Research on Reactive Power Optimization of Power System Based on Improved Particle Swarm Algorithm

Kang Li¹*, Guige Gao¹

¹School of Electric Engineering, Shanghai DianJi Universitye, Shanghai, China
*Corresponding author’s e-mail:1074421240@st.sdju.edu.cn

Abstract. Artificial intelligence algorithms are widely used to optimize problems in power systems, and reactive power optimization in power systems has achieved good results in particle swarm optimization, but there are also problems. This paper optimizes the particle swarm algorithm. The particle swarm algorithm is improved mainly by increasing the inertia weight and improving the convergence parameters. This algorithm overcomes the blindness of local optimization solution and particle swarm algorithm, and improves the calculation speed. At the same time, MATLAB is used to compile the calculation program, and the simulation results are used to verify the feasibility of the reactive power optimization algorithm used in the research of power system.

1. Introduction
At present, various artificial intelligence algorithms have been applied to optimize the reactive power of the power system [1-3], and have shown good results. These include Particle Swarm optimization[4-5] (Particle Swarm optimization, PSO), a survey of bird predation behavior, and the invented evolutionary calculation method [6-7]. The algorithm has fast convergence speed, but it has the disadvantages of easy local convergence and poor convergence accuracy [8]. This is because in the pursuit of the best effect, all particles aim at the best particle and search in the same direction, which causes convergence of the particles in the later stage, and the ability to search for unknown areas is lost. Therefore, the parameters of the basic particle swarm algorithm need to be improved [9-10].

This article solves the problem of optimizing the power system by increasing the inertia weight and improving the convergence parameters. The improved particle swarm algorithm has the following advantages: (1) It has global convergence. (2) Can not meet the requirements of differentiation and continuity. The most important thing is that it has a global optimization function and is easy to program to achieve reactive power optimization in actual power transmission.

2. Mathematical Model of Reactive Power Optimization
The mathematical model consists of the following parts: objective function, variable constraints and power constraints. Usually, adjust terminal voltage and reactive power compensation capacity under appropriate conditions. Reduce network loss under the requirements of ensuring reliable voltage quality. For economic reasons, the active network loss is usually used as the objective function:

$$\min (P_{\text{loss}}) = \sum_{\alpha = 1}^{n} G_{\alpha} \left( V_{\alpha}^{2} + V_{\alpha}^{2} \angle \cos \delta_{\alpha} \right)$$

In (1): In the calculation of active power loss, the number of branches of the system is set to O, G_{\alpha} is the conductance between node m and branch n; V_{\alpha} is the voltage at node m, and V_{\alpha} is the voltage at node n. State variables include the reactive power output of the generator and the voltage value of each n-ode. The constraints of each variable are divided into inequality constraints and equality constraints. The equation constraint is the power flow equation of active power and reactive power [7], as in equation (2):

$$\begin{align*}
    P_{\alpha} &= V_{\alpha} \sum_{\gamma = 1}^{n} V_{\gamma} (G_{\alpha\gamma} \cos \delta_{\alpha\gamma} + B_{\alpha\gamma} \sin \delta_{\alpha\gamma}) \\
    Q_{\alpha} &= V_{\alpha} \sum_{\gamma = 1}^{n} V_{\gamma} (G_{\alpha\gamma} \sin \delta_{\alpha\gamma} + B_{\alpha\gamma} \cos \delta_{\alpha\gamma})
\end{align*}$$

In (2): the active power injected at node m is P_{\alpha}, the reactive power is Q_{\alpha}, and the voltage is V_{\alpha}; the conductance between m and n is G_{\alpha\gamma}, the electrical susceptance is B_{\alpha\gamma}, and the phase angle difference is \delta_{\alpha\gamma}; the total number of nodes is set to OB. The inequality constraints are as in formula (3):

$$\begin{align*}
    T_{\text{mmin}} &< T_{\alpha} < T_{\text{mmax}} \\
    C_{\text{nmin}} &< C_{\alpha} < C_{\text{nmax}} \\
    V_{\text{fmin}} &< V_{\alpha} < V_{\text{fmax}}
\end{align*}$$

In (3): T_{\alpha} is the adjustable transformer ratio, T_{\text{mmax}} is its upper limit, and T_{\text{mmin}} is its lower limit; C_{\alpha} is the compensation amount of node n, C_{\text{nmax}} is its upper limit, and C_{\text{nmin}} is its lower limit; V_{\alpha} is the voltage value of the generator node has an upper limit value of V_{\text{fmax}} and a lower limit value of V_{\text{fmin}}. Inequality constraint formula:

$$\begin{align*}
    V_{\text{mmax}} &< V_{\alpha} < V_{\text{mmax}} \\
    Q_{\text{gmax}} &< Q_{\alpha} < Q_{\text{gmax}}
\end{align*}$$

