

The Power Unit Coordinated Control via Uniform Differential Evolution Algorithm

Zain Zaharn^{*1}, Ruifeng Shi², Xiangjie Liu³

School of Control and Computer Engineering, North China Electric Power University
Beinong Road 2, Changping District, Beijing, 02206, China

*Corresponding author, e-mail: zainozahran@gmail.com

Abstract

This paper modified the differential evolution (DE) algorithm adaptively to solve the power unit coordinated control (PUCC) problem. It was modified in two aspects: 1) a uniform initialization with a controlling zone factor (m), 2) a regular mutation process, to develop an effective searching mechanism and improve the performance of the basic DE algorithm. A numerical case study was employed to verify the performance of our proposed uniform differential evolution (UDE) algorithm, by contrast to the basic DE, genetic algorithm (GA), and particle swarm optimization (PSO) algorithm. The experimental simulation results show that our proposed UDE algorithm has outperformed the other comparative algorithms, which demonstrate the effectiveness and efficiency of the new algorithm.

Keywords: power unit coordinated control problem, differential evolution algorithm, uniform differential evolution algorithm, genetic algorithm, particle swarm optimization algorithm

Copyright © 2013 Universitas Ahmad Dahlan. All rights reserved.

1. Introduction

The PUCC problem is an interesting task for many researchers during these decades. The power unit is composed of a boiler-turbine system coupled to an electric power generator. The boiler-turbine configuration is a multivariable, nonlinear, and time varying-system with complex operation due to the uncertainties and high settling time. So, the main goal of the boiler-turbine control is to justify the generator output power to maintain a high response to the load fluctuation, while keeping the steam pressure and temperature within the permissible values. For simplification, the power unit might be considered as three-input three-output system, in which the essential inputs are the boiler firing rate, position of the steam valve, and the feeding water, while the outputs are the electric power, steam pressure, and water level deviation. Controlling of this system can be based on the stored thermal energy in the boiler or on the boiler-turbine governor as two alternative conventional techniques. As the first technique is slow with high stability, the latter is fast to follow the load variations but might be unstable. The coordinated control is an integration of the two conventional techniques to result in a stable control system with high level of response. Then the power unit is to be optimized under considerations of minimizing the load tracking error, injecting fuel, steam losses, and feeding water. There are different published works in the literature, through which the PUCC problem is solved according to the mentioned considerations. In [1], it was solved traditionally to prevent cyclic damage to the plant by designing prototype load controllers to adjust the flow of air and fuel to the boiler and flow of steam to the turbine. In [2], the solution was based on coordinated control with pressure set point scheduling by finding a single solution with the preferences of the problem. In [3-5], it was solved by modern heuristic algorithms such as GA, DE, PSO, and Pareto multi-objective optimization (PMOO) techniques, depending on a reference governor and optimization unit. In [6, 7], a multi-agent system was presented to solve this problem, through which a computer software programs work independently in a number of stages to establish the required coordinated control. In [8], a nonlinear multivariable power plant coordinated control by constrained predictive scheme was created to maintain the thermal constrained during load fluctuations. In [9], a new control strategy was used to control the power plant by Smith's predictor to overcome the time-lag influences.

The DE algorithm was proposed in 1996 by Storn and Price [10] as a new evolutionary algorithm, which has a simple structure and efficient solutions. Thus, its applications are

growing rapidly during these decades through a wide range of problems [11] as shown in the current literature. Such as [12], in which the DE algorithm was used to solve the problem of the optimal power flow taking the Algerian electrical network as case-study. In [13], a new multi-populated DE algorithm was proposed to solve the real-parameter constrained optimization problems. In [14], the DE algorithm was extended to solve the multi-objective optimization problems by using Pareto-based approach. Furthermore, the DE algorithm has the ability to be combined with other algorithms to strengthen their effectiveness, such as [15], in which it was used to combine with a simple algorithm to result in a powerful technique for the measured data alignment to the original surface model. As the DE algorithm has a poor rate of convergence when dealing with the problem of high objectives [16], it is required to improve its convergence to become more robust algorithm. Through this issue, the convergence of DE algorithm is improved to result in the proposed UDE algorithm to adaptively solve the PUC problem. Then its performance is compared with the basic DE, GA, and PSO algorithms.

