International Journal of Energy
ISSN: 2957-9473 | Vol. 2, No. 3, 2023

Power Flow Calculation of Distribution Network with Distributed Power Sources based on Improved Newton Method

Xiaona Yang *

School of Electrical Information and Engineering, Northeast Petroleum University, Daqing 163318, China

Abstract: Aiming at the shortcomings of the existing power flow calculation methods in the distribution network with distributed power supply, a new power flow calculation method for the distribution network with distributed power supply is proposed. The processing methods of 4 different types of grid-connected nodes of distributed power supply are analyzed. The calculation rate of Jacobian matrix is increased by introducing two steps of predicted value and correction value in order to improve the sensitivity of Newton method to initial value. At last, IEEE33-node power distribution system is taken as an example to verify the correctness and performance of the improved Newton method, and its applicability to power flow calculation of distribution network with distributed power supply.

Keywords: Distributed Power Supply; Newton-raphson Method; Forward Push-back Method; Distribution Network.

1. Introduction

The access of Distributed Generation (DG) to the distribution network will change the original structure of the network and the direction of the power flow, and some traditional power flow calculation methods are not suitable for the calculation of the power flow including the DG distribution network, and because the photovoltaic, wind power and other DG is greatly affected by the natural environment, its output active power is not fixed [1-3].

The improvement of some existing power flow calculation is an important method to deal with the power flow calculation after DG is connected to the distribution network, and scholars at home and abroad have done a lot of research on this. In view of the inability of traditional power flow calculation to deal with PV nodes, literature [4] selects the polar coordinate form of Newton's method and linearizes the three nonlinear terms inside it, so that the power flow calculation method can deal with PV nodes. In literature [5], in order to enable the forward push-back method to calculate the power flow of the distribution network containing DG, the forward push-back method was improved by simulating different node types of injection current. In order to solve the problem that the forward push-back substitution method cannot calculate PV nodes effectively, the literature [6] introduced the reactive power correction equation during reactive power correction to improve the forward push-back substitution method. In reference [7], aiming at the shortcomings of the existing linear power flow calculation methods of distribution network, such as poor generality and low calculation accuracy, Taylor series was used to linearize the nonlinear terms of the basic power flow equation, and a linear power flow calculation method including DG distribution network was proposed. Literature [8] proposes a power electron-dominated power flow algorithm based on time domain iteration in order to solve the convergence difficulty of nonlinear power flow equation caused by improper setting of initial conditions, and the traditional power flow calculation is no longer applicable as a large number of power electrons are connected to the distribution network. In order to solve the problem that traditional push-back substitution cannot deal with PV nodes in the distribution network, literature [9] uses the equivalent reactance matrix of PV nodes to obtain the reactive power correction of PV nodes, and then proposes an improved push-back substitution method that can deal with PV type DG on this basis. Literature [10] studied the power flow calculation of distribution network with wind turbines, introduced the interval analysis theory, and used the Krawczyk-Moore operator to solve the interval power flow calculation model, and proposed the power flow calculation of distribution network interval with wind turbines on the basis of considering the uncertainty of wind power.

Most scholars have improved the power flow algorithm on the basis of the forward push-back method, and the improved forward push-back method often deals with some problems by equivalence. The approximation generated in the equivalence process has a certain impact on the accuracy of the power flow calculation, which may lead to the increase of the calculation amount and the calculation time is too long. In contrast, Newton method is widely used in power flow calculation of distribution network because of its excellent convergence speed and accuracy.

2. Common Power Flow Calculation of Distribution Network

2.1. Newton-Raphson Method

Newton's method is a typical method to solve nonlinear equations in mathematics, and its main idea is the process of successive linearization. With the continuous study of Newton's method by scholars at home and abroad, Newton's method has gradually surpassed other methods in convergence, memory requirements, computing speed and so on. It is a widely used power flow calculation method.

If there are n nodes in the network, the voltage of node i is expressed in polar coordinates as follows:

$$\dot{U}_i = U_i e^{j\theta_i} = U_i\left(\cos\theta_i + j\sin\theta_i\right) \quad (i = 1, 2, \cdots, n) \quad (1)$$

Where, U_i is the voltage amplitude of node i; θ_i Is the

voltage phase Angle value of node *i*.

The node power equation is:

$$\begin{cases} \Delta P_i = P_{is} - U_i \sum_{j=1}^{n} U_j \left(G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij} \right) \\ \Delta Q_i = Q_{is} - U_i \sum_{j=1}^{n} U_j \left(G_{ij} \sin\theta_{ij} - B_{ij} \cos\theta_{ij} \right) \end{cases} \tag{2}$$

Where, P_{is}、Q_{is} represents the active power and reactive power of node *i*; θ_{ij} represents the phase Angle difference between node *i* and node *j*; G_{ij} represents the conductance of the branch between node *i* and node *j*; B_{ij} represents the susceptance of the branch between node *i* and node *j*.

