Optimal reactive power dispatch (ORPD) using particle swarm optimization based on individual difference evolution algorithm (IDE-PSO)

Cite as: AIP Conference Proceedings **2543**, 040022 (2022); https://doi.org/10.1063/5.0095113
Published Online: 16 November 2022

Muhammad Ridhwan, Yusran and Indar Chaerah Gunadin

Optimal Reactive Power Dispatch (ORPD) using Particle Swarm Optimization based on Individual Difference Evolution Algorithm (IDE-PSO)

Muhammad Ridhwan[1,a)], Yusran[2,b)], and Indar Chaerah Gunadin[2,c)]

[1]*Power Electrical Engineering Research Group, Department of Electrical Engineering, Universitas Hasanuddin, Indonesia*
[2]*Department of Electrical Engineering, Universitas Hasanuddin, Gowa, Indonesia*

[a)]ridhwan.eng@gmail.com
[b)]Corresponding author: yusran@unhas.ac.id
[c)]indarcg@gmail.com

Abstract. The Optimal Reactive Power Dispatch (ORPD) is one of a non-linear complex optimization problem. The main purpose of ORPD is to obtain minimum power loss by control variable system adjusting in all system limitation requirements. The control variables that need to be optimized are the value of bus generator voltage, tap transformer settings and shunt reactive elements. The Particle Swarm Optimization (PSO) has developed in several new variants. In this research, the PSO algorithm was improved using Individual Different Evolution (IDE-PSO) algorithm. This algorithm was conducted to solve ORPD problem on IEEE 30 bus test system as the research object. The simulation results show that the total value of active and reactive power losses decreased by 1.39% and 1.95% respectively, comparing with the system initial condition. The IDE-PSO algorithm results also were compared with the Standard PSO results. The IDE-PSO algorithm proved to be more effective than the Standard PSO algorithm for ORPD problem.

INTRODUCTION

Optimal Reactive Power Dispatch (ORPD) greatly affects the optimal power system operation. The reactive power optimization is a sub problem of the optimal power-flow (OPF) calculation, which determines all kinds of controllable variables, such as reactive-power outputs of generators and static reactive power compensators, transformer taps, shunt capacitors, and minimize transmission loss or other suitable objective functions, as well as complement a specific physical and operation series [1].

Since the ratio of the tap transformer and the output of the shunt capacitor has discrete properties, while the reactive power of the generator, the magnitude of the bus voltage, and the angle are, on the other hand, continuous variables, the reactive power optimization problem can be exactly formulated using a mixed-integer/nonlinear programming model, cast as a nonlinear optimization problem with a mixture of discrete and continuous variables [2]. The supply of active power and reactive power by the power plant to the load greatly affects the voltage drop. In conventional systems, the voltage is usually controlled by an on-load tap changer and a capacitor bank [3,4]. Gradient-based algorithms and Newton's method have been applied to solve this problem but these methods cannot solve non-linear, discrete-continuous problems.

Now many new metaheuristic optimization techniques have been applied to solve optimization problems. This method has been developed to overcome the main drawbacks of conventional methods and has been successfully applied to solve ORPD problems such as particle swarm optimization (PSO) dan genetic algorithm (GA). The GA method is a searching technique, based on natural selection and genetic mechanism This technique is used on practical problems, with a focus on finding optimal parameters, with ease of implementation, and the ability to quickly find good solutions [3,4]. The other methods are the ant colony algorithm (ACO) and gravitational search algorithm (GSA) etc. [5].

The Proceedings of the 4th EPI International Conference on Science and Engineering (EICSE) 2020
AIP Conf. Proc. 2543, 040022-1–040022-8; https://doi.org/10.1063/5.0095113
Published by AIP Publishing. 978-0-7354-4358-7/$30.00

The algorithm has been improved like PSO by combining it with other algorithms. As a stochastic optimization algorithm, particle swarm optimization (PSO) has attracted a lot of attention from researchers around the world, which has produced many variants of the basic algorithm, mostly the selection of parameters/strategy control. However, most of these algorithms expand their population using a single, fixed pattern, thus reducing the intelligence of the entire herd.

Some PSO variants do not have dynamic adaptability even though they have adopted a multimode evolution strategy. Furthermore, competition between particles is ignored, without considering the ability to think or make individual decisions, while the improved PSO algorithm based on individual difference evolution (IDE-PSO) This algorithm allocates a competition coefficient called emotional status to each particle to measure individual differences, separating the entire group into three subgroups, and selecting specific evolutionary methods for each particle according to their emotional status and current fitness. The coefficient values are adjusted dynamically according to the evolutionary performance of each particle [6].