The GA was proposed by John Holland in 1975 as the first evolutionary algorithm, which was created from the regeneration and natural selection [17]. Then, it was developed through these years to represent a powerful algorithm. PSO was proposed by Kennedy and Eberhart in 1995 [18] as a population-based stochastic search algorithm. It was successfully employed to solve a wide range of problems during the recent years.

2. The Improved Differential Evolution Algorithm

2.1. The Basic Differential Evolution Algorithm

It depends on the weighted differences between random individuals, so it is more generic than other evolutionary algorithms (EAs). The process of this algorithm is based on selection, encoding, mutation, and crossover as other EAs, with addition to the ability of using floating-point representation. The DE algorithm is initialized according to Equation (1) to find the initial population (v_i^0). Then a trail of individuals can be generated by mutating the current vector (v_i^p), according to the three schemes of mutation as in Equations (2) to (4). The DE algorithm depends on mutation as essential operator to reproduce the new vectors rather than crossover operator in searching process. In fact, mutation process is used in the DE algorithm to find the diversity through the individual of each population, based on the mutation factor $F(0,1)$, which is used to amplify the vector differences. On the other hand, the crossover probability is used to determine the recombined vector according to Equation (5) with the crossover factor $f(0,1)$. It is required to reduce the computational time cost by usage of small population size. Therefore, the population size is to be tuned carefully to avoid the slow finishing or premature convergence in case of using small, or high population size respectively. The preferable population size through the literature is $20D$ (D is the number of parameters). Selection is an important operator in DE procedure, by which the candidate individuals are to be determined for the next stage, according to evaluation process, which is based on tournament or other stochastic techniques. This procedure is repeated until the stopping criterion is satisfied.

$$v_{i,j}^0 = u_{i,j}^L + \text{rand}(0,1)(u_{i,j}^H - u_{i,j}^L). \quad (1)$$

$$v_{i,j}^c = v_{i,j}^p + F(v_{1,j}^p - v_{2,j}^p). \quad (2)$$

$$v_{i,j}^c = v_{best(i,j)}^p + F(v_{1,j}^p - v_{2,j}^p). \quad (3)$$

$$v_{i,j}^c = v_{i,j}^p + F(v_{best(i,j)}^p - v_{i,j}^p) + F(v_{1,j}^p - v_{2,j}^p). \quad (4)$$

for $i = 1, 2, \dots, N$, and $j = 1, 2, \dots, D$.

Where $v_{i,j}^0$ is the initial population; $u_{i,j}^L$ and $u_{i,j}^H$ are lower and higher boundaries of the power-pressure window; $\text{rand}(0,1)$ is a random number between 0 and 1; $v_{i,j}^p$ are the

generated individuals; $v_{i,j}^p$ are the current individuals, $v_{1,j}^p$ and $v_{2,j}^p$ are two random individuals; $v_{best(i,j)}^p$ is the best individual in each population; N is the population size; and D is the number of parameters.

$$v_{i,j}^c = \begin{cases} f(v_{i,j}^p, v_{best(i,j)}^p, v_{1,j}^p, v_{2,j}^p) & \text{if } rand_{i,j} \subseteq CR \text{ and } j = I_{rand}, \\ v_{i,j}^p & \text{otherwise.} \end{cases} \quad (5)$$

for $i = 1, 2, \dots, N$, $\text{ and } j = 1, 2, \dots, D$.

Where $rand_{i,j}$ is a random number between $(0,1)$ and I_{rand} is a random integer from 1 to D .

