

Comparative learning global particle swarm optimization for optimal distributed generations' output

Jasrul Jamani JAMIAN^{1,*}, Hazlie MOKHLIS², Mohd Wazir MUSTAFA¹,
Mohd Noor ABDULLAH^{2,3}, Muhammad Ariff BAHARUDIN¹

¹Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Johor, Malaysia

²Faculty of Engineering, University of Malaya, Kuala Lumpur, Malaysia

³Faculty of Electrical and Electronics Engineering, Universiti Tun Hussein Onn Malaysia, Johor, Malaysia

Received: 30.12.2012



Accepted: 20.02.2013



Published Online: 15.08.2014



Printed: 12.09.2014

Abstract: The appropriate output of distributed generation (DG) in a distribution network is important for maximizing the benefit of the DG installation in the network. Therefore, most researchers have concentrated on the optimization technique to compute the optimal DG value. In this paper, the comparative learning in global particle swarm optimization (CLGPSO) method is introduced. The implementation of individual cognitive and social acceleration coefficient values for each particle and a new fourth term in the velocity formula make the process of convergence faster. This new algorithm is tested on 6 standard mathematical test functions and a 33-bus distribution system. The performance of the CLGPSO is compared with the inertia weight particle swarm optimization (PSO) and evolutionary PSO methods. Since the CLGPSO requires fewer iterations, less computing time, and a lower standard deviation value, it can be concluded that the CLGPSO is the superior algorithm in solving small-dimension mathematical and simple power system problems.

Key words: Distributed generator, particle swarm optimization, power loss reduction, standard mathematical test function

1. Introduction

Many improvements have been made in distribution systems to increase the efficiency and reliability of the network, such as the implementation of reconfiguration techniques [1–3], the installation of FACTS devices [4,5] or capacitors [6,7], and the use of small-scale power generation technologies in the distribution network, also known as distributed generation (DG) [8–11]. Therefore, from the centralized power system in the last decade, the existence of DG units has transformed the topology of the network into a decentralized system.

The installation of DG units near the load center could help the system to improve power losses. However, uncontrolled power injection by the DG units to the system will cause problems in the distribution network as well as the grid network, even though the proportion of the DG is only a few tens of a percent of the total power consumption [12]. Therefore, the requirement to have an algorithm that can obtain the optimal DG's output is an important task that needs to be tackled by the power system planner in order to fully utilize the benefits of DG to the system performance. The metaheuristic optimization method is one of the alternative approaches that can be used to obtain the optimal DG output. Although many types of metaheuristic methods have been introduced, such as the genetic algorithm (GA), artificial bee colony, plant growth optimization,

*Correspondence: jasrul@fke.utm.my

fire fly optimization, and others, all of these methods have a similar operation mode, which is a random-based solution. This means that the methods use a possible random set solution as their initial process and update the solution until the optimal result is reached.

The optimal result value, the consistency in the results (giving similar results every time the algorithm is used), and the total computing time required by the algorithm in searching for the optimal solution are the important criteria to evaluate the effectiveness and robustness of the algorithm. In this paper, the modification on the original particle swarm optimization (PSO), which is known as the comparative learning global PSO (CLGPSO), is done by introducing the comparative learning concept for the acceleration coefficient (c_1 and c_2) and the 'sharing experience' concept in finding the best updating. With these 2 new concepts, the CLGPSO can give the best optimal and consistent results in many cases. The speed for the algorithm to search for the optimal solution is also improved (i.e. faster). The performance of the proposed algorithm is compared with 2 other PSO methods, which are inertia weight PSO (IWPSO) and evolutionary PSO (EPSO), in solving 6 benchmark mathematical functions as well as finding the optimal multiple DG units' output in a 33-bus distribution system.

The remainder of the paper is organized as follows. Section 2 presents related studies about the implementation of optimization methods in power system analysis, followed by the problem formulations with the list of constraints in Section 3. Section 4 introduces the optimization methods that are used in this study, as well as the new CLGPSO algorithm. Section 5 describes the performances of the PSO methods in solving the mathematical functions and multiple DG output problems, and Section 5 concludes the whole analysis of the study.

2. Related research

Many techniques and analyses have been carried out to obtain the optimal location and size of DG units in the distribution network, especially using the metaheuristic method. With the optimal location and output of DG units, the maximum benefit from this technology can be achieved. The study of Moradi et al. [13] is one of the latest to focus on finding the optimal DG's output in the distribution network. The authors used the GA to determine the best location of the DG and PSO for determining the optimal size of the DG. The result of the combination method was compared to the original PSO that was used to find the location and output of the DG and the original GA, which also found the location and output of the DG. Each technique resulted in a different allocation and DG output; ultimately, the GA/PSO combination gave the optimal location and best output in the 33- and 69-bus systems. However, since the GA and PSO analyses were done separately, the computing time for this combination method became higher compared to the original algorithm. Moreover, the performances of the GA/GA and GA/PSO in the analyses for finding the optimal DG location should be the same due to the separating analysis in the GA/PSO. However, the location given by the GA/PSO was different than that of the GA/GA result. In other papers, the performances of the optimization methods were compared with similar DG locations in the network [14–18]. With this in mind, the performance of the proposed algorithm in determining the optimal value, computing time, and number of iterations will be compared with existing optimization methods for the similar location of DGs (fair comparison).

Mouti et al. [19] implemented the artificial bee colony (ABC) in solving the DG output problem. They used 5 different DG operation mode cases in order to test the performance of the ABC in solving the optimal DG output problem. Moreover, there were 2 load types used in that study, which were the original load size and a load with a 50% increment in the system. The analysis only covered a single DG unit with a specified DG location. Although the ABC has only 2 parameters to be tuned compared to the evolutionary programming

(EP) and the PSO, as claimed by the authors, the ABC does not consist of the stopping criterion as the PSO and EP do (all of the particles achieve the same value at a certain iteration). The stopping criterion for the ABC is based on the maximum number of iterations. Therefore, in terms of computing time, the EP and PSO can give faster results compared to the ABC. However, in their research, the analysis showed that the ABC can give better optimal DG output and lower power losses compared to the GA approach. Lalitha et al. [20] proposed an analytical method to allocate and find multiple DG units in the distribution network. The use of multiple DGs improved the power losses to the lowest value compared to a single DG connection. The algorithm proposed by the authors used approximation methods to determine the maximum power reduction indicator to allocate the DG units without running the load flow analysis in multiple times. After completing the allocation process, the PSO algorithm was implemented to obtain the best DG size in the network.

