Research of Power Flow Parallel Partition Algorithm Applied to AC Power Grid Based on Two Layers Tree Structure

Tan Tao[1,a], Long Rishang[1], Hu Hao[1], Liu Yang[1], Meng Yuan[1], Zhang Jianhua[1], Xu Yunfei[2], Jin Guofeng[2]

[1]North China Electric Power University, Beijing, China, 102206

[2]Inner Mongolia Eastern Electric Power Co., Ltd. Economic and Technical Research Institute, Inner Mongolia, China,010010

[a]taotan163@163.com

Keywords: calculation of two layers tree structure, power flow, ac network, parallel partition algorithm.

Abstract. This paper present a parallel partition power flow algorithm applied to ac grid based on two layers of tree structure. The algorithm introduces network partition thought in Newton-Raphson power flow algorithm and form two layers tree model to realize the fast calculation of the nodes voltage correction. Network partition principles are given after briefly expounding of tree structure algorithm so called "upward substitution" and "downward computing" step-by-step calculation method. To demonstrate the correctness of the algorithm, this paper sets the IEEE30 nodes system as an example and the result shows that parallel partition algorithm is faster than traditional ones. The characteristics of the algorithm are as follows: The algorithm partition the Jacobi matrix and omits unnecessary calculation so that reduce the dimension of the matrix, so it is realized that greatly reducing the time of matrix inversion; Algorithm in this paper just conducts linear transformation in the primary Jacobi matrix and do not involve simplification and approximation calculation, nor to introduce extra variables like virtual current, so the convergence and accuracy of algorithm can equal traditional serial algorithm.

Introduction

Power flow calculation is the basis of most electric power system application analysis like planning, static analysis and safety evaluation. It can be divided into two parts, establishing model and solving. The requirements of calculating speed and precision can be moderately relaxed to speed up the calculation process in planning and safety evaluation but in real-time scheduling especially automation scheduling, the power flow calculation must be fast and accurate enough.

Considering calculation speed, convergence, memory and other factors, the most widely used method in power flow calculation is the Newton-Raphson algorithm[1] and fast decoupling algorithm. However, these two kinds of algorithm can not achieve fast convergence and even invalid in some special cases: 1) there are many small impedance and negative impedance in the radial network; 2) the initial value is not reasonable; 3) the Jacobi matrix is irreversible; 4) there are too many nodes to calculate fastly. To solve above problems, experts have accumulated a lot of research results at home and abroad. To solve the problem two, experts present modified initial value algorithm[2]. Some other experts put forward that to use the optimal multiplier Newton method in the rectangular coordinates and polar coordinates to solve the problem of convergence, but when the initial value is not reasonable, it will also get the local minimum results, even non-convergent results[3]. Continuous power flow are used in paper[4] to obtain good convergence by introducing extra parameter in Jacobian matrix. Algorithms above only focus on the first three problems. As to the fourth one, especially after the interconnecting of big power grids and the connecting of microgrids, sometimes it is hard to balance calculation precision and speed relying on computer processors and power flow algorithm applying in industry.

Some experts put forward a kind of alternating iterative partition parallel algorithm, which sacrifice precision to achieve fast calculation, apply to large power system based on cluster machine in paper [5]. Inexact Newton power flow algorithm based on block weighted average is proposed in paper [6]. Convergence and speed increase with the increase of number of partitions. Another parallel algorithm based on branch cutting is proposed in paper [7]. To make the convergence and accuracy meet the requirement, this algorithm introduce coordinate variables in cutting point, but the calculation time is not significantly reduced. The method mentioned above not only has its characteristics, but also some deficiencies in application. Considering two layers tree structure model, this paper puts forward a kind of parallel partition algorithm.

Two layers tree structure

Elements of two layers tree structure

Power system design is according to the principle of "geographical division, voltage layers", so the characteristics of partition coordination and layered management make the two layers tree applied in system has a natural partition layer properties. In this paper, the connection between the grid and regional grid is "the first layer division"; and the connection between regional grid to the provincial grid is "the second layer division". After the partition, three will be three types of nodes according to the scope and level: root nets, leaf net and the middle and their relationship are as Fig.1. 1) root node which include nodes besides the cutting lines is the starting of all subnets and locates on the top of the tree model. 2) through the first layer division, we can achieve middle nodes which are located in the middle of the tree. 3) Leaf nodes are located in the bottom of the tree and they are achieved through the second layer division.