2.2. The Proposed Algorithm

In this paper, the DE algorithm is modified to adaptively solve the PUC problem. The proposed UDE algorithm is initialized uniformly rather than the random initialization of the basic DE algorithm. In which the initial population is controlled and distributed uniformly by a constant deviation (d) in the search space. Depending on the value of the zone factor (m) , the initial searching process is conducted optionally, according to the problem experience. If N is the number of the initial population in the solution space of the PUC problem, then the constant deviation between the initial points will be:

$$d = (u^H - u^L) / (mN), \quad (6)$$

$$m > 0.$$

The searching zone is varying according to the value of (m) . If $m > 1$ the searching zone will take a limited part of the solution space, if $m = 1$ the searching zone will be the complete space, and if $m < 1$ the zone will extend to include the margins of the space. In this case, the preferable value for (m) is 1. Then Equation (1) is to be modified to Equation (7) to give the new initialization of UDE algorithm. In addition, the three mutating Equations (2) to (4) are modified to Equations (8) to (10).

$$v_{i,j}^0 = u_{i,j}^L + (i / mN)(u_{i,j}^H - u_{i,j}^L). \quad (7)$$

$$v_{i,j}^C = v_{i,j}^p + F(v_{1,j}^p - v_{2,j}^p) / (mN). \quad (8)$$

$$v_{i,j}^C = v_{best(i,j)}^p + F(v_{1,j}^p - v_{2,j}^p) / (mN). \quad (9)$$

$$v_{i,j}^C = v_{i,j}^p + F(v_{best(i,j)}^p - v_{i,j}^p) / (mN) + F(v_{1,j}^p - v_{2,j}^p) / (mN). \quad (10)$$

3. The Case Study

3.1. System Model

In this paper, the PUC problem is solved through, a typical case study of 160 MW oil-fired drum-type boiler-turbine generating unit. Which is based on the third-order model of Bell-Astrom [19], as shown in Equations (11) to (13):

$$dP / dt = 0.9u_1 - 0.0018u_2 P^{9/8} - 0.15u_3. \quad (11)$$

$$dE / dt = ((0.73u_2 - 0.16) P^{9/8} - E) / 10. \quad (12)$$

$$d\rho_f / dt = (141u_3 - (1.1u_2 - 0.19)P) / 85. \quad (13)$$

Where P (kg/cm^2) is the drum steam pressure, ρ_f (kg/cm^3) is the fluid density, and u_1, u_2, u_3 are the valve positions of the fuel, flowing steam, and feeding water respectively. The water deviation is to be calculated as shown below:

$$q_e = (0.85u_2 - 0.14)P + 45.59u_1 - 2.51u_3 - 2.09. \quad (14)$$

$$\alpha_s = (1 / \rho_f - 0.0015) / (1 / (0.8P - 25.6) - 0.0015). \quad (15)$$

$$L = 50(0.13\rho_f + 60\alpha_s + 0.11q_e - 65.5). \quad (16)$$

Where α_s is the steam quality, q_e is the evaporation rate in (kg/s), and L is the drum water level output. Noting that, the positions of the valve actuators are constrained to $[0,1]$, and their rates of change (pu/s) are limited to:

$$-0.007 \leq du_1 / dt \leq 0.007. \quad (17)$$

$$-2.0 \leq du_2 / dt \leq 0.02. \quad (18)$$

$$-0.05 \leq du_3 / dt \leq 0.05. \quad (19)$$

3.2. Optimization Equation

For the case study, there will be four objective functions to be minimized: load tracking error, and the valve positions of the fuel, steam, and feeding water. Figure 1(a) shows the power-pressure window that gives the upper and lower boundaries of the pressure for each value of the generated power. Figure 1(b) shows the process of the three actuators during the load variations. These characteristics are known as the power-input windows, which are obtained from the power-pressure window according to Equations (20) to (22).

$$u_1 = (0.0018u_2P^{9/8} + 0.15u_3) / 0.9. \quad (20)$$

$$u_2 = (0.16 + E / P^{9/8}) / 0.73. \quad (21)$$

$$u_3 = (1.1u_2 - 0.19)P / 141. \quad (22)$$

These equations are deduced by solving the model of Bell-Astrom when the system is at its steady state. The multi-objective PUC problem can be described in Equation (23), as a typical minimization function, which is composed of four weighted terms of the optimized objectives. The weights are chosen according to the size of the solution spaces for the objective functions. As it is seen in Figure 1 (b), u_2 has the largest solution space so it can be weighted by 1, u_1 is weighted by 0.5 regards to its solution space, which is relatively smaller than the solution space of u_2 . The solution space of u_3 is so small, thus u_3 is weighted by 0. The error of the generated power is weighted by 1 due to its wide range of optimization.