Small-scale renewable energy sources (RESs) like photovoltaic (PV), wind, and solar systems are also categorized as DG units in the distribution network. For the general analysis, all of these RESs can be presented as DGs that operate in constant voltage (PV) or constant power (PQ) mode based on their characteristics. Moreover, from the recent advancement in the power electronic field, it is possible for some RESs to operate in both modes [21]. Furthermore, some researchers also considered the cost of operation and its availability in the analysis. Taher et al. [22] considered 3 types of RESs in their studies, which were PV, wind, and fuel cells. The authors implemented the modified honey ABC optimization method to allocate 8 fuel cell units, 2 PV units, and 2 wind energy units in the network, where the sizes of the RESs were assumed to be fixed. The proposed algorithm showed superior results compared to the traditional PSO and GA methods, where it gave the minimum total cost and power loss in the network. The same area of research was also investigated by Lee et al. [23]. However, in this study, the authors try to obtain the optimal wind turbine generator output with consideration of the voltage regulation as well as the protection system.

3. Background–problem formulation and constraints

In this manuscript, assuming that the locations of the DG units are fixed due to some constraints (example: near energy resource locations), the outputs for the DG are adjusted to minimize the power losses in the 33-bus distribution network using a novel optimization method called the CLGPSO. Since the connection of DG units in the distribution network changes the power flow direction in the system, with the proper DG output, it will significantly reduce the power losses in the network and also improve the voltage profile of the system. Therefore, the main objective is to obtain the minimum active power losses in the system based on a power loss formulation, as shown in Eq. (1):

$$P_{losses} = \sum_{L=1}^n |I_L|^2 R_L, \quad (1)$$

where L is the line's number in the system, I_L is the line current, and R_L is the line resistance.

Furthermore, the constraints that are considered during the process of finding the optimal DG output using the optimization methods are as follow.

a) Generator operation constraint:

$$P^{\min} \leq P_{DG} \leq P^{\max}. \quad (2)$$

All of the DGs are only allowed to operate within the acceptable limit, where P^{\min} and P^{\max} are the lower and upper bounds of the DG output. Therefore, the results of the DG's output must not exceed this limit during the initialization process or the updating process in the optimization technique.

b) Power injection constraint:

$$\sum_{B=1}^k P_{DG-B} < P_{Load} + P_{Losses}, \quad (3)$$

where k is the number of DG unit(s) in the system and B is the bus number.

In order to avoid the power injection to the main grid (substation) from the DG units, the total power output of the DGs must be less than the summation of total load (P_{load}) and power losses (P_{losses}) in the network. With this constraint, all of the DG units will always operate as generators in the system.

c) Power balance constraint:

$$\sum_{i=1}^k P_{DG} + P_{Substation} = P_{Load} + P_{Losses}. \quad (4)$$

The total power generated in the network, which is from the DG units and substation ($P_{substation}$), must be equal to the summation of total load and the total power losses.

d) Voltage bus constraint:

$$V_B^{new} \geq V_B^{old}. \quad (5)$$

The overall voltage profiles of the network must be higher than previous conditions (without DG) after the optimal DG's size is obtained. The voltage for each bus (V_B) should operate within the acceptable limit, which is within $\pm 5\%$ of the rated value.

The implementation of the CLGPSO in this study gives excellent performance in finding the optimal DG's output that fulfills all of the constraints. Furthermore, in order to validate the CLGPSO results, a comparison between the CLGPSO and 2 other optimization methods, which are the IWPSO and EPSO, is conducted in this paper.

4. Overview of the optimization methods in this study

The original PSO (OPSO) was developed by Eberhart and Kennedy [24] as an algorithm that works based on a stochastic population optimization strategy, inspired by the social behavior of a flock of birds and school of fish. The fish or birds will move in the same shape or formation until they reached the food source location. In other words, all of the fish/birds will have to adjust their position based on their speed to maintain the formation in the group. Therefore, their movement will depend on their own experience (local best: P_{best}) and other 'friends' in the group (global best: G_{best}). The updating of the local best and global best parameters for each iteration gives the potential to the OPSO to reach the most optimal solution. Many modifications or hybridizations are made for the OPSO algorithm to make the algorithm more robust and efficient in searching the optimal results.

4.1. IWPSO

The operation and process of IWPSO in finding the optimal solution is similar to that of the OPSO in all of the steps, except for the 'velocity' formula, which is used for finding a new population in the new iteration. Since

the OPSO algorithm originates from efforts of a model social system, the updating process for the algorithm is simple, as shown in Eq. (6).

$$v_{i-d}^{k+1} = v_{i-d}^k + c_1 r_1 (P_{best(i-d)}^k - x_{i-d}^k) + c_2 r_2 (G_{best-d}^k - x_{i-d}^k) \quad (6)$$

However, from an optimization point of view, proper control in global and local searching is needed so that the algorithm will have the capability of obtaining the global optimal point. With this in mind, Shi and Eberhart [25] adopted a new parameter in the velocity formula, called inertia weight (w). This inertia weight can be used to balance the global search and local search ability in the optimization process and reach the global optimum point. Thus, the new velocity formula for the IWPSO is shown in Eq. (7). Furthermore, rather than the 2 controllable parameters in the OPSO, the IWPSO algorithm consists of 3 controllable parameters that influence the exploration and exploitation of the algorithm. These 3 controllable parameters are the cognitive acceleration coefficient (c_1), social acceleration coefficient (c_2), and weight coefficient (w). Therefore, after obtaining all of the parameter values in the ‘velocity’ formula, the new position for N number of particles can be achieved using Eq. (8). Figure 1 shows the process to obtain the optimal solution in the IWPSO algorithm.

$$v_{i-d}^{k+1} = \omega^k v_{i-d}^k + c_1 r_1 (P_{best(i-d)}^k - x_{i-d}^k) + c_2 r_2 (G_{best-d}^k - x_{i-d}^k) \quad (7)$$

$$x_{i-d}^{k+1} = v_{i-d}^{k+1} + x_{i-d}^k \quad (8)$$

Here, x is the particle/variable, v is the velocity of the particle, i is the particle’s number in the N population, k is the iteration index, d is the d dimension (variable) of the i th particle, c_1 and c_2 are the cognitive and social parameter coefficients, w^k is the weight coefficient for the k th iteration, r_1 and r_2 are random numbers with a range of [0,1], P_{best}^k is the best position particle achieved based on its own experience, and G_{best}^k is the best particle position based on the swarm’s overall experience.