By expanding the above equation with Taylor series, the modified equation is obtained as:

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = -\begin{bmatrix} H & N \\ M & L \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta U / U \end{bmatrix} \tag{3}$$

When expressed in polar coordinates, the expression of each element of the non-singular matrix Jacobian matrix is:

$$H_{ij} = \frac{\partial \Delta P_i}{\partial \Delta \theta_j} = \begin{cases} -U_i U_j \left(G_{ij} \sin\theta_{ij} - B_{ij} \cos\theta_{ij} \right) (j \neq i) \\ U_i^2 B_{ii} + Q_i (j = i) \end{cases} \tag{4}$$

$$N_{ij} = \frac{\partial \Delta P_i}{\partial \Delta U_j} U_j = \begin{cases} -U_i U_j \left(G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij} \right) (j \neq i) \\ -U_i^2 G_{ii} - P_i (j = i) \end{cases} \tag{5}$$

$$M_{ij} = \frac{\partial \Delta Q_i}{\partial \Delta \theta_j} = \begin{cases} U_i U_j \left(G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij} \right) (j \neq i) \\ U_i^2 G_{ii} - P_i (j = i) \end{cases} \tag{6}$$

$$L_{ij} = \frac{\partial \Delta Q_i}{\partial \Delta U_j} U_j = \begin{cases} -U_i U_j \left(G_{ij} \sin\theta_{ij} - B_{ij} \cos\theta_{ij} \right) (j \neq i) \\ U_i^2 B_{ii} - Q_i (j = i) \end{cases} \tag{7}$$

The steps for calculating the *k*+1 iteration by Newton's method is:

(1)The voltage amplitude $U_i^{(k)}$ and phase angle $\theta_i^{(k)}$ after the *k* iteration are brought into equation (2) to calculate the active power unbalance $\Delta P_i^{(k)}$ and reactive power unbalance $\Delta Q_i^{(k)}$;

(2) The elements of the non-singular Jacobian matrix are calculated by inserting $U_i^{(k)}$ and $\theta_i^{(k)}$ into the equations (4), (5), (6) and (7), and then the values of the new Jacobian matrix are improved;

(3) The parameters calculated in the above two steps are brought into the formula (3) to calculate the unbalance of voltage amplitude $\Delta U_i^{(k)}$ and voltage phase Angle $\Delta\theta_i^{(k)}$;

(4) The voltage amplitude $U_i^{(k+1)}$ and phase Angle $\theta_i^{(k+1)}$ of each node are corrected:

$$\begin{cases} U_i^{(k+1)} = U_i^{(k)} + \Delta U_i^{(k)} \\ \theta_i^{(k+1)} = \theta_i^{(k)} + \Delta\theta_i^{(k)} \end{cases} \tag{8}$$

(5) Determine whether the convergence condition is met:

$$\begin{cases} \Delta P_i^{(k+1)} < \xi \\ \Delta Q_i^{(k+1)} < \xi \end{cases} \tag{9}$$

If the above condition is met, the calculation ends; otherwise, the next iteration continues [11].

2.2. Backward-forward Substitution

In the radial distribution network, the path of any bus node and root node is unique, and the power flow calculation method is to make full use of this characteristic, through this unique path to correct the voltage and power of each node. Under a constant number of iterations, the calculation amount of each iteration of the forward push-back algorithm is proportional to the number of bus bars, and the calculation complexity will increase with the increase of the network scale, so the forward push-back algorithm has a good performance in solving large-scale radial distribution networks. The push-back method is described by a radial line, as shown in Fig.1.

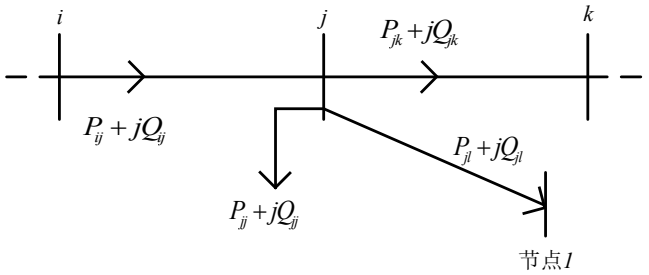


Fig 1. Typical radial circuit diagram

The iterative equation for the *k*+1 iteration with the forward push-back method is:

$$\begin{cases} P_{ki}^{(k+1)} = P_{Di}^{(k)} + \sum_{j \in C_i} P_{ij}^{(k+1)} + \Delta P_{ki}^{(k+1)} \\ Q_{ki}^{(k+1)} = Q_{Di}^{(k)} + \sum_{j \in C_i} Q_{ij}^{(k+1)} + \Delta Q_{ki}^{(k+1)} \end{cases} \tag{10}$$

$$\begin{aligned} \Delta P_{ki}^{(k+1)} &= r_{ki} \left[\left(\sum_{j \in C_i} P_{ij}^{(k+1)} + P_{Di}^{(k)} \right)^2 + \left(\sum_{j \in C_i} Q_{ij}^{(k+1)} + Q_{Di}^{(k)} \right)^2 \right] / \left(U_i^{(k)} \right)^2 \\ \Delta Q_{ki}^{(k+1)} &= x_{ki} \left[\left(\sum_{j \in C_i} P_{ij}^{(k+1)} + P_{Di}^{(k)} \right)^2 + \left(\sum_{j \in C_i} Q_{ij}^{(k+1)} + Q_{Di}^{(k)} \right)^2 \right] / \left(U_i^{(k)} \right)^2 \end{aligned} \tag{11}$$