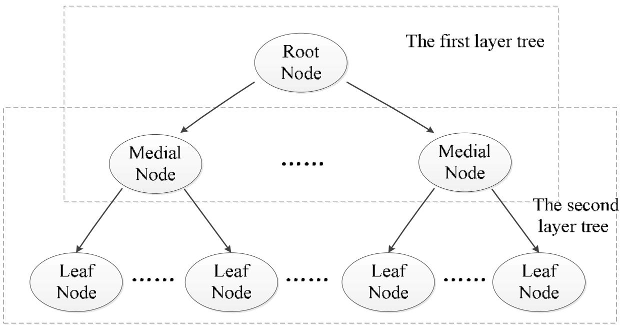

Fig.1. Two layer tree structure mode

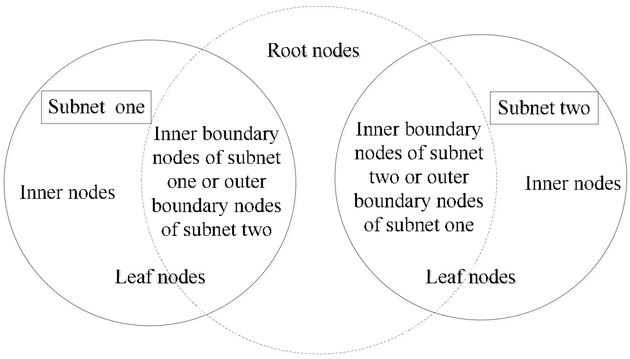

Fig.2. Relationship of nodes in the grid

After division, all subnets connected by links are composed of a number of cables and nodes. Every node in the system belongs to only one subnet and subnets are neither cross nor independent. Nodes can be classified into three categories according to their locating in the subnets: inner nodes, inner boundary nodes and outer boundary nodes. Relationship of three categories nodes is shown in Fig.2. 1) Inner nodes belong to leaf nodes but don’t belong to the root node. They are in subnet and don’t connect with outside. 2) Inner boundary nodes belong to both leaf nodes and the root node. They are in subnet and connecting with outside. 3) Outer boundary nodes belong to the root nodes but don’t belong leaf nodes and they are in other subnet.

principles of division

Grid division generally falls into two categories, branch cutting, which means cutting the branch in the original network to achieve the decomposition of the grid, and the node tearing, which means tearing some nodes to achieve the decomposition of the grid. In addition to differences in the form of the two methods, coordination variables are also different. Variables of branch cutting are line currents but counterparts of node split are voltages. Principles of division are as follows: 1) To balance the calculating time and reduce the waiting time, node number of each subnet node should be equal or close; 2) As far as possible to reduce the amount of the nodes in the root node, that is to reduce the line cutting; 3) Root nodes must consisted of PQ nodes because if there exists balance or PV nodes which have known voltage, it's impossible to reach a single result. 4) The number of partitions should follow the moderation principle, in general, less than ten.

Principles of tree structure parallel partition algorithm

Calculation thought

Power flow calculation will inherit the direction characteristic after introducing in tree structure. The process of the algorithm mainly includes two steps: "upward substitution" and "downward computing".

Upward substitution is an inverse process which starts from leaf nodes, and ends at the root node. Achieve the linear expression of boundary nodes voltage correction though changing the Jacobi matrix, then combine as a new equation set, by solving which we can get the voltage correction of boundary nodes.