4.2. EPSO

The EPSO is an algorithm that hybridizes the PSO algorithm with EP. By applying the concepts of combination, competition, and selection of the EP into the PSO algorithm, the EPSO maintains only the strongest particle in the population, while eliminating the others (the weak ones). Thus, the strongest particles in the EPSO allow the algorithm to achieve a faster convergence solution compared to the original PSO [26,27]. The difference between the EPSO and OPSO starts after the new population set is obtained. The new population is combined with the previous population and creates $2N$ number of particles (since the size of each population = N). After that, based on the competition rate that has been set by the user, the competition between $2N$ particles is executed. Let the rate of competition be 25% and $N = 20$, and each particle will compete against the other particles 10 times ($2 \times 20 \times 25\% = 10$) for the best fitness value. The winning particle (lower or higher, based on its objective function) will obtain a score, while the loser will have a mark of zero. The competition is continued until all of the particles complete the competition. Next, the $2N$ particles are ranked based on their competition mark and the best N particles are forwarded to the next iteration. The G_{best} and P_{best} values in the EPSO are automatically obtained after this selection process. Since all of the survival particles are selected as the P_{best} value, the EPSO’s velocity formula for the second iteration onward becomes simpler, as shown in Eq. (9). The process to update the new position of the particles is still similar to that in the PSO or IWPSO algorithm, which uses Eq. (8).

$$v_{i-d}^{k+1} = \omega^k v_{i-d}^k + c_1 r_1 (P_{best(i-d)}^k - x_{i-d}^k) \quad (9)$$

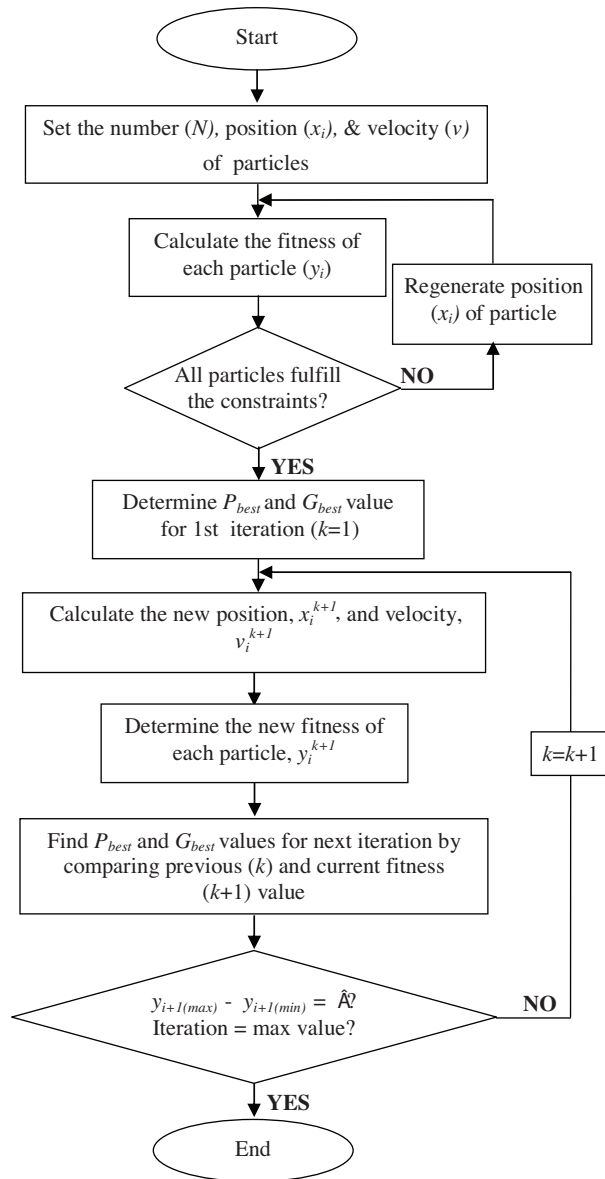


Figure 1. Flow of the IWPSO algorithm in searching for the optimal solution.

4.3. Proposed CLGPSO

In order to improve the quality and robustness of the OPSO, a new CLGPSO is introduced. In the OPSO, all of the particles move in the group within the specific shape or formation and share the same cognitive (c_1) and social (c_2) acceleration coefficient values to move toward the new position. However, in the CLGPSO, the c_1 and c_2 values for each particle vary, where they are depending on the fitness difference between the current particles' position and the best position, which holds the best fitness (fit_{best}) value in the current iteration as shown in Eqs. (10) and (10). The difference between the best fitness values and the current particle's fitness value will identify either the c_1 or c_2 values for that particle as larger or smaller [Eqs. (12) and (13)]. Since all of the particles have their own cognitive and social acceleration coefficients, a competition concept between

the particles exists. All of the particles try to adjust their own speed using an individual c_{1i} and c_{2i} to arrive at the best position (optimal value).

$$fit(x_i^k) = \frac{1}{1 + f(x_i^k)} : \text{For the minimization problem}$$

$$fit(x_i^k) = \frac{1}{1 - f(x_i^k)} : \text{For the maximization problem} \tag{10}$$

$$fit_{best}^k = \max [fit(x_1^k), fit(x_2^k), \dots, fit(x_N^k)], \tag{11}$$

$$c_{1i}^k = \frac{fit_{best}^k - fit(x_i^k)}{fit_{best}^k} (c_{1\max}), \tag{12}$$

$$c_{2i}^k = 2 - c_{1i}^k. \tag{13}$$

where $f(x_i)$ is the objective function value in the optimization problem and $c_{1\max} = 2.0$.