Where, C_i represents the set formed by the children of node *i*; $\left(P_{ki}^{(k+1)}, Q_{ki}^{(k+1)}\right)$ indicates the active power and reactive power of branch *ki*; $(P_{ij}^{(k+1)}, Q_{ij}^{(k+1)})$ indicates the active power and reactive power of branch *ij*; $\left(P_{Di}^{(k)}, Q_{Di}^{(k)}\right)$ indicates the load of node *i*; $\left(\Delta P_{ki}^{(k+1)}, \Delta Q_{ki}^{(k+1)}\right)$ represents active and reactive power loss on branch *ki*; r_{ki} indicates the resistance of branch *ki*; x_{ki} represents the reactance of branch *ki*; U_i represents the voltage of node *i*. The formula of voltage $\dot{U}_i^{(k+1)}$ and current $\dot{I}_{ki}^{(k+1)}$ of node *i* obtained by stacking *k*+1 is:

$$\dot{I}_{ki}^{(k+1)} = \frac{P_{ki}^{(k+1)} - jQ_{ki}^{(k+1)}}{\dot{U}_k^{*(k+1)}} \tag{12}$$

$$\dot{U}_i^{(k+1)} = \dot{U}_k^{(k+1)} - \dot{I}_{ki}^{(k+1)} \left(r_{ki} + jx_{ki} \right) \tag{13}$$

The specific calculation steps of the backward substitution method are as follows:

(1) Initialization: Assign an initial value to the node voltage, i.e., $\dot{U}_i^{(0)} = 1, \quad (i = 1, 2, \ldots, n)$, where *n* represents the number of nodes in the network;

(2) Step forward calculation power: use formula (10) and (11) to calculate branch loss and node power step by step from the end of the line to the beginning of the line;

(3) Back generation calculation voltage: According to the branch power calculated in the previous step, the node voltage and voltage drop of the line are gradually calculated from the first end of the line to the end of the line using formulas (12) and (13)

(4) Given convergence criterion: a given value ξ is determined by judging the convergence of the node voltage, so that the difference between the voltage amplitude at the *k*

iteration and the voltage amplitude at the *k*-1 iteration is less than a fixed value, and the formula expression is:

$$\left|U_i^{(k)} - U_i^{(k-1)} < \xi\right| \tag{14}$$

Judging by the convergence of the power, the unbalance of the injected power at the *k* iteration is less than this given value, and the formula expression is:

$$\begin{cases} \left|\Delta P_i^{(k)} = P_{ii} - U_i^{(k)} \sum_{j=i} U_j^{(k)} \left(G_{ij} \cos\theta_{ij}^{(k)} + B_{ij} \sin\theta_{ij}\right)^{(k)}\right| < \xi \\ \left|\Delta Q_i^{(k)} = Q_{ii} - U_i^{(k)} \sum_{j\in i} U_j^{(k)} \left(G_{ij} \sin\theta_{ij}^{(k)} + \cos B_{ij}\theta_{ij}\right)^{(k)}\right| < \xi \end{cases} \tag{15}$$

(5) If it is judged that the convergence condition is not met in the previous step, the voltage obtained by iterative calculation is taken as the initial value, and the corresponding value is recalculated from step (2) until the convergence condition is met.

3. DG Grid-connected Node Processing Mode

In the power flow calculation of grid-connected DG, different DG nodes need to be equivalent to different node types. Besides balance node, PV node and PQ node, PI node and PQ (V) node are added.

3.1. PQ Node Type DG

In power flow calculation, the active power *P* and reactive power *Q* emitted by DG equivalent to PQ node are constant values, and PQ node can be directly regarded as "negative load" to deal with. For this type of DG, it is necessary to modify the symbol of the input data during power flow calculation, so the injection current $\dot{I}$ of the grid-connected node of DG can be obtained:

$$\dot{I} = \left[\frac{-P - jQ}{\dot{U}}\right]^* \tag{16}$$

In the formula, P_{DG}、Q_{DG} represent the actual power transmitted by DG to the grid; $\dot{U}$ indicates the node voltage of the DG integrated into the distribution network.

3.2. PV Node Type DG

The active power *P* and voltage amplitude *U* emitted by DG equivalent to PV node are constant, but in power flow calculation, if the voltage amplitude *U* of PV node is not equal to the set voltage amplitude, then a certain current is injected into the PV node to make the voltage amplitude of the node reach the set value, and the compensation of the injected current and the reactive power of the node need to be calculated. The formula for calculating the initial value of reactive power Q_{Gi}^0 and the deviation of injected current ΔI^k at the k iteration is:

$$Q_{Gi}^0 = \sum_{i=1}^{i-1}\left(Q_{iN}/2\right) + \sum_{i=i+1}^{n} Q_{iN} \tag{17}$$

$$\Delta I^k = \mathrm{Z}^{-1}\Delta U^k \tag{18}$$

Where $\sum_{i=1}^{i-1}\left(Q_{iN}/2\right)$ represents the sum of half of the reactive power between the PV node and the first node; $\sum_{i=i+1}^{n} Q_{iN}$ represents the sum of reactive power between the PV node and the first node; ΔU^k represents the unbalance of the point voltage value in the *k* iteration; Z represents the sensitivity impedance matrix in the distribution network, which does not need to be solved several times in the iterative process.