$$\min F = 1.0 \sum_{i=1}^{SP} |E_{uld}^i - E_{ss}^i| + 0.5 \sum_{i=1}^{SP} u_1^i + 1.0 \sum_{i=1}^{SP} -u_2^i + 0.0 \sum_{i=1}^{SP} u_3^i. \quad (23)$$

$$s.t. \quad u \in |(u_j^i)_L, (u_j^i)_H|,$$

$$i = 1, 2, 3, \dots, SP; \quad j = 1, 2, 3.$$

For a set point (SP) i , E_{load}^i is the load demand, E_{ss}^i is the actual generated power. u_1 represents the valve position of the fuel injection, u_2 represents the valve position of the steam flow (it is predicted by a negative sign to convert the maximization into minimization process), u_3 represents the valve position of feeding water. $(u_j^i)_L$ and $(u_j^i)_H$ are the lower and higher boundaries of the solution space.

3.3. Problem Description

Figure 2 shows a summarized structure for the coordinated control scheme as in [8], which is required to create the optimal operating map for the power unit according to the load demand. By aid of the power-pressure and power-input windows the power-pressure mapping and the input signals can be derived. Consequently, a vector of optimal control signals (u^*) is generated to determine a pressure set-point (P_d) according to the steady-state model (M_{ss}) of the power unit as in Equation (24). Then a successful searching mechanism for the mentioned optimal input vector is developed to solve this multi-objective problem.

$$P_d = M_{ss}(u^*) \tag{24}$$

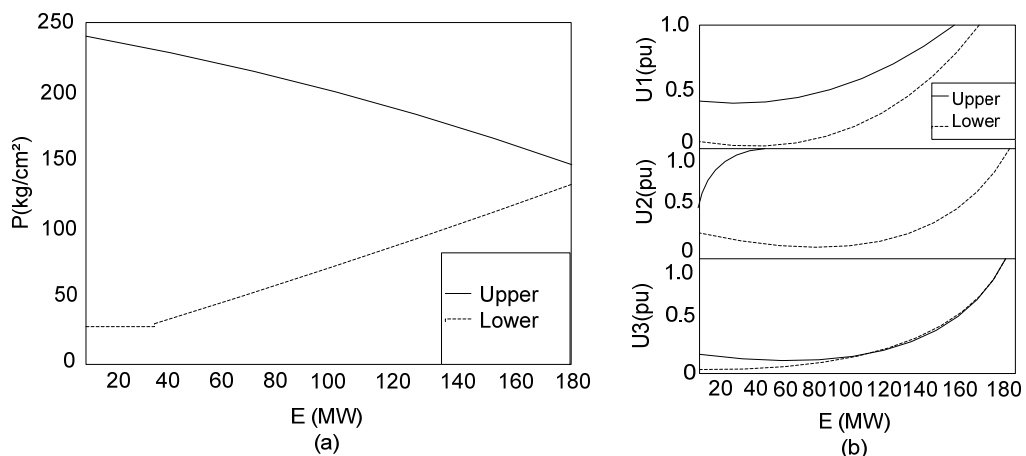


Figure 1. (a) The power-pressure window, (b) The power-input windows

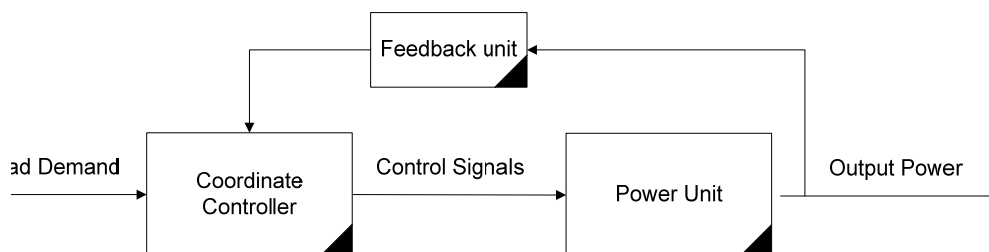


Figure 2. Coordinated control structure

3.3 Algorithms Application

DE and UDE algorithms are employed to explore for the optimum set of input vector $U^* = [U_1, U_2, U_3]^T$ to satisfy the optimization process according to the flowchart in Figure 3, with the following steps:

Step 1: initialize the first population according to Equations (1) and (7).