Furthermore, in order to prevent the CLGPSO from being trapped in the local minima during the competition, the value of the cognitive acceleration is given priority (higher value) compared to the social acceleration when the fitness difference between the best position and current position is bigger. This is needed to prevent all of the particles from moving to the local optimal position during the searching process. On the other hand, when the fitness difference becomes smaller, the c_2 value will be higher so that the particle can determine the best global position result. Figures 2 and 3 illustrate the CLGPSO concept and the example of the c_1 and c_2 results between 2 particles in finding the optimal position, respectively. From Figure 3, it can be seen that the values of c_1 and c_2 for both particles are different at each iteration (depending on the random number and new position generated). In addition, the value of c_1 decreases to a smaller value at a higher number of iterations, while the value of c_2 becomes higher due to the fitness difference between the particles and the global solution becomes smaller.

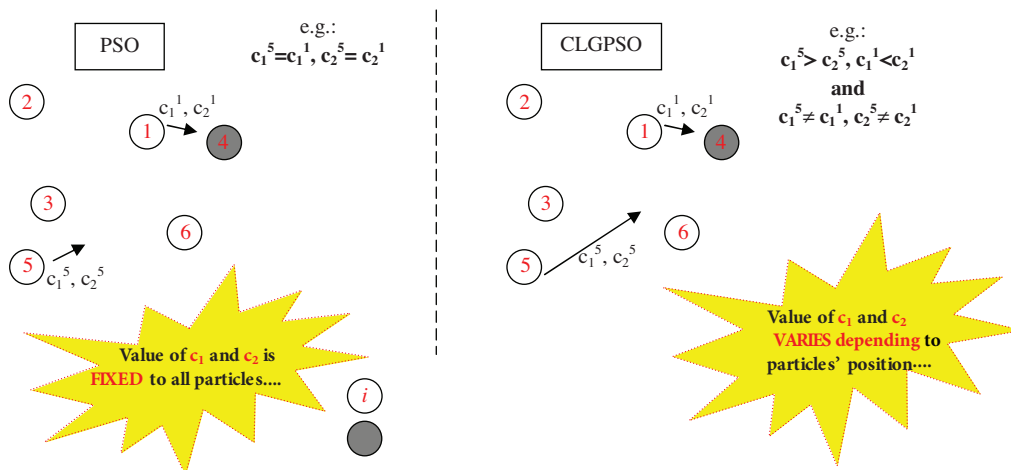


Figure 2. Illustration and comparison of c_1 and c_2 between PSO and CLGPSO.

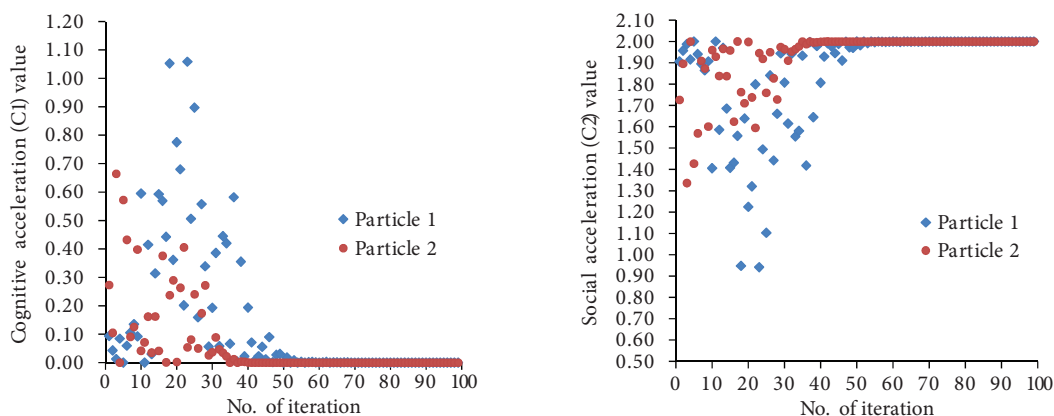


Figure 3. Example of the c_1 and c_2 variation between 2 particles in 100 iterations.

Additionally, in the OPSO, the particle is only considered the best obtained value until the current iteration (P_{best}) and best value in the population (G_{best}) update the velocity. This means that there is no information sharing between current populations and other populations from any previous iteration. This information sharing is important to help guide the particle in finding the suitable velocity based on other populations' 'experiences'. Therefore, the new parameter, E_{best} , in the CLGPSO is defined as the value of G_{best} that is randomly selected among the existing particles at any iteration until the current iteration ($1:k$) within its own dimension (d), as shown in Eq. (14). The acceleration coefficient equation for the E_{best} parameter (c_3) is shown in Eq. (15), where the value is also different for each particle.

$$E_{best(i-d)}^k = f(rand(G_{best}), 1 : k, d) \tag{14}$$

$$c_{3i}^k = c_{1i}^k \left(1 - e^{-2 \times c_{1i}^k \times iter} \right) \tag{15}$$

Here, $iter$ is the number of current iterations.

Thus, the new velocity formula for updating the next position of the particles in the CLGPSO algorithm is:

$$v_{i-d}^{k+1} = \omega^k v_{i-d}^k + c_{1i}^k r_1 (P_{best(i-d)}^k - x_{i-d}^k) + c_{2i}^k r_2 (G_{best-d}^k - x_{i-d}^k) + c_{3i}^k (E_{best(i-d)}^k - x_i^k). \tag{16}$$

Although E_{best} in the CLGPSO uses the concept of the random selection global best value, the topology for this algorithm is still a fully connected topology due to the parameters of the local best (P_{best}) that still exist in the algorithm. E_{best} is only used to interrelate the previous results with the current updating position.