The formula for calculating reactive power deviation ΔQ^k at the *k* iteration is:

$$\Delta Q^k = -\Delta I^k = -Z^{-1}\Delta U^k \tag{19}$$

The formula for obtaining reactive power Q^{k+1} of the *k*+1 iteration is:

$$Q^{k+1} = Q^k + \Delta Q^k \tag{20}$$

In order to make the calculation program simple and further improve the efficiency of calculation, the PV node is treated as PQ node by the above method.

3.3. PQ(V) Node Type DG

The DG of PQ(V) nodes has constant active power and unsteady voltage. Because the reactive power is related to voltage, the reactive power of DG of such nodes is also unsteady. The generator that can handle the PQ(V) node cannot output reactive power itself, so the voltage cannot be adjusted. The grid or synchronous generator provides its own excitation reactive power to such a generator. The reactive power absorbed by the PQ(V) node depends on the slip rate between the voltage of the node and the generator. In each iteration, the voltage amplitude of the node needs to be corrected first, and then the reactive power is obtained. At the *k* iteration, the reactive power Q^k of the obtained PQ(V) DG is:

$$Q^k = -\frac{\left(U^k\right)^2}{x_{\mathrm{p}}} + \frac{-\left(U^k\right)^2 + \sqrt{\left(U^k\right)^4 - 4P^2x^2}}{2x} \tag{21}$$

Where, reactance $x = x_1 + x_2$; Reactance $x_{\mathrm{p}} = \frac{x_{\mathrm{c}}x_{\mathrm{m}}}{x_{\mathrm{c}} - x_{\mathrm{m}}}$; U^k represents the voltage amplitude of the point at the *k* iteration; *P* represents constant active power output by DG; x_1 represents the stator leakage reactance; x_2 represents the leakage reactance of the rotor; x_{c} represents the reactance of the shunt capacitor at the end of the unit; x_{m} represents excitation reactance. Therefore, in order to facilitate the calculation of power flow, the PQ(V) node can be transformed into PQ node.

3.4. PI Node Type DG

In power flow calculation, the active power *P* and current *I* output by DG equivalent to PI node are both constant values. When dealing with such nodes, the voltage obtained after the last iteration is used to reasonably calculate the reactive power. In this way, the processing method of PQ nodes can be used to deal with PI nodes:

$$Q^k = \sqrt{|I|^2\left|U^k\right|^2 - P^2} \tag{22}$$

According to the power relationship of the nodes, it can be obtained that the injection current of PI node $I_{\text{in-PI}}{}^k$ is:

$$I_{\text{in-PI}}{}^k = \left[\frac{-P - jQ^k}{U^k}\right]^* \tag{23}$$

Where, Q^k represents the reactive power of PI type DG at the *k* iteration of power flow calculation; U^k represents the voltage obtained after the *k* iteration; *P* stands for constant active power; *I* represent the constant current amplitude of

DG.

4. Modified Newton Method

In the traditional distribution network, the radial topology is relatively large, and the access of DG changes the structure of the traditional distribution network. The more complex power network is not easy to operate stably, and the workload of the operation and maintenance of the power network is increased. Some DG, such as photovoltaic power generation and wind power generation, are easily affected by weather and natural environment, and their active power output is greatly affected by different weather conditions, with relatively large uncertainties. DG has obvious advantages in many aspects, but the integration of DG into the distribution network brings great challenges to the stable operation of the whole network, especially in the calculation of power flow. First of all, because the access of DG may add new nodes to the original power system that cannot be processed by traditional power flow calculation, it is necessary to use mathematical methods to equivalent process these nodes, so that they can be processed by conventional power flow calculation methods. Secondly, the access of DG will also cause the voltage distribution of the original network to change, which may cause some power flow calculation methods that require high initial value selection cannot converge effectively.

Based on the above analysis and research of various power flow calculations, this paper selects Newton method for power flow calculation of distribution network with DG and makes some improvements according to the shortcomings of the algorithm in solving the problem. As a widely used power flow calculation method, Newton method has good convergence, but in the distribution, network containing DG, the selection of initial value has a very important impact on the result and convergence of power flow calculation. When the initial value is selected appropriately, Newton method is a frequently used power flow calculation method, because of its very good convergence. However, the selection of initial value of power flow calculation of distribution network with DG has a great influence on the result and convergence. Newton's method has the characteristic of square convergence. If the initial value is chosen properly, Newton's method can obtain the solution with higher precision while using fewer iterations. If the initial value is not chosen properly, the power flow calculation will converge to an inappropriate solution or even fail to converge. In addition, Newton's method needs to continuously modify the parameters, resulting in the re-formation of the Jacobian matrix in each iteration, which makes Newton's method have a lot of work in the calculation of the power flow of the distribution network including DG, increasing the calculation amount and memory occupation, and still needs a long time to calculate when the number of convergent times is small. Therefore, for the network with DG access, it is necessary to adopt some improvements to Newton's method to carry out effective power flow calculation and ensure the accuracy and reliability of the results.