Step 2: evaluate each individual by a fitness value.

Step 3: explore a new population (recombination) by mutation operator according to Equations (2) to (4) and (8) to (10), taking into account the crossover probability according to Equation (5),

Step 4: evaluate the new population and give a fitness for each individual.

Step 5: select the best individual according to the best fitness relative to other individuals.

Step 6: update the best individual from the current population.

Step 7: If the criterion is satisfied, terminate the algorithm and output the final solution, otherwise go to step 3.

4. Research Method

Four simulation experiments were developed by aid of software programs using MATLAB (version 7.10.0), to conduct DE and UDE algorithms to solve the PUC problem. Thirty independent pilot runs were use in each experiment. It is necessary to determine the optimal parameter setting for the basic DE and the proposed UDE algorithms empirically. Figure 4 gives the fitness variations against the mutation factor. According to this investigation, the optimal mutation factor is chosen (0.8 for the two algorithms).

4.1. First Experiment

The first experiment was developed to show the improvement of the modified UDE relative to the basic DE algorithm in solving the PUC problem. Table 1 shows the optimal parameter-setting of the algorithms. Then, the results of this experiment are shown in Table 2.

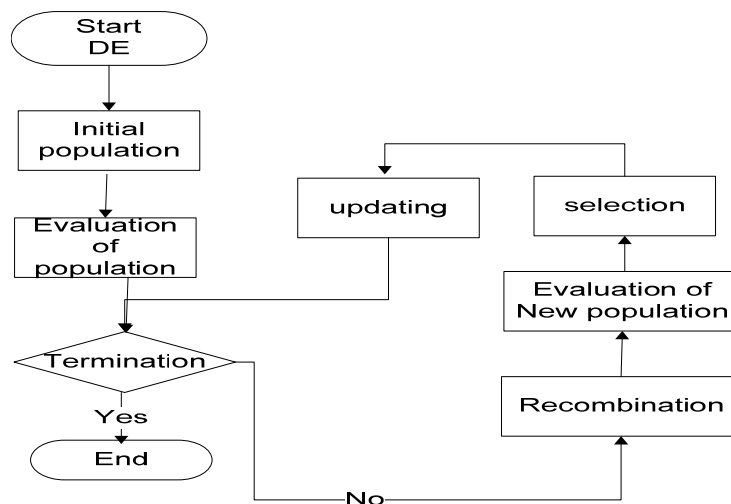


Figure 3. The flowchart of DE and UDE algorithms

4.2. Second Experiment

It is necessary to evaluate the performance of the proposed UDE algorithm with respect to other heuristic algorithms. Thus, a second experiment was made to compare UDE algorithm with GA, DE and PSO algorithms. Table 3 shows the optimal parameter setting, while Table 4 shows the experiment results.

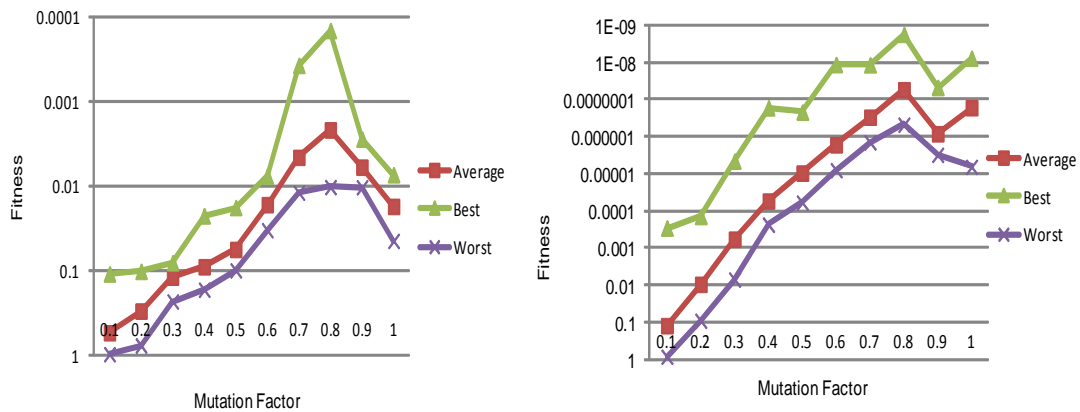


Figure 4. The parameter setting investigation, the left graph is for DE and the right graph is for UDE