5. Results and discussion

Three different types of improved PSO (IWPSO, EPSO, and CLGPSO) are compared in this section for 6 mathematical standard test functions, as well as in solving power system problems (minimizing power losses). Since the problem of finding the optimal DG output is simple (low-dimensional mathematical problem), the performances of these optimization methods are also tested on a fixed and low-dimensional function. The population size for the particles is standardized to 20 ($N = 20$) with the cognitive and social coefficient values (c_1 and c_2) set as 2.0 ($c_{max} = 2.0$ for the CLGPSO). All of the algorithms are tested multiple times until reaching the maximum iteration (100) for all of the mathematical functions, as well as in the power system problem, and the convergence iteration is recorded for evaluating the effectiveness and robustness of the algorithm.

5.1. Comparison of the CLGPSO, EPSO, and PSO performances for the mathematical standard test functions

Six classical benchmark functions with fixed dimensions are used in this study, consisting of unimodal (U), multimodal (M), separate (S), and nonseparable (N) models, as shown in Table 1. All of the range and global minimum values for the test functions are stated as well. The performance of the CLGPSO is compared with those of the IWPSO and EPSO 50 times, where every repeating process involves a new random population and the results of the best value, worst value, mean, and standard deviation (SD) are recorded, as shown in Table 2.

Table 1. Fixed dimensional benchmark functions used in the experiments.

Formulation	Range	Min	Char*
$f_1(x) = (1.5 - x_1 + x_1x_2)^2 + (2.25 - x_1 + x_1x_2^2)^2 + (2.25 - x_1 + x_1x_2^3)^3$	[-4.5, 4.5]	0	UN
$f_2(x) = 0.26(x_1^2 + x_2^2) - 0.48x_1x_2$	[-10, 10]	-48.1538	UN
$f_3(x) = x_2 \sin(4x_1) + 1.1x_1 \sin(2x_2)$	[0, 10]	-18.5916	UN
$f_4(x) = (x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6)^2 + 10(1 - \frac{1}{8\pi}) \cos(x_1) + 10$	[-5, 15]	106.0730	MS
$f_5(x) = x_1 \sin(4x_1) + 1.1x_2 \sin(2x_2)$	[0, 10]	-18.5547	MS
$f_6(x) = x_1^2 + 2x_2^2 - 0.3 \cos(3\pi x_1 + 4\pi x_2) + 0.3$	[-100, 100]	-1.03163	MN

*Char = characteristic.

Table 2. Best, worst, mean, and SD values after 50 runs of functions from f_1 to f_6 .

Function	Dim	Min		Best	Worst	Mean	SD
f_1	2	0	IWPSO	0.02552	1.094985	0.196635	0.396053
			EPSO	0.02552	1.094985	0.089688	0.256562
			CLGPSO	0.02552	1.094985	0.089688	0.256562
f_2	2	-48.1538	IWPSO	-48.1538	-48	-48.1508	0.021757
			EPSO	-48.1538	-48	-48.1508	0.021757
			CLGPSO	-48.1538	-48.1538	-48.1538	0
f_3	2	-18.5916	IWPSO	-18.5916	-11.7088	-16.0854	2.078569
			EPSO	-18.5916	-11.7088	-17.023	2.166951
			CLGPSO	-18.5916	-15.4525	-17.7127	1.423769
f_4	2	106.0730	IWPSO	106.073	106.073	106.073	0
			EPSO	106.073	106.2417	106.0767	0.023895
			CLGPSO	106.073	106.073	106.073	0
f_5	2	-18.5547	IWPSO	-18.5547	-9.03554	-16.1967	2.673396
			EPSO	-18.5547	-9.03554	-17.1695	2.329804
			CLGPSO	-18.5547	-15.1079	-17.8654	1.392716
f_6	2	0	IWPSO	-1.03163	-1.03163	-1.03163	0
			EPSO	-1.03163	-1.03163	-1.03163	0
			CLGPSO	-1.03163	-1.03163	-1.03163	0

From the results, the IWPSO, EPSO, and CLGPSO algorithms give the same best values for all of the test functions, where the best optimal values for f_1 , f_2 , f_3 , f_4 , f_5 , and f_6 are 0.02552, -48.1538, -18.5916, 106.073, -18.5547, and -1.03163, respectively. Compared with the standard optimal values in Table 2, almost all of the functions give the same minimum value, except for f_1 , where the PSO algorithms have a higher minimum value compared to the standard minimum value. However, the difference between these results is not

very critical and can be accepted. Although the IWPSO, EPSO, and CLGPSO give similar optimum results, the CLGPSO is the best optimization method compared to the other 2 PSOs. This is because the worst result (biggest value in 50 trials), which is obtained by the CLGPSO, is closer to the optimal value in all of the test functions and also has the lowest SD value after the algorithm is tested 50 times, as shown in Figure 4.

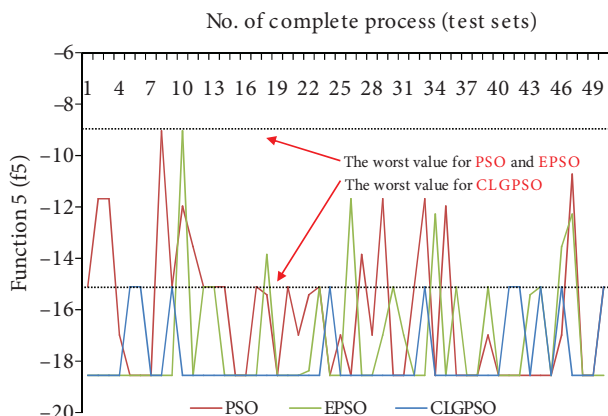


Figure 4. Final results of PSO, EPSO, and CLGPSO at the maximum iteration.

Comparing the results between the EPSO and IWPSO, we see that the EPSO gives a lower SD value for f_1 and f_5 and has the same performance for f_2, f_4 , and f_6 . The IWPSO algorithm has better performance for the other functions. However, the worst value obtained by both algorithms is similar, except for f_4 . Therefore, the performances of the IWPSO and EPSO are not much different in these 6 standard test functions. Thus, it can be concluded that the CLGPSO gives the best results in solving the classical benchmark functions used in this analysis.