OPTIMAL REACTIVE POWER DISPATCH (ORPD)

The Optimal Reactive Power Dispatch (ORPD) has an important impact on reducing the power loss from the transmission line and adjusting for voltage changes. The parameters used for ORPD are the transformer tap ratio and the reactive power output of the shunt compensators such as capacitors etc. Several conventional mathematical models and techniques such as gradient-based algorithms and Newton's method have been applied to solve this problem. Unfortunately, this method has errors in malfunction and non-linear, discrete-continuous errors [6].

The purpose of sending reactive power is to minimize the loss of active power in the transmission network, which is described as follows:

$$\min \sum_{keNE} P_{kloss} = \sum_{keNE} g_k \left(v_i^2 + v_j^2 - s v_i v_j \theta_{ij} \right) \tag{1}$$

where:

$k = (i, j); i \in N_B$: total number of buses

$j \in N_i$: number of buses adjustment to bus *i,* including bus *i*

$\sum_{k \in N_E} P_{kloss}$: total active power loses at transmission system

g_k : conductance of branch *k* (pu)

v_i, v_j : voltage magnitude (pu) of bus *i* and *j*

θ_{ij} : load angle difference between bus *i* and *j* (rad)

Equality constraints and Inequality constraints are stated as follows.

Equality Constraints

Active power flow balance equations at all buses excluding slack bus

$$P_{Gi} - P_{Di} = V_i \sum_{JeNi} V_J \left(G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij} \right) \tag{2}$$

Reactive power flow balance equations at all PQ buses (load buses)

$$Q_{Gi} - Q_{Di} = V_i \sum_{JeNi} V_J \left(G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij} \right) \tag{3}$$

where:

P_{Gi} and Q_{Gi} : active and reactive power generator on the bus *i*

P_{Di} and Q_{Di} : active and reactive load on bus *i*

V_i and V_j : the amount of voltage on buses *i* and *j*,

G_{ij} and B_{ij} : the conductance and susceptance on the transmission line

θ_{ij} : the angular difference from the *i*-*j* transmission line

Inequality Constraints

Reactive power generation limit

$$Q_{gi}^{\min} \le Q_{gi} \le Q_{gi}^{\max}, i \in N_B \tag{4}$$

Voltage magnitude limit for each bus

$$v_i^{\min} \le v_i \le v_i^{\max}, i \in N_B \tag{5}$$

Transformer tap setting

$$T_k^{\min} \le Tk \le T_k^{\max} \tag{6}$$

Power flow limit constraint of each transmission line

$$S_i \le S_i^{\max} \tag{7}$$

The static square penalty function is used to handle inequality constraints. So the augmented objective function (fitness function).

$$F_p = \sum\nolimits_{k \in N_E} P_{kloss} + Penalty\ Function \tag{8}$$

where,

Penalty Function =

$$k_1 \times \sum\nolimits_{i-1}^{N_C} f\left(Q_{gi}\right) + k_2 \times \sum\nolimits_{i-1}^{N} f\left(V_i\right) + k_3 \times \sum\nolimits_{i-1}^{N_L} f\left(S_{im}\right)$$

$$k_1, k_2, k_3 = 10.000$$

$$f(x) = \begin{Bmatrix} 0 & if\ x^{\min} \le x \le x^{\max} \\ \left(x - x^{\max}\right)^2 & if \quad x > x^{\max} \\ \left(x^{\min} - x\right)^2 & if \quad x < x^{\min} \end{Bmatrix}$$

INDIVIDUAL DIFFERENCE EVOLUTION PARTICLE SWARM OPTIMIZATION (IDE-PSO) ALGORITHM

The Particle Swarm Optimization (PSO) algorithm is a member of the wide category of swarm intelligence methods for solving global optimization problems. Originally, the PSO was introduced by Kennedy in 1995, as a social behavior simulation, as an optimization method. The PSO is related with artificial life, and specifically to swarming theories, and also with evolutionary computation, especially evolutionary strategies and genetic algorithm [7,8]. Each particle updates its position based upon its own best position, global best position among particles and its previous velocity vector according to the following equations:

$$v_i^{k+1} = w \times v_i^k + c_1 \times r_1 \times \left(P_{best} - x_i^K\right) + c_2 \times r_2 \times \left(G_{best} - x_i^K\right) \tag{9}$$

$$x_i^{k+1} = x_i^{k+1} + X \times v_i^{k+1} \tag{10}$$

where

v_i^{k+1} : the velocity of i^{th} particle at $(k+1)^{th}$ iteration
w : inertia weight of the particle
v_i^k : the velocity of i^{th} particle at k^{th} iteration
$c_{1,}c_2$: positive constants having values between [0, 2.5]
r_1, r_2 : randomly generated numbers between [0, 1]
p_{best_i} : the best position of the i^{th} particle obtained based upon its own experience
g_{best} : global best position of the particle in the population
x_i^{k+1} : the position of i^{th} particle at $(k+1)^{th}$ iteration
x_i^k : the position of i^{th} particle at k^{th} iteration
χ : constriction factor

Selection of suitable inertia weights provides a good balance between global and local exploration.