Table 1. The parameter setting of DE and UDE

Algorithm	Mutation factor(F)	Crossover factor(f)	Max. Gen.	Pop. Size
DE	0.8	0.5	500	60
UDE	0.8	0.5	200	20

Table 2. Simulation results of comparing the three schemes of DE and UDE

Algorithm	Theoretical	Average	Best	Worst	Std
DE1	0	5.88E-03	1.97E-04	0.0301	0.00653
UDE1	0	2.61E-03	1.84E-04	0.0078	0.002425
Improvement (%)	-	55.61%	6.60%	74.09%	62.86%
DE2	0	1.52E-03	8.30E-07	0.0114	0.00262379
UDE2	0	5.43E-08	1.15E-09	2.10E-07	5.01E-08
Improvement (%)	-	99.98%	99.86%	99.97%	100%
DE3	0	1.77E-03	3.12E-05	0.0083	0.00195
UDE3	0	3.57E-08	2.17E-010	1.53E-07	3.62E-08
Improvement (%)	-	100%	100%	100%	100%

4.3. Third Experiment

As UDE and PSO algorithms have comparable results, more investigations are required to develop a comparison between them. A third experiment was made by tuning the maximum generation and population size using the parameter setting in Table 3. The results of this experiment are shown in Figure (5).

4.4. Fourth Experiment

Through this experiment the optimization Equation (23) is conducted to show the effect of the comparative algorithms on the minimization process, through the PUCG problem using the same parameter-setting in Table 3. Table 5 presents the results of the minimization process.

Table 3. The parameter setting of the comparative algorithms

Algorithm	Mutation factor (F)	Crossover factor (f)	Learning factor (c ₁)	Learning factor (c ₂)	Weight (w)	Max. Gen.	Pop. Size
GA	0.1	0.9	-	-	-	500	100
DE	0.8	0.5	-	-	-	500	60
PSO	-	-	2.1	2.0	0.8	500	40
UDE	0.8	0.5	-	-	-	200	20

Table 4. Simulation results of the comparative algorithms

Algorithm	Theoretical	Average	Best	Worst	Std
GA	0	0.057391	4.86E-04	0.20950	0.16318
DE	0	1.77E-03	3.12E-05	0.0083	0.00195
PSO	0	2.61E-07	8.53E-13	6.15E-06	1.10E-06
UDE	0	3.57E-08	2.17E-10	1.53E-07	3.61951E-08

Table 5. Optimization results

Algorithm	Average of u_1	Average of u_2	Average of u_3	Average of error	Optimization results
GA	0.6946207	0.89316	0.65276	0.01847	1.25894035
DE	0.73107096	0.883819	0.614268	0.0025	1.251854
PSO	0.556226	0.899677	0.71456516	6.87E-07	1.177790687
UDE	0.5566	0.8459	0.6771	3.80E-08	1.124200038