5.2. Comparison of the CLGPSO, EPSO, and PSO performances for the DG output problem

The performances of the IWPSO, EPSO, and CLGPSO in finding the optimal DG output are tested on a 33-bus distribution system, as shown in Figure 5. By having the optimal DG output, the distribution system will operate at a minimum power loss value. Therefore, in this analysis, the DG output is assigned as the particles (variables, ' x_i ') and the power loss value is assigned as the fitness (objective function, ' y '). Furthermore, all of the power system constraints listed in Section 3 are checked by the algorithm at the initialization process, as well as every time the updating position of the particles is generated. The penalty factor concept is used for the particles, which does not fulfill all of the constraints. The analysis is performed using MATLAB 7.8 on a 2.0-GHz Intel Core2Duo processor with 2 GB of RAM. The DG units are located at buses 6, 16, and 25, with the initial size and location found using the analytical method reported in [20]. However in this analysis, the DG operation mode is PV mode. Thus, it can be said that the analysis in this paper consists of a single objective function (minimizing power loss) with 3 variables, which are the DG output for units 1 (bus 6), 2 (bus 16), and 3 (bus 25).

From Table 3, the power loss in the network is reduced from 203.1854 kW to 51.4814 kW after the DG units are located in the distribution network, and it further improves to 33.2618 kW after the DG outputs are optimized using the PSO methods. Since the DG units are located and sized one at a time in the algorithm [20], it causes the solution to be trapped in the local optimal value. That is why the PSO methods give better DG outputs for reducing the power losses. Not only that, but among the PSO methods, it can be seen that

the optimal DG outputs are nearly the same where the different results in all 3 units are only at the fourth decimal places. However, the CLGPSO is superior due to the number of iterations, and the computing time per iteration for the algorithm to solve the problem is less than that of the IWPSO and EPSO. Among 20 sets of simulation results, the CLGPSO obtains the optimal DG output within 35 to 59 iterations. Meanwhile, the IWPSO and EPSO require at least 58 to 96 and 48 to 65 iterations, respectively.

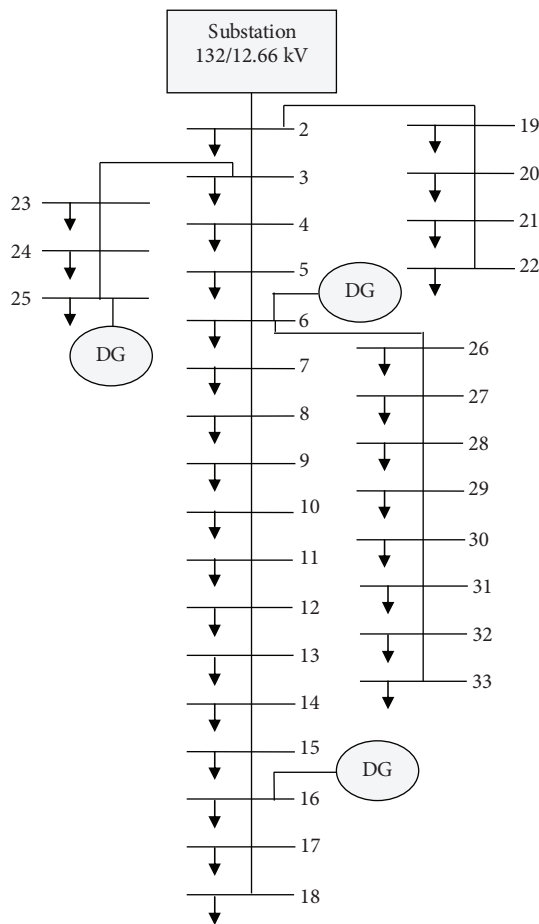


Figure 5. The 33-bus radial distribution system with 3 DG units operated in PV mode.

Table 3. Performance of the proposed algorithms in minimizing the power losses.

	w/o DG	Method [20]*	IWPSO	EPSO	CLGPSO
DG 1 (MW)	-	2.4878	1.70763	1.70769	1.70767
DG 2 (MW)	-	0.3556	0.54711	0.54694	0.54708
DG 3 (MW)	-	0.7291	0.76939	0.76942	0.76946
P. losses (kW)	203.1854	51.4814	33.26178	33.26178	33.26178
Min. iteration	-	-	58	48	35
Max. iteration	-	-	96	65	59
CT/I**	-	-	8.31441	12.72179	7.620771
SD	-	-	0.00000058150	0.0000002283	0.0000002283

*: Without optimization.

** : Computing time/iteration.

Although the EPSO requires fewer iterations to solve the problem compared to the IWPSO method, in terms of the computing time, the IWPSO is faster due to its simplicity. The process of combination, competition, and selection in the EPSO algorithm makes the time for the procedure to complete one iteration (to find a new position) longer. However, the CLGPSO gives the lowest value for both results (computing time and number of iterations). In terms of consistency, the EPSO and CLGPSO give better performance compared to the IWPSO. The consistency can be seen from the SD values. In 20 sample test results, both the EPSO and CLGPSO have SD values equal to 0.0000002283, while the IWPSO has a slightly higher value of 0.00000058150. Therefore, it can be said that the CLGPSO can give faster computing time, fewer iterations, and consistent results in this analysis.

Figure 6 shows the comparison of the best results of IWPSO, EPSO, and CLGPSO in determining the minimum power losses in the network. From the pattern, the IWPSO algorithm has the capability to reach close to the optimal results faster than EPSO and CLGPSO (between the 1st to 30th iterations). However, in terms of the global optimal solution, CLGPSO gives the fastest results, followed by EPSO and IWPSO at the 48th and 58th iterations, respectively. The control method of the c_1 and c_2 methods and the additional parameter (E_{best} and c_3) in CLGPSO help the algorithm to reach the optimal results faster than the others.