In (4): V_{\alpha} is the node voltage, V_{\text{mmax}} is the upper limit, V_{\text{mmin}} is the lower limit; Q_{\alpha} is the reactive power output of the engine node, Q_{\text{max}} is the upper limit, and Q_{\text{min}} is the lower limit. When artificial intelligence algorithms are used to solve optimization problems with constraints, inequality constraints are generally handled by penalty functions. Therefore, when artificial intelligence methods solve constrained optimization problems, the penalty function is usually used to deal with the inequality constraints of the problem. Therefore, in this article, the penalty function is used to deal with the over-limit of the node voltage and the over-limit of the reactive power output of the generator (that is, the limit of state variables)

$$\begin{align*}
    \min f &= P_{\text{loss}} + \lambda_{1} \sum_{\alpha = 1}^{n} (V_{\alpha} - V_{\text{lim}})^{2} + \\
    &+ \lambda_{2} \sum_{\alpha = 1}^{n} (Q_{\alpha} - Q_{\text{lim}})^{2}
\end{align*}$$
In (4): $P_{\text{loss}}$ is the active power loss obtained in the system calculation; $\gamma_1$ is the node voltage amplitude exceeding the limit, $\gamma_2$ is the penalty factor for the generator reactive power output exceeding the limit, the mathematical expression is: $\gamma = \gamma_0 \times \text{iter}$, penalty The initial value of the factor is set to $\gamma_0$, generally the empirical value 0.01 is used, and the number of iterations is iter; $Z_{\alpha\beta\gamma\delta}$ is the number of all nodes except the generator; $Z_G$ is the number of generator nodes; $V_{\text{lim}}$ is the upper limit or lower limit of the state variable voltage:

$$V_{\text{lim}} = \begin{cases} V_{\text{lim min}}, & V_{\text{a min}} < V_{\text{a max}} \\ V_{\text{a}}, & V_{\text{a min}} < V_{\text{a}} < V_{\text{a max}} \\ V_{\text{lim max}}, & V_{\text{a}} > V_{\text{a max}} \end{cases}$$  

(6)

$$Q_{\text{lim}} = \begin{cases} Q_{\text{lim min}}, & Q_{\text{b min}} < Q_{\text{b max}} \\ Q_{\text{b}}, & Q_{\text{b min}} < Q_{\text{b}} < Q_{\text{b max}} \\ Q_{\text{lim max}}, & Q_{\text{b}} > Q_{\text{b max}} \end{cases}$$  

(7)

3. Particle Swarm Optimization (PSO)

In 1995, Dr. Kennedy and Dr. Eberhar proposed the particle swarm algorithm [15-16], which is an algorithm used to study bird behavior. Its basic core is to use the information exchange between the various entities in the group, so that the whole group obtains a method from disorder to order in the space of movement, and finally obtains the best solution to the problem.

Suppose a group of birds happen to be looking for food in a spatial area. Although birds do not know the location of the food initially, they can feel the distance between the food and themselves. For solving problems, food is the best answer. The distance between birds and food is the adaptability of the function. The fitness value determined by the optimization function is used to represent the flight of the particles, and the speed of each particle has its specific Flight direction and distance, and adjust the speed according to the experience of individuals and groups.

The particles track each extreme value and all extreme values at the same time, so that the position of the best solution can be better grasped. As shown in the following formula:

$$v_{id}^{k+1} = w v_{id}^k + c_1 r_1 (p_{id}^k - x_{id}^k) + c_2 r_2 (p_{id}^k - x_{id}^k)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1}$$  

(8)

The particle updates itself according to the velocity and position of formula (8).

4. Improved Particle Swarm Algorithm

In view of the fact that the PSO algorithm is easy to fall into the local optimum, the PSO can be improved by increasing the inertia weight and improving the convergence parameters.

4.1. Linear Decreasing Method of Inertia Weight

The linear decreased weighting strategy proposed by the linear decreasing method of inertia weight is introduced. Improve the search accuracy, as shown in the following formula (9).

$$w = w_{\text{max}} - \frac{w_{\text{max}} - w_{\text{min}}}{s_{\text{max}}} s$$  

(9)

The maximum number of iterations is $s_{\text{max}}$, and the current number of iterations is $s$. Take $w_{\text{max}} = 0.9$, $w_{\text{min}} = 0.4$.

4.2. Increase Convergence Parameters
Adding the convergence parameters to the standard particle swarm optimization algorithm is beneficial to the algorithm that maintains diversification. The position and speed update formula of the algorithm is as follows:

\[
\begin{align*}
\omega_i^{k+1} &= \chi \left( \omega_i^k + c_1 r_1 (p_i^k - x_i^k) + c_2 r_2 (g_i^k - x_i^k) \right) \\
\end{align*}
\]  \hspace{1cm} (10)

Where \( \chi = \frac{2}{\phi \sqrt{\phi - 4}} \) is the convergence parameter, \( \phi = c_1 + c_2 > 4 \). \( \phi = 4.1, \chi = 0.729 \).