In this paper, the predictor-correction mechanism of $1+\sqrt{2}$ -order Newton method is introduced to address the shortcomings of the slow calculation rate of the traditional Newton method. Its main advantages are that it does not need to calculate high-order derivatives and does not need too much data storage, and the value of the Jacobian matrix of the current iteration can be reused in the next iteration, and its univariable iteration formula is as follows:

k=0:

$$\begin{cases} \hat{x}^{(0)} = x^{(0)} \\ x^{(1)} = x^{(0)} - \dfrac{f\left(x^{(0)}\right)}{f'\left(\left[x^{(0)} + \hat{x}^{(0)}\right]/2\right)} = x^{(0)} - \dfrac{f\left(x^{(0)}\right)}{f'\left(x^{(0)}\right)} \end{cases} \tag{24}$$

Where, $x^{(0)}$ is the initial value of the system; $\hat{x}^{(0)}$ is the estimated value of $x^{(0)}$; $f\left(x^{(0)}\right)$ is the function value corresponding to $x^{(0)}$; $f'\left(\left[x^{(0)} + \hat{x}^{(0)}\right]/2\right)$ is the slope of $f(x)$ at $x = \left[x^{(0)} + \hat{x}^{(0)}\right]/2$; $x^{(1)}$ is the correction of the initial value of the system after the first iteration.

k≥1:

$$\begin{cases} \hat{x}^{(k)} = x^{(k)} - \dfrac{f\left(x^{(k)}\right)}{f'\left(\left[x^{(k-1)} + \hat{x}^{(k-1)}\right]/2\right)} \\ x^{(k+1)} = x^{(k)} - \dfrac{f\left(x^{(k)}\right)}{f'\left(\left[x^{(k)} + \hat{x}^{(k)}\right]/2\right)} \end{cases} \tag{25}$$

Where, $\hat{x}^{(k)}$ is the estimated value of $x^{(k)}$; $x^{(k+1)}$ represents the correction of x; $f\left(x^{(k)}\right)$ is the function corresponding to $x^{(k)}$; $f'\left(\left[x^{(k)} + \hat{x}^{(k)}\right]/2\right)$ is the slope of $f(x)$ at $x = \left[x^{(k)} + \hat{x}^{(k)}\right]/2$.

Its multivariable matrix form is shown in the following formula, *k*=0:

$$\begin{cases} \hat{x}^{(0)} = x^{(0)} \\ \Delta x^{(0)} = \dfrac{f\left(x^{(0)}\right)}{J\left(\left[x^{(0)} + \hat{x}^{(0)}\right]/2\right)} \Rightarrow \Delta x^{(0)} = J\left(x^{(0)}\right)^{-1} f\left(x^{(0)}\right) \\ x^{(1)} = x^{(0)} - \Delta x^{(0)} \end{cases} \tag{26}$$

Where, $\Delta x^{(0)}$ is the error generated by the initial value of the system after the first iteration; $J\left(\left[x^{(0)} + \hat{x}^{(0)}\right]/2\right)$ is the Jacobian matrix corresponding to $f(x)$ at $x = \left[x^{(0)} + \hat{x}^{(0)}\right]/2$.

k≥1:

$$\begin{cases} \Delta\hat{x}^{(k)} = \dfrac{f\left(x^{(k)}\right)}{J\left(\left[x^{(k-1)} + \hat{x}^{(k-1)}\right]/2\right)} \Rightarrow \Delta\hat{x}^{(k)} = J\left(\left[x^{(k-1)} + \hat{x}^{(k-1)}\right]/2\right)^{-1} f\left(x^{(k)}\right) \\ \hat{x}^{(k)} = x^{(k)} - \Delta\hat{x}^{(k)} \end{cases} \tag{27}$$

$$\begin{cases} \Delta x^{(k)} = \dfrac{f\left(x^{(k)}\right)}{J\left(\left[x^{(k)} + \hat{x}^{(k)}\right]/2\right)} \Rightarrow \Delta x^{(k)} = J\left(\left[x^{(k)} + \hat{x}^{(k)}\right]/2\right)^{-1} f\left(x^{(k)}\right) \\ x^{(k+1)} = x^{(k)} - \Delta x^{(k)} \end{cases} \tag{28}$$

Where, $\Delta\hat{x}^{(k)}$ is the estimated value of $\Delta x^{(k)}$; $\Delta x^{(k)}$ represents the error after *k* iterations; $J\left(\left[x^{(k)} + \hat{x}^{(k)}\right]/2\right)$ is the Jacobian matrix of $f(x)$ at $x = \left[x^{(k)} + \hat{x}^{(k)}\right]/2$.