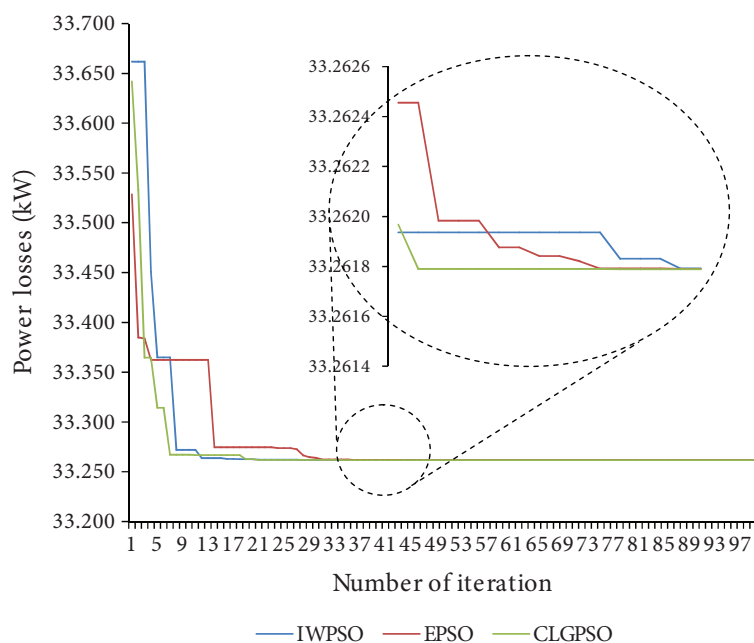


Figure 6. Comparison of convergence curve by PSO, EPSO, and CLGPSO.

The impacts of the optimal DG's output can also be seen from the voltage profile of the network. Figure 7 shows the voltage profile improvement by the DG units compared to the original voltage profile. From the results, a tremendous voltage profile improvement can be seen when 3 DGs units are located at buses 6, 16, and 25. For the system without any DG units, almost half of the total buses in the system operated at lower than the acceptable limit (black dashed line). With the DG units, the voltage value for all of buses in the network is in the range of $1.05 < V < 0.95$. By comparing the voltage profile results that are obtained after the optimal DG's output is determined using mathematical and PSO techniques, it can be seen that the voltage profile given by the mathematical approach is better than that of the PSO technique. However, the difference between these 2 approaches is very small. From the smaller figure (zoomed in), the difference between both results is only

around 0.001 p.u. (0.1%) between buses 7 to 12, while the other buses have the same voltage value. Comparing the power loss improvements, the PSO method gives an 8.9% improvement over the mathematical approach.

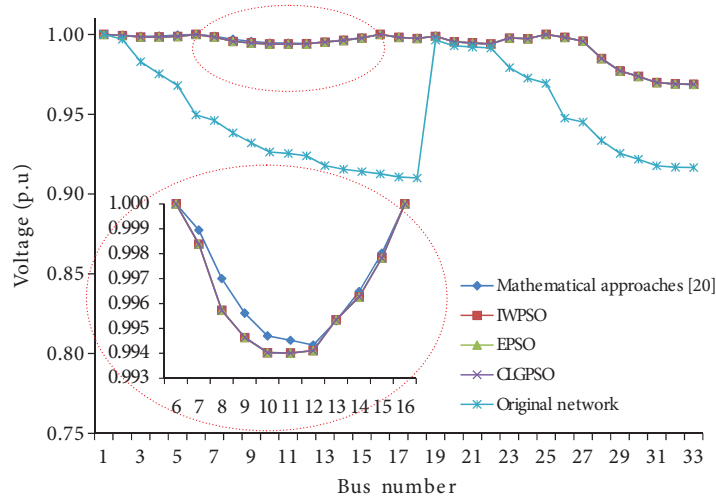


Figure 7. Comparison of the distribution network’s voltage profile for all of the cases.

Since CLGPSO gives similar results to those of IWPSO and EPSO for the voltage profile and power losses, it can be concluded that the performance of the CLGPSO in finding the optimal DG’s output is superior to that of IWPSO and EPSO due to its faster computing time, fewer iterations, and smaller SD.

6. Conclusion

The implementation of varying cognitive (c_1) and social (c_2) acceleration coefficients for each particle with the new additional parameter in CLGPSO helps the algorithm to have an optimal solution faster than IWPSO and EPSO, both in terms of the computing time and the number of iterations. The performance of CLGPSO was tested on 6 standard benchmark low-dimensional mathematical functions, as well as in a power system for finding the optimal DG’s output. CLGPSO gave the most consistent results, indicated by the lowest SD value in the mathematical functions after it was executed 50 times in individual runs. Furthermore, the worst value obtained by CLGPSO was close to the standard optimal value when compared to the other optimizations.

For the power system case, the analysis showed that all of the PSO methods gave similar results for the optimal DG’s output, which indirectly gave similar power loss values in the system. The power loss improvement given by the PSO method was nearly 84% compared to the original network power loss in the 33-bus distribution system. Although the voltage profile given by the mathematical approached was better than that of the PSO techniques, it only occurred at certain buses and the difference in the value was very small (acceptable). However, just as in the mathematical analysis, CLGPSO always gave the fastest computing time, lowest number of iterations, and smallest SD compared to IWPSO and EPSO for the same optimal DG results. Since CLGPSO provided excellent results compared to IWPSO and EPSO in all aspects, it can be concluded that CLGPSO is the superior optimization method in solving the problems in this study. Finally, the results (optimal power output for each DG) obtained in the analysis give an indicator of the suitable size for the DG units so that overinvestment in DG size can be avoided.