Firstly, partition the grid, and reorder the nodes in each subnet as the new admittance matrix (1):

$$\boldsymbol{Y} = \begin{bmatrix} \boldsymbol{Y}_{IN} & \boldsymbol{Y}_{INIB} & \boldsymbol{Y}_{INOB} \\ \boldsymbol{Y}_{IBIN} & \boldsymbol{Y}_{IB} & \boldsymbol{Y}_{IBOB} \\ \boldsymbol{Y}_{OBIN} & \boldsymbol{Y}_{OBIB} & \boldsymbol{Y}_{OB} \end{bmatrix}. \quad (1)$$

$\boldsymbol{Y}_{IN}$ is the admittance matrix of inner nodes in subnet. $\boldsymbol{Y}_{IB}$ is the admittance matrix of inner boundary nodes. $\boldsymbol{Y}_{OB}$ is the admittance matrix of outer boundary nodes. $\boldsymbol{Y}_{INIB}$ is the admittance matrix between inner nodes and inner boundary nodes. Other symbols in (2) can be analogized.

Newton-Raphson power flow algorithm is a process that iterates the correction equation for many times and approaches the final solution by solving the correction equation. Correction equation is as Eq.2

$$\boldsymbol{J}\Delta\boldsymbol{x} = \Delta\boldsymbol{F} \ . \quad (2)$$

$$\Delta\boldsymbol{x} = \begin{bmatrix} \Delta e \\ \Delta f \end{bmatrix} \qquad \Delta\boldsymbol{F} = \begin{bmatrix} \Delta P \\ \Delta Q \\ \Delta U^2 \end{bmatrix} \ . \quad (3)$$

Set the initial voltage of each node and the initial Jacobi matrix can be obtained according to the admittance matrix.

$$\boldsymbol{J}^{(0)} = \begin{bmatrix} \boldsymbol{J}_{IN}^{(0)} & \boldsymbol{J}_{INIB}^{(0)} & \boldsymbol{J}_{INOB}^{(0)} \\ \boldsymbol{J}_{IBIN}^{(0)} & \boldsymbol{J}_{IB}^{(0)} & \boldsymbol{J}_{IBOB}^{(0)} \\ \boldsymbol{J}_{OBIN}^{(0)} & \boldsymbol{J}_{OBIB}^{(0)} & \boldsymbol{J}_{OB}^{(0)} \end{bmatrix}. \quad (4)$$

Divide the matrix (4) and substitute it and matrix (3) into Eq.2, then Eq.5 can be obtained.

$$
\begin{array}{c} \\ \\ n \end{array}
\overset{m}{\left[\begin{array}{cc:c} \boldsymbol{J}_{IN} & \boldsymbol{J}_{INIB} & \boldsymbol{J}_{INOB} \\ \boldsymbol{J}_{IBIN} & \boldsymbol{J}_{IB} & \boldsymbol{J}_{IBOB} \\ \hdashline \boldsymbol{J}_{OBIN} & \boldsymbol{J}_{OBIB} & \boldsymbol{J}_{OB} \end{array}\right]}
\left[\begin{array}{c} \Delta\boldsymbol{x}_{IN} \\ \Delta\boldsymbol{x}_{IB} \\ \hdashline \Delta\boldsymbol{x}_{OB} \end{array}\right] = \left[\begin{array}{c} \Delta\boldsymbol{F}_{IN} \\ \Delta\boldsymbol{F}_{IB} \\ \hdashline \Delta\boldsymbol{F}_{OB} \end{array}\right]. \tag{5}
$$

$\Delta\boldsymbol{x}$ is the corrective vector, $\Delta\boldsymbol{F}$ is the power imbalance vector. Suppose the subnet includes m inner boundary nodes and n outer boundary nodes, then $\Delta\boldsymbol{x}_{IB}$ and $\Delta\boldsymbol{F}_{IB}$ are m dimensional vector, $\Delta\boldsymbol{x}_{OE}$ and $\Delta\boldsymbol{F}_{OB}$ are n dimensional vector.

Remove the correction equations corresponding to outer boundary nodes and the Eq.5 can be rewritten in the form of Eq.6.