The prediction step and correction step are introduced into the traditional Newton method to reduce the memory occupation and speed up the calculation. The main calculation diagram is shown in Fig.2.

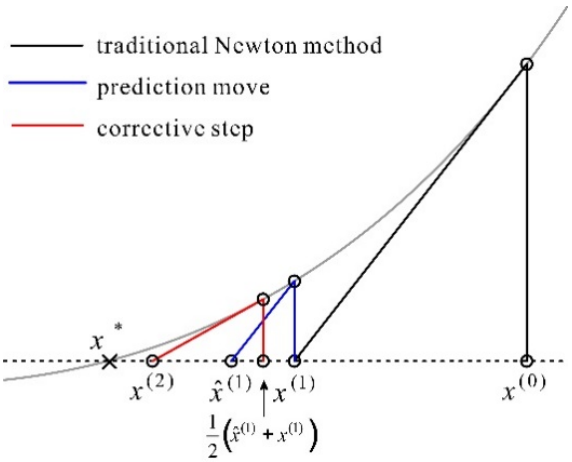


Fig 2. Schematic diagram of 12th-order Newton method

As can be seen from Fig.2, the first iteration is similar to the traditional Newton-Raphson method. In addition, the following steps should be taken:

(1)The Jacobian matrix value $J([x^{(0)}+\hat{x}^{(0)}]/2)\rightarrow J(x^{(0)})$ is used repeatedly to find the middle prediction value $\hat{x}^{(1)}$, this stage is called prediction;

(2) Calculate $x=\left[x^{(1)}+\hat{x}^{(1)}\right]/2$;

(3) Calculating the value of $x^{(2)}$, this stage is called correction;

(4) In the process of each calculation iteration, the value of $J\left(\left[x^{(k)}+\hat{x}^{(k)}\right]/2\right)$ obtained from the solution does not need to be recalculated in the next iteration, and the value of the last iteration can be reused;

(5) The iterative process needs to be repeated until an ideal convergence requirement is met or the maximum number of iterations is exceeded.

In this paper, the Jacobian matrix of Newton method is modified by introducing two steps of calculation of predicted value and correction value in $1+\sqrt{2}$ -order Newton method, which reduces a large amount of work in the calculation of power flow of distribution network including DG, reduces the memory space occupied by data in the process of power flow calculation iteration, and improves the calling and running speed of the program. Then the speed of power flow calculation is improved. However, the improved Newton method still has certain requirements for the selection of the initial value, so in the power flow calculation, the forward push-back method can be used to make a rough estimate of the initial value. DG's access to the distribution network makes the structure of the power grid more complicated, and it is difficult to use the forward push-back method to make an accurate power flow calculation, but it is easy to make a rough estimate of the initial value. And the simple estimation of the initial value by the push-back method can effectively reduce the number of iterations in the calculation of Newton method, and improve the convergence accuracy and calculation speed of power flow calculation.

By combining $1+\sqrt{2}$ -order Newton method and forward push-back method, the initial value of the network can be roughly calculated by using the forward push-back method. Combining the advantages of the two methods, the number of iterations can be effectively reduced and the speed of power flow calculation can be increased. The specific flow chart of the improved Newton method for power flow calculation is shown in Fig.3.

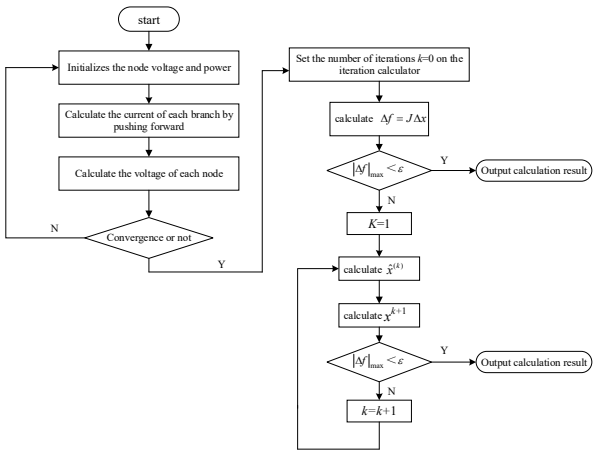


Fig 3. Flow chart of power flow calculation with improved Newton method

5. Analysis of Examples

In this chapter, the IEEE33-node distribution network system is taken as an example, and the topology structure of the system is shown in Fig.4. To facilitate calculation using per unit values, set the base value $U_B=12.66\text{kV}$, $S_B=1000\text{kVA}$.

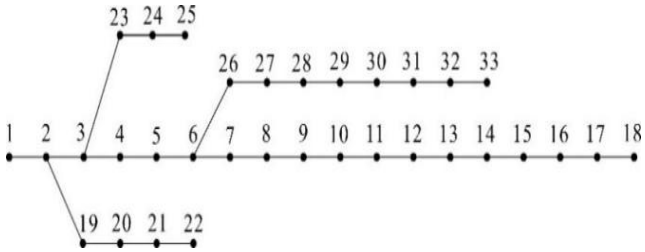


Fig 4. Power distribution system of the IEEE33 nodes

5.1. Analysis of Examples 1

The improved Newton method, the traditional Newton method and the forward push-back method are used to calculate the power flow of the example system by MATLAB simulation, and the accuracy and calculation time of the three algorithms are compared to determine the correctness and superiority of the improved power flow calculation. The voltage of each node obtained by the three power flow calculation methods is shown in Tab.1, and the voltage amplitude of each node is shown in Fig.5. The calculation error statistics of the three-power flow are shown in Tab.2, and the calculation time of the three-power flow is shown in Tab.3.