References

- [1] M.R. Haghifam, O.P. Malik, “Genetic algorithm-based approach for fixed and switchable capacitors placement in distribution systems with uncertainty and time varying loads”, *IET Generation, Transmission & Distribution*, Vol. 1, pp. 244–252, 2007.
- [2] B. Venkatesh, R. Ranjan, H.B. Gooi, “Optimal reconfiguration of radial distribution systems to maximize loadability”, *IEEE Transactions on Power Systems*, Vol. 19, pp. 260–266, 2004.
- [3] C.T. Su, C.F. Chang, J.P. Chiou, “Distribution network reconfiguration for loss reduction by ant colony search algorithm”, *Electric Power Systems Research*, Vol. 75, pp. 190–199, 2005.
- [4] E.J. De Oliveira, J.W. Marangon Lima, K.C. De Almeida, “Allocation of FACTS devices in hydrothermal systems”, *IEEE Transactions on Power Systems*, Vol. 15, pp. 276–282, 2000.
- [5] M.M. El Metwally, A.A. El Emary, F.M. El Bendary, M.I. Mosaad, “Optimal allocation of FACTS devices in power system using genetic algorithms”, *12th International Middle-East in Power System Conference*, pp. 1–4, 2008.
- [6] G. Carpinelli, C. Noce, D. Proto, A. Russo, P. Varilone, “Single-objective probabilistic optimal allocation of capacitors in unbalanced distribution systems”, *Electric Power Systems Research*, Vol. 87, pp. 47–57, 2012.
- [7] G. Carpinelli, F. Mottola, D. Proto, A. Russo, “Optimal allocation of dispersed generators, capacitors and distributed energy storage systems in distribution networks”, *Proceedings of the International Symposium in Modern Electric Power Systems*, pp. 1–6, 2010.
- [8] S. Ghosh, S.P. Ghoshal, S. Ghosh, “Optimal sizing and placement of distributed generation in a network system”, *International Journal of Electrical Power & Energy Systems*, Vol. 32, pp. 849–856, 2010.
- [9] S.G.B. Dasan, S.S. Ramalakshmi, R.P.K. Devi, “Optimal siting and sizing of hybrid Distributed Generation using EP”, *International Conference on Power Systems*, pp. 1–6, 2009.
- [10] S.M.H. Nabavi, S. Hajforoosh, M.A.S. Masoum, “Placement and sizing of distributed generation units for congestion management and improvement of voltage profile using particle swarm optimization”, *IEEE PES in Innovative Smart Grid Technologies Asia*, pp. 1–6, 2011.
- [11] T. Gözel, M.H. Hocaoglu, “An analytical method for the sizing and siting of distributed generators in radial systems”, *Electric Power Systems Research*, Vol. 79, pp. 912–918, 2009.
- [12] E. Haesen, J. Driesen, R. Belmans, “A long-term multi-objective planning tool for distributed energy resources”, *IEEE PES in Power Systems Conference and Exposition*, pp. 741–747, 2009.
- [13] M.H. Moradi, M. Abedini, “A combination of genetic algorithm and particle swarm optimization for optimal DG location and sizing in distribution systems”, *International Journal of Electrical Power Energy Systems*, Vol. 34, pp. 66–74, 2012.
- [14] V. Hengsrıtawat, T. Tayjasant, N. Nimpitiwan, “Optimal sizing of photovoltaic distributed generators in a distribution system with consideration of solar radiation and harmonic distortion”, *International Journal of Electrical Power & Energy Systems*, Vol. 39, pp. 36–47, 2012.
- [15] L. Bo, Y. Zhiqiang, B. Hailong, “Intelligent optimization algorithm for the locating and sizing of distributed generation planning”, *China International Conference on Electricity Distribution*, pp. 1–6, 2010.
- [16] J.J. Jamian, M.W. Mustafa, H. Mokhlis, M.N. Abdullah, “Comparative study on distributed generator sizing using three types of particle swarm optimization”, *International Conference on Intelligent Systems, Modelling and Simulation*, pp. 131–136, 2012.
- [17] A.M. El-Zonkoly, “Optimal placement of multi-distributed generation units including different load models using particle swarm optimization”, *Swarm and Evolutionary Computation*, Vol. 1, pp. 50–59, 2011.
- [18] M. Mohammadi, S.H. Hosseinian, G.B. Gharehpetian, “GA-based optimal sizing of microgrid and DG units under pool and hybrid electricity markets”, *International Journal of Electrical Power & Energy Systems*, Vol. 35, pp. 83–92, 2012.

- [19] F.S. Abu-Mouti, M.E. El-Hawary, "Optimal distributed generation allocation and sizing in distribution systems via artificial bee colony algorithm", *IEEE Transactions on Power Delivery*, Vol. 26, pp. 2090–2101, 2011.
- [20] M.P. Lalitha, V.C.V. Reddy, V. Usha, "Optimal DG placement for minimum real power loss in radial distribution systems using PSO", *Journal of Theoretical and Applied Information Technology*, Vol. 13, pp. 107–116, 2010.
- [21] S.M. Moghaddas-Tafreshi, E. Mashhour, "Distributed generation modeling for power flow studies and a three-phase unbalanced power flow solution for radial distribution systems considering distributed generation", *Electric Power Systems Research*, Vol. 79, pp. 680–686, 2009.
- [22] T. Niknam, S.I. Taheri, J. Aghaei, S. Tabatabaei, M. Nayeripour, "A modified honey bee mating optimization algorithm for multiobjective placement of renewable energy resources", *Applied Energy*, Vol. 88, pp. 4817–4830, 2011.
- [23] H.J. Lee, G.T. Son, J.W. Park, "Study on wind-turbine generator system sizing considering voltage regulation and overcurrent relay coordination", *IEEE Transactions on Power Systems*, Vol. 26, pp. 1283–1293, 2011.
- [24] J. Kennedy, R. Eberhart, "Particle swarm optimization", *IEEE Proceedings of the International Conference on Neural Networks*, Vol. 4, pp. 1942–1948, 1995.
- [25] S. Yuhui, R.C. Eberhart, "Fuzzy adaptive particle swarm optimization", *Proceedings of the Congress on Evolutionary Computation*, Vol. 1, pp. 101–106, 2001.
- [26] J.J. Jamian, M.W. Mustafa, H. Mokhlis, M.A. Baharudin, "Implementation of evolutionary particle swarm optimization in distributed generation sizing", *International Journal of Electrical and Computer Engineering*, Vol. 2, pp. 137–146, 2012.
- [27] V. Miranda, N. Fonseca, "EPSO - best-of-two-worlds meta-heuristic applied to power system problems", *Proceedings of the Congress on Evolutionary Computation*, pp. 1080–1085, 2002.