4.3. Nonlinear Inertia Weight Linear Decrease Method and Added Improved Convergence Parameter Combination

This paper proposes to improve the PSO algorithm by using the method of inertial weight linear decreasing and the method of adding improved convergence parameters. The specific improvement formula is as follows:

\[
\begin{align*}
\omega_i^{k+1} &= \chi \left( \omega_i^k + c_1 r_1 (p_i^k - x_i^k) + c_2 r_2 (g_i^k - x_i^k) \right) \\
\end{align*}
\]  \hspace{1cm} (11)

\[
\begin{align*}
\chi &= \frac{2}{|\cos \phi|} \hspace{1cm} (12) \\
\phi &= 2 \times 3.14^4 f \hspace{1cm} (13) \\
f &= c_1 + c_2 \hspace{1cm} (14)
\end{align*}
\]

Among them, considering the properties of the cosine function that continuously enlarges and shrinks within the range of 2, the optimal value can be obtained. This method improves convergence while ensuring search accuracy. The algorithm flow chart is shown in Figure 1:

![Figure 1](image-url).

**Figure 1.** Improved PSO reactive power optimization flowchart
5. Analysis of Programming Examples

This article uses MATLAB language to compile the PSO algorithm, adopts the node IEEE14, and uses the Newton-Raphson method for power flow calculation. Optimize the program, the reference value of the test power is 100 MVA, the voltage amplitude is the standard unit value, and the voltage deviation of the test result is stable between 0.87 and 1.15. The comparison values are shown in Table 1. Meet the voltage quality requirements.

| node | PSO algorithm voltage value (p.u) | Improved PSO algorithm voltage value (p.u) |
|------|----------------------------------|------------------------------------------|
| 1    | 1.11                             | 1.1                                      |
| 2    | 0.92                             | 0.93                                     |
| 3    | 0.87                             | 0.89                                     |
| 4    | 1.06                             | 1.03                                     |
| 5    | 1.03                             | 1.05                                     |
| 6    | 1.02                             | 0.99                                     |
| 7    | 0.9                              | 0.93                                     |
| 8    | 0.92                             | 0.96                                     |
| 9    | 0.96                             | 1.07                                     |
| 10   | 1.13                             | 1.06                                     |
| 11   | 1.15                             | 1.1                                      |
| 12   | 0.89                             | 0.98                                     |
| 13   | 0.96                             | 1.02                                     |
| 14   | 0.98                             | 1.05                                     |

Table 1. Node voltage table

| algorithm       | Active power loss/MW | Attrition rate |
|-----------------|----------------------|----------------|
| PSO algorithm   | 14.40774             | 4.80%          |
| Improved PSO algorithm | 14.40761     | 4.50%          |

Table 2. Comparison of active power loss

Table 1 and Table 2 analyze the improved PSO algorithm. The active power loss is reduced by 0.13KW, the voltage deviation is smaller, the active power loss value is lower, and the convergence speed is faster.
6. Conclusion

It can be concluded from this paper that the reactive power optimization results of the improved particle swarm optimization algorithm converge faster, the convergence value is smaller than the convergence value before the improvement, and the convergence speed is faster. The example analysis shows that the linear regression method of inertia weight and the increase of the convergence parameters can effectively improve the particle swarm optimization algorithm, which can reduce the active power loss of the network and converge faster. However, the convergence value range is not very stable. The method proposed in this paper has also been improved, but it also needs further improvement to obtain a more stable value.

References

[1] Li, S. S. (2020) Application research of particle swarm optimization in optimization problems. Science and Technology Innovation, (32):103-104.

[2] Huang, B., Zeng, C. Y., Wang, Q. S., Gong, J. (2020) An improved particle swarm optimization algorithm for solving reactive power optimization in power systems. Electrical Engineering Technology, (12): 107-109.

[3] Lei, X. Y. (2018) Power system reactive power optimization based on improved pseudo-gradient particle swarm optimization. Northeast Petroleum University.

[4] Multimedia; (2020) Investigators at Indian Institute of Technology (IIT) Patna Report Findings in Multimedia (A Multi-objective Based Pso Approach for Inferring Pathway Activity Utilizing Protein Interactions). Journal of Technology.

[5] Sensor Research; (2020) Research from Duy Tan University Yields New Data on Sensor Research [Novel Ensemble Approach of Deep Learning Neural Network (DLNN) Model and Particle Swarm Optimization (PSO) Algorithm for Prediction of Gully Erosion Susceptibility]. Robotics & Machine Learning.

[6] Wu, Y. Z., Zou, X. S., Yuan, X. F., Xiong, W., Yao, G. (2018) Power system multi-objective reactive power optimization based on improved particle swarm optimization. New Industrialization, 8(12): 5-9.

[7] Cui, Y. (2018) Research on reactive power optimization of distribution network based on particle swarm optimization. Northeast Petroleum University.

[8] Que, S. M. (2017) Power system reactive power optimization based on improved particle swarm algorithm. Nanjing University of Posts and Telecommunications.

[9] Lu, J., Tuo, X. G., Peng, G. L. (2017) Power system reactive power optimization based on particle swarm optimization. Microcomputers and Applications, 36(13): 19-21.

[10] Pu, H (2017) Reactive power optimization of power system with wind power based on particle swarm optimization. Rural Electrification, (04): 50-51.