As can be seen from Tab.1 and Fig.5, the three power flow calculation methods can converge reliably before the set maximum number of iterations, proving the correctness of the improved Newton method. However, compared with the reference voltage amplitude, it can be seen that the calculation accuracy of the three power flow calculation methods is different. The node voltage amplitude obtained by the improved Newton method is the closest to the reference value, and the node voltage amplitude obtained by the forward push-back method is the largest gap between the node voltage amplitude and the reference value.

Table 1. Voltage values of each node

Node numbering	Reference voltage amplitude	Modified Newton method	Traditional Newton method	Backward-forward substitution
1	1.000	1.00000	1.00000	1.00000
2	0.996	0.99704	0.99712	0.99716
3	0.982	0.98289	0.98293	0.98389
4	0.975	0.97540	0.97638	0.97689
5	0.968	0.96811	0.96900	0.97002
6	0.946	0.94889	0.94951	0.95310
7	0.945	0.94601	0.94615	0.94959
8	0.940	0.94142	0.94189	0.94521
9	0.934	0.93451	0.93485	0.93942
10	0.928	0.92901	0.92902	0.93412
11	0.928	0.92816	0.92839	0.93346
12	0.925	0.92666	0.92689	0.93195
13	0.919	0.92054	0.92079	0.92631
14	0.919	0.91851	0.91978	0.92456
15	0.917	0.91739	0.91686	0.92294
16	0.916	0.91588	0.91545	0.92169
17	0.913	0.91347	0.91370	0.91988
18	0.912	0.91286	0.91309	0.91956
19	0.997	0.99658	0.99620	0.99762
20	0.993	0.99312	0.99389	0.99678
21	0.992	0.99222	0.99241	0.99312
22	0.991	0.99146	0.99172	0.99239
23	0.978	0.97902	0.97945	0.98179
24	0.973	0.97268	0.97261	0.98111
25	0.968	0.96889	0.96944	0.97316
26	0.948	0.94757	0.94772	0.95728
27	0.945	0.94512	0.94531	0.94846
28	0.934	0.93373	0.93343	0.93716
29	0.925	0.92530	0.92561	0.92886
30	0.921	0.92174	0.92196	0.92531
31	0.918	0.91789	0.91759	0.92114
32	0.916	0.91662	0.91689	0.92031
33	0.915	0.91621	0.91676	0.91995

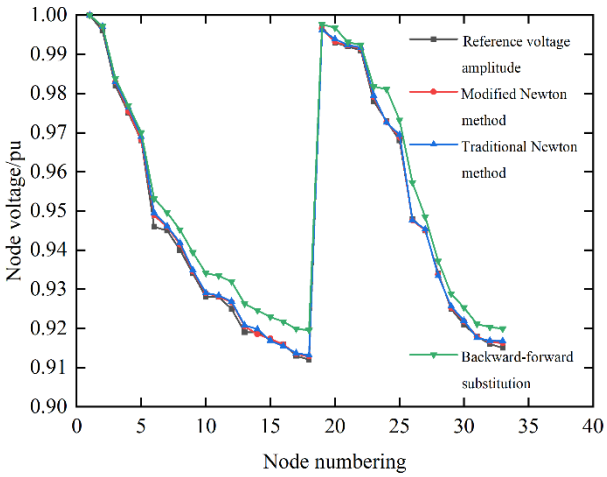


Fig 5. Voltage distribution diagram of each node in the system

Table 2. Statistical table of calculation errors of the three power flows

Power flow calculation method	Minimum error value	Maximum error value	Minimum error rate %	Maximum error rate %
Modified Newton method	0.00011	0.00289	0.012	0.305
Traditional Newton method	0.00031	0.00351	0.033	0.371
Backward-forward substitution	0.00062	0.00928	0.062	0.978

The minimum and maximum errors of the three power flow algorithms can be calculated through the simulation results. As can be seen from Tab.2, the minimum and maximum errors of the improved Newton method are smaller than those of the traditional Newton method and the forward push-back method, among which the former has the largest error. Therefore, the improved Newton method has better accuracy in the power flow calculation of the distribution network.

Table 3. Calculation schedules of the three power flows

Power flow calculation method	computation time
Modified Newton method	0.015s
Traditional Newton method	0.058s
Backward-forward substitution	0.106s

It can be seen from the data in Tab.3 that the improved Newton method takes the least time and the fastest time to calculate the power flow of the example system, while the forward push-back method takes the most time and the slowest time. Therefore, compared with the other two methods, the improved Newton method has a better calculation speed in the power flow calculation of distribution network.