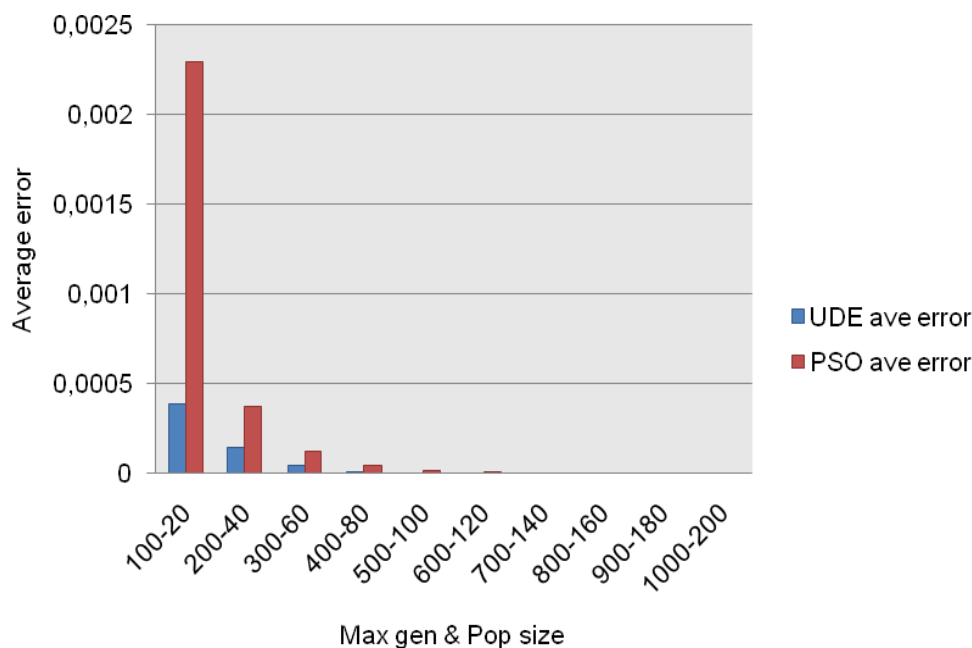


Figure 5. The Average error versus max gen & pop size for PSO & UDE algorithms

4.5. Comparing the Proposed Algorithm with a similar work in the literature

In [4], DE algorithm was improved by developing a new fast differential evolution (FDE) algorithm to solve the PUC problem. The FDE algorithm has a dynamic mutation factor instead of the fixed one of the basic DE algorithm. A comparison through this work and our proposed UDE algorithm is developed to evaluate the effects on the basic DE algorithm. Table 6 shows the percentage improvement of FDE and UDE on the basic DE algorithm among the three mentioned schemes of DE algorithm. Through which our proposed algorithm has better improvement than FDE algorithm.

Table 6. The percentage improvement of FDE & UDE on DE algorithm

Algorithm	Percentage improvement
FDE1	3.96%
UDE1	55.61%
FDE2	2.10%
UDE2	99.98%
FDE3	2.80%
UDE3	100%

4.6. Results Analyses

As shown in Table 2, UDE algorithm has improved the performance of DE algorithm due to its effective search mechanism. Table 4 proves that UDE algorithm can highly outperform GA and DE algorithm. When comparing the performance of UDE and PSO algorithms, the first has been slightly better than the latter. Noting that, UDE algorithm can give the best results at a small maximum generation number of individuals (about 40% of the maximum generation size that is used by PSO algorithm).

The UDE algorithm is easier in application than GA and PSO algorithm with regard to its simple structure. Figure 5 demonstrates the effect of tuning the maximum generation and population size on the average error for UDE and PSO algorithm. It is clear that the average error decreases as the maximum generation increases for the two algorithms. UDE algorithm can give the best results early, rather than PSO algorithm.

Table 5 shows the typical results of the minimization process when conducting equation (23). The first four columns of this table are the average values of the four objective functions that are to be minimized. In which UDE algorithm has better performance than GA and DE algorithm, and has comparable results with PSO algorithm. The fifth column gives the summation of the four objective values according to the weighted optimization equation, in which UDE has the best value overall the comparative algorithms. Table 6 proves that UDE has higher percentage improvement on the basic DE than FDE algorithm through the three schemes.

5. Conclusion

According to the results shown above we conclude that the arrangement of UDE algorithm can improve the performance of the basic DE algorithm. Nevertheless, slight alteration on the basic DE is required, so UDE algorithm is still simple and easy to be applied for solving multi-objective PUCC problem. Empirically, UDE algorithm can give better results than that of GA, and DE algorithms. On the other hand, UDE and PSO algorithms have quite comparable results as the first has slight best results than the latter. Noting that, UDE has the advantage of using small maximum generation of individuals over other comparative algorithms that reduces its computation cost. Furthermore, our proposed algorithm has better improvement on the basic DE algorithm than another similar work in the literature

Acknowledgements

The authors would like to thank the National Science Foundation of China (Grant Number 61203100 and Grant Number 61273144) and The Science Foundation of Hebei Province (Grant Number F2010001714) for their financial supports with this research.