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{iter_{\max}} \times iter \tag{11}$$

where

$w_{\max}$: the value of inertia weight at the beginning of iterations,
$w_{\min}$: the value of inertia weight at the end of iterations,
$iter$: the current iteration numbers
$iter_{\max}$: the maximum number of iterations

The PSO algorithm is improved by combining 2 algorithms which are commonly called hybrids [9]. This paper uses the Particle Swarm Optimization algorithm based on Individual Difference Evolution (IDE-PSO), which is an algorithm that uses the new parameter eX as a direct measure of the emotional states of particles to guide particle movement. following the scheme that social status affects a person's emotions, and these emotions guide individual actions and can be described as follows:

Social status → Emotion of Individual → Action of Individual

The equivalent form of this scheme in PSO is:

Fitness ($f(X_i)$) → Particle's Emotion (*eXi*) → Particle's Motion (V_i *and* X_i).

The PSO algorithm is improved based on the IDE. In addition, a modified strategy was used to increase exploration and herd diversity. accuracy and stability, and generalization algorithms for solving multi-optima and multi-objective optimization problems [10].

In the proposed evolutionary mechanism, the emotional state of each particle is adaptively adjusted according to its suitability in each iteration. If the fitness increases compared to the previous iteration, the emotional state of the particles increases. Conversely, decreased fitness causes a decrease in emotional status. We define three emotional states for particles: weak, normal, and good. Thus, based on the current performance of the particles, the entire population is dynamically divided into three subgroups.

The ascending order of the particles *eX*,

$$eX_{weak} \rightarrow eX_{normal} \rightarrow eX_{good}$$

each iteration [11].

MECHANISM OF IDE

Weak Particle

Particles with weak emotional states usually have their fitness reduced (or only slightly increased) over successive iterations. Such particles are more likely to be moving in bad directions or becoming stuck around the local optima, meaning that their positions and velocities are of low value. According to the psychology model, people with a poor social status are on the verge of collapse. Weak particles should abandon their current personal information and learn from the best particles in the swarm.

We build a restarting strategy to solve the problem of weak particles. The velocity of a weak particle is reinitialized by:

$$V_i^t\left(j\right)=0.2\left(u-l\right)\cdot rand \tag{12}$$

The position of a weak particle is initialized at the dimensional level: values of each dimension have two options, random reinitialization or learning the value of the global best particle in the swarm.

$$\begin{aligned}&if\ \ rand<0.5\\&\qquad X_i^t\left(j\right)=gb^{t-1}\left(j\right)\\&\quad Else\\&\qquad X_i^t\left(j\right)=l+\left(u-l\right)\cdot rand\left[1,1\right]\\&End\ \ if\end{aligned} \tag{13}$$

Compared to a simple restarting strategy, this approach combines random initialization with information about the global best particle, and is applied based on the particle's dimension. Thus, we obtain a compromise between convergence and population diversity.

Normal Particle

One particle with a normal emotional status learns from both its individual and social experience, which is a relatively stable evolutionary method. Here, the normal motion strategy is used:

$$v_i^{k+1}=x\times v_i^k+c_1\times r_1\times\left(P_{best}-x_i^K\right)+c_2\times r_2\times\left(P_{best}-x_i^K\right) \tag{14}$$

$$x_i^{k+1}=x_i^{k+1}+X\times x_i^{k+1} \tag{15}$$

Good Particles

When a particle has a good emotional status, it becomes "confident," which indicates its fitness has been considerably improved for a certain number of generations. This means that the particle moves in a good direction, and retains its current tendency (velocity and position) regardless of social information. The perturbation term 0.2 * (u - l) * rand, which is the same as the normal initialization of velocity, is added to the velocity operator Eq. (17) to enhance the exploitative ability of good particles [11]. The moving strategy can be described as:

$$v_i^{k+1}=w\times v_i^k+w\times 0.2\times\left(u-l\right)\times rand \tag{16}$$

$$x_i^{k+1}=x_i^{k+1}+v_i^{k+1} \tag{17}$$

RESULT AND DISCUSSION

IEEE-30 bus system is used where this system consists of six generators, 41 transmission lines, four transformers and three injected shunt reactive elements located at buses 3, 10 and 24. To test the effectiveness of the IDE-PSO algorithm by comparing the standard PSO algorithm to solve the ORPD problem, the two algorithms are set with a maximum number of iterations of 200.