5.2. Analysis of Examples 2

In order to verify the applicability of the improved Newton method in the distribution network containing DG, PQ and PI type DG are selected to connect to the IEEE33-node system. Tab.4 shows the specific access scheme of DG. The improved Newton method and the traditional Newton method are used to calculate the power flow of the example system connected to DG by MATLAB simulation, and the applicability of the improved Newton method in the distribution network with DG is verified. The simulated voltage values of each node obtained by the two power flow calculation methods are shown in Fig.6.

Table 4. Calculation time of the three power flow modes Tab.4 distributed power supply access schemes

Access the distributed power supply type	Access node	Parameters for adding a distributed power supply
PI	15	$P = 200\text{kW}, I = 2\text{A}$
PQ	32	$P = 130\text{kW}, Q = 25\text{kvar}$

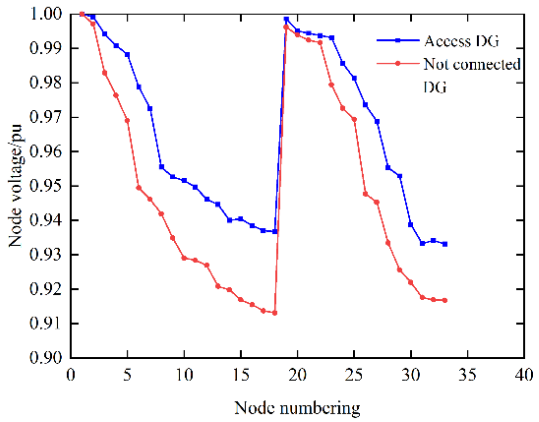


Fig 6. System voltage distribution before and after DG is connected

As can be seen from Fig.6, after DG is connected to the distribution network, the power flow distribution of the

system is changed, which plays a positive role in voltage improvement, and the applicability of improved Newton method power flow calculation in the distribution network system with DG is verified.

6. Conclusion

In order to solve the problem of slow calculation rate of traditional Newton method, Newton method is improved by introducing the predicted value and correction value of $1+\sqrt{2}$ -order Newton method. The improved Newton method revises the Jacobian matrix of the system and increases the calculation rate of the Jacobian matrix. At the same time, in order to improve the problem of sensitivity to the initial value, the push-back method is used to estimate the initial value, which effectively reduces the number of iterations in the calculation of Newton method and improves the convergence speed of power flow calculation. Finally, the IEEE33-node power distribution system is taken as an example to verify the correctness, superiority and applicability of the improved Newton method in the power flow calculation of the power distribution network with DG.

References

[1] Zhang Xuezi. Application of intelligent particle swarm optimization in Distributed energy planning [D]. China University of Mining and Technology, 2019.

[2] Xu Jianjun, Huang Lida, Yan Limei, Yi Na. Insulator Self-Explosion Defect Detection Based on Hierarchical Multi-Task Deep Learning[J]. Transactions of China Electrotechnical Society, 2021,36(07):1407-1415.

[3] Limei,LIU Yongqiang,XU Jianjun,et al.Broken string diagnosis of composite insulator based on Grabcut segmentation and filler area discrimination[J].Power System Protection and Control,2021,49(22):114-119.

[4] Yi, Q. Wang, L. Yan, et al., A multi-stage game model for the false data injection attack from attacker's perspective. Sustainable Energy Grids & Networks 28 (2021).

[5] Na Yi,Jianjun Xu,Limei Yan,Lin Huang. Task Optimization and Scheduling of Distributed Cyber-physical System Based on Improved Ant Colony Algorithm. Future Generation Computer Systems, 109(Aug. 2020),134-148.

[6] Yang Zhao, Jianjun Xu, Jingchun Wu. A New Method for Bad Data Identification of Oilfield Power System Based on Enhanced Gravitational Search-Fuzzy C-Means Algorithm. IEEE Transactions on Industrial Informatics. VOL. 15, NO. 11, NOVEMBER 2019 5963-5970.

[7] Jing Han,Xi Wang,LiMei Yan, Aida Dahlak, et al. Modelling the performance of an SOEC by optimization of neural network with MPSO algorithm. International Journal of Hydrogen Energy, Volume 44, Issue 51, 22 October 2019, Pages 27947-27957. https://doi.org/10.1016/j.ijhydene.2019.09.055.

[8] Zhu Y X, Zong C X, Fan Q G, et al. A Generic Power Flow Algorithm for the Microgrid Based on Time Domain Iteration Concept[J]. Ieee Access, 2020, 8(03): 223626-223636.

[9] Wang Bin, Huang Yanquan, Yang Xi, et al. Forward push-back power flow calculation for distributed Power distribution network with PV [J]. Electrical Automation, 2019, 41(02): 49-52+118.

[10] Liu Gang, Wang Xiuru, Li Hua, et al. Power flow calculation in distribution system considering uncertainty of wind power [J]. Electrical Measurement and Instrumentation, 2022, 59(02): 126-132.

[11] Zhang Libin. Research on three-phase power flow and Multi-objective Optimization Algorithm of distribution network with Distributed Power Supply [D]. South China University of Technology, 2020.