References

- [1] Metcalfe M, J Waddington, R Wilson. Operational experience with fossil-fired generating plant under automatic load control. *Electrical Engineers*. 1979; 126(4): 327-332.
- [2] Garduno-Ramirez R, KY Lee. Multiobjective optimal power plant operation through coordinate control with pressure set point scheduling. *IEEE Transaction on Energy Conversion*. 2001; 16(2): 115-122.
- [3] Heo JS, KY Lee, Garduno-Ramirez R. Multiobjective control of power plants using particle swarm optimization techniques. *IEEE Transaction on Energy Conversion*. 2006; 21(2): 552-561.
- [4] Van Sickle JH, KY Lee, JS Heo. *Differential evolution and its applications to power plant control*. IEEE Proceedings on Intelligent Systems Applications to Power Systems (ISAP). Kaohsiun, Taiwan. 2007; 14: 1-6.
- [5] Van Sickle JH, P Venkatesh, KY Lee. *Analysis of the pareto front of a multi-objective optimization problem for a fossil fuel power plant*. IEEE Proceedings on Power and Energy Society General Meeting-Conversion and Delivery of Electrical Energy in the 21st Century. Pittsburgh. 2008: 1-8.
- [6] Head JD, Gomes JR, Williams CS, Lee K.. *Implementation of a multi-agent system for optimized multiobjective power plant control*. IEEE Symposium on North American Power (NAPS). Arlington. 2010: 1-7.
- [7] Lee KY, Head JD, Gomes JR, Williams CS. *Multi-agent system based intelligent distributed control system for power plants*. IEEE Proceedings on Power and Energy Society General Meeting. Detroit. 2011: 1-7.

- [8] Liu X, P Guan, C Chan. Nonlinear Multivariable Power Plant Coordinate Control by Constrained Predictive Scheme. *IEEE Transaction on Control Systems Technology*. 2010; 18(5): 1116-1125.
- [9] Zhongxu H, Zhi Z., Xiao Juan H, Ying S. *The simulation research of model predicting control (MPC) arithmetic of unit plant coordinated control system*. IEEE Proceedings on Power System Technology. Kunming, China. 2002; 3: 1518-1521.
- [10] Price KV. *An introduction to differential evolution*. Proceedings of McGraw-Hill on New ideas in optimization. UK.1999: 79-108.
- [11] Storn R, K Price. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*. 1997; 11(4): 341-359.
- [12] Slimani Linda, Tarek Bouktir. Optimal Power Flow Solution of the Algerian Electrical Network using Differential Evolution Algorithm. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2012; 10(2): 199-210.
- [13] Tasgetiren MF, P Suganthan. A multi-populated differential evolution algorithm for solving constrained optimization problem. IEEE Proceedings on Evolutionary Computation (CEC). Vancouver. 2006: 33-40.
- [14] Madavan N.K. Multiobjective optimization using a Pareto differential evolution approach. IEEE Proceedings On Evolutionary Computation (CEC). Honolulu. 2002; 2: 1145-1150.
- [15] G Jian, L Xu, Y Chen, X Chen, D Zheng. Precise Pose Alignment through the Combined Method of Differential Evolution and Simplex Algorithm. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(3): 1600-1608.
- [16] Sun C, H Zhou, L Chen. *Improved differential evolution algorithms*. IEEE Proceedinon Computer Science and Automation Engineering (CSAE). Zhangjiajie, China. 2012; 3: 142-145.
- [17] Whitley D. A genetic algorithm tutorial. *Statistics and computing*. 1994. 4(2): 65-85.
- [18] Kennedy J, R Eberhart. *Particle swarm optimization*. IEEE Proceedings on Neural Networks. 1995; 4: 1942-1948.
- [19] Bell R, KJ Åström. Dynamic models for boiler-turbine-alternator units: data logs and parameter estimation for a 160 MW unit. *Electrical Engineers*. 1987; 126(4): 327-332.