$$
\begin{bmatrix} \boldsymbol{J}_{IN} & \boldsymbol{J}_{INIB} \\ \boldsymbol{J}_{IBIN} & \boldsymbol{J}_{IB} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{x}_{IN} \\ \Delta\boldsymbol{x}_{IB} \end{bmatrix} + \begin{bmatrix} \boldsymbol{J}_{INOB} \\ \boldsymbol{J}_{IBOB} \end{bmatrix} \Delta\boldsymbol{x}_{OB} = \begin{bmatrix} \Delta\boldsymbol{F}_{IN} \\ \Delta\boldsymbol{F}_{IB} \end{bmatrix}. \tag{6}
$$

$$
\begin{bmatrix} \boldsymbol{J}_{IN} & \boldsymbol{J}_{INIB} \\ \boldsymbol{J}_{IBIN} & \boldsymbol{J}_{IB} \end{bmatrix} = \boldsymbol{J}_A \qquad \begin{bmatrix} \boldsymbol{J}_{INOB} \\ \boldsymbol{J}_{IBOB} \end{bmatrix} = \boldsymbol{J}_B . \tag{7}
$$

$\boldsymbol{J}_A$ and $\boldsymbol{J}_B$ are subblocks of Jacobi matrix 4, then Eq.6 can be reformed as Eq.8.

$$
\boldsymbol{J}_A \begin{bmatrix} \Delta\boldsymbol{x}_{IN} \\ \Delta\boldsymbol{x}_{IB} \end{bmatrix} + \boldsymbol{J}_B \Delta\boldsymbol{x}_{OB} = \begin{bmatrix} \Delta\boldsymbol{F}_{IN} \\ \Delta\boldsymbol{F}_{IB} \end{bmatrix}. \tag{8}
$$

Left multiply by $\boldsymbol{J}_A^{-1}$ on both sides of the equation, get Eq. 9.

$$
\mathrm{m}\left[\begin{array}{c} \Delta\boldsymbol{x}_{IN} \\ \hdashline \Delta\boldsymbol{x}_{IB} \end{array}\right] = \boldsymbol{J}_A^{-1} \left[\begin{array}{c} \Delta\boldsymbol{F}_{IN} \\ \hdashline \Delta\boldsymbol{F}_{IB} \end{array}\right] - \boldsymbol{J}_A^{-1} \boldsymbol{J}_B \Delta\boldsymbol{x}_{OB} . \tag{9}
$$

If $\boldsymbol{J}_A^{-1}\boldsymbol{J}_B = \boldsymbol{a}$ and $\boldsymbol{J}_A^{-1} \begin{bmatrix} \Delta\boldsymbol{F}_{IN} \\ \Delta\boldsymbol{F}_{IB} \end{bmatrix} = \boldsymbol{b}$, Eq.9 can be reformed as Eq.10.

$$
\mathrm{m}\left[\begin{array}{c} \Delta\boldsymbol{x}_{IN} \\ \hdashline \Delta\boldsymbol{x}_{IB} \end{array}\right] = \boldsymbol{b} - \boldsymbol{a}\Delta\boldsymbol{x}_{OB} . \tag{10}
$$

Extract the last m elements of column vector a and b, new column vector $\boldsymbol{a}'$ and $\boldsymbol{b}'$ can be obtained. Extract the last m equations of Eq.10, new equation set (11) can be rebuild, too.

$$
\Delta\boldsymbol{x}_{IE} = \boldsymbol{b}' - \boldsymbol{a}'\Delta\boldsymbol{x}_{OB} . \tag{11}
$$

Eq.11, which expresses the voltage correction of inner boundary nodes by outer one, is called a linear expression of boundary nodes voltage correction.

Inner boundary nodes and all outer boundary nodes of any one subnet combine form the root node. Considering the real part and imaginary part of voltage phasor, when combine voltage correction equations of all boundary nodes, we can get an equation set including $2\sum_{i=1}^{k} m_i$ equations, in which the k means there are k subnets and the m_i means the number of inner boundary nodes of subnet i. The outer boundary nodes of one subnet are the sum of inner boundary nodes of other subnets, so variables in the equations is also $2\sum_{i=1}^{k} m_i$. When the number of nonlinear correlation equations equals to the number of variables, there exists only one result. We can reach the voltage correction of the root node though solving the new equation set so that realizing the replacement from leaf nodes to the root node.

Downward computing is a forward process which starts from the root node, and ends at leaf nodes. Solve the Eq.9 with voltage correction of root node from above, then voltage correction if inner nodes can be obtained.

Calculating process

The algorithm mentioned is part Newton-Raphson power flow algorithm when applying to a large scale grid.

