15 & 12 & 14 & 28 & 19 & 20 & 22 & 20 \\ 13 & 18 & 15 & 18 & 15 & 30 & 20 & 24 & 24 & 25 \\ 15 & 19 & 14 & 16 & 16 & 30 & 22 & 24 & 30 & 30 \\ 15 & 20 & 13 & 16 & 17 & 28 & 20 & 25 & 30 & 31 \end{bmatrix}$$

## Biographies

**Thang Trung Nguyen** received his BEng and MEng degrees in Electrical Engineering from Ho Chi Minh City University of Technical Education, Ho Chi Minh City, Vietnam, in 2008 and 2010, respectively. Now, he is teaching in the Department of Electrical and Electronics Engineering, Ton Duc Thang University, Ho Chi Minh city, Vietnam. His research interests include optimization of power system, power system operation and control, and renewable energy.

**Dieu Ngoc Vo** received BEng and MEng degrees in



Electrical Engineering from HCM City University of Technology, Vietnam, in 1995 and 2000, respectively, and his DEng degree in Energy from Asian Institute of Technology (AIT), Pathumthani, Thailand, in 2007. He is now teaching in the Faculty of Electrical and

Electronic Engineering, HCM City University of Technology, Vietnam. His interests are applications of AI in power system optimization, power system operation and control, power system analysis, and power systems under deregulation.