Table 1 shows the comparison of losses obtained before and after optimization using the standard PSO algorithm and the IDE-PSO algorithm. IDE-PSO is a PSO algorithm that has been developed in a hybrid way using the base individual difference evolution algorithm and it can be seen that the results of the developed PSO algorithm (IDE-PSO) get 17.312 MW losses. It was lower compared to standard PSO.

TABLE 1. Comparison of losses obtained before and after optimization with PSO and IDE-PSO

No	Losses	Before optimization	After optimization with PSO	After optimization with IDE-PSO
1	Active losses (MW)	17.557	17.3579	17.312
2	Reactive losses(MVAr)	67.69	67.00	66.37

TABLE 2. Values of control variables after optimization with IDE-PSO

Bus	Control variables	Optimized values (xmin)
1	Vg1 (pu)	1.062
2	Vg2	1.047
5	Vg5	1.014
8	Vg8	1.021
11	Vg11	1.036
13	Vg13	1.060
6-9	T1 (pu)	0.909
6-10	T2	1.029
4-12	T3	0.961
27-28	T4	0.948
3	QC3 (MVAr)	10.822
10	QC10	7.731
24	QC24	12.129

Analysis with the effect of control variable setting for the IEEE 30 bus system, even though the minimum losses is obtained, the allowance for each bus should be operated within its limits where for the detailed results of the voltage values for all buses tabulated in Table 3. It can be seen that all the voltage values operate at the specified limit.

TABLE 3. Voltage magnitude results for IEEE 30 bus system

No. Bus	Before	PSO	IDE-PSO
1	1.060	1.062	1.062
2	1.045	1.044	1.047
3	1.021	1.036	1.039
4	1.012	1.022	1.027
5	1.010	1.011	1.014
6	1.011	1.016	1.020
7	1.003	1.006	1.010
8	1.010	1.014	1.021
9	1.051	1.021	1.034
10	1.045	1.018	1.036
11	1.082	1.038	1.036
12	1.057	1.026	1.042
13	1.071	1.040	1.060
14	1.043	1.013	1.029
15	1.038	1.010	1.027
16	1.045	1.015	1.032
17	1.040	1.012	1.029
18	1.028	1.000	1.018
19	1.026	0.998	1.015
20	1.030	1.002	1.020
21	1.033	1.008	1.027
22	1.034	1.010	1.028
23	1.027	1.007	1.025
24	1.022	1.011	1.031
25	1.018	1.007	1.026
26	1.000	0.989	1.008
27	1.024	1.012	1.032
28	1.007	1.012	1.017
29	1.004	0.992	1.012
30	0.992	0.981	1.001

CONCLUSION

In this paper, the IDE-PSO algorithm is proposed. An algorithm is a PSO algorithm developed based on Individual Difference Evolution. The particles in PSO are grouped into 3 sub-sections (weak, normal and good). The IDE-PSO was conducted to solve ORPD problems on IEEE 30 bus system. The results show that IDE-PSO algorithm is more effective than the standard PSO algorithm.

REFERENCES

1. B. Zhao, C.X. Guo, and Y.J. Cao, IEEE Trans. Power Syst. **20**, (2005).
2. A. Amin, S. Kamel, and M. Ebeed, in *Eighteenth Int. Middle East Power Syst. Conf.* (2016).
3. Y. Yusran, Y.A. Rahman, I.C. Gunadin, S.M. Said, and S. Syafaruddin, Przegląd Elektrotechniczny **1**, (2020).

4. Yusran, in *4th Makassar Int. Conf. Electr. Eng. Informatics* (2014).
5. A. Khorsandi, A. Alimardani, B. Vahidi, and S.H. Hosseinian, Transm **5**, 249–256 (2011).
6. S.M. Sadeghzadeh, A. Khazali, and S. Zare, in *IEEE EUROCON* (2009).
7. D. Kaur, T.T. Lie, N.K.C. Nair, and B. Valles, in *IEEE Int. Conf. Sustain. Energy Technol.* (2016).
8. C. Tanti and D. Pipalava, Int. J. Sci. Res. (2013).
9. H. Cao, J. Xu, D. Ke, C. Jin, S. Deng, C. Tang, and M. Cuim, in *North Am. Power Symp.* (2016).
10. Y. Jiang, T. Hu, C. Chao, and H.X. Wu, Appl. Math. Comput. **193**, 231–239 (2007).
11. J. Gou, Y.X. Lei, W.P. Guo, C. Wang, Y.Q. Cai, and W. Luo, Appl. Soft Comput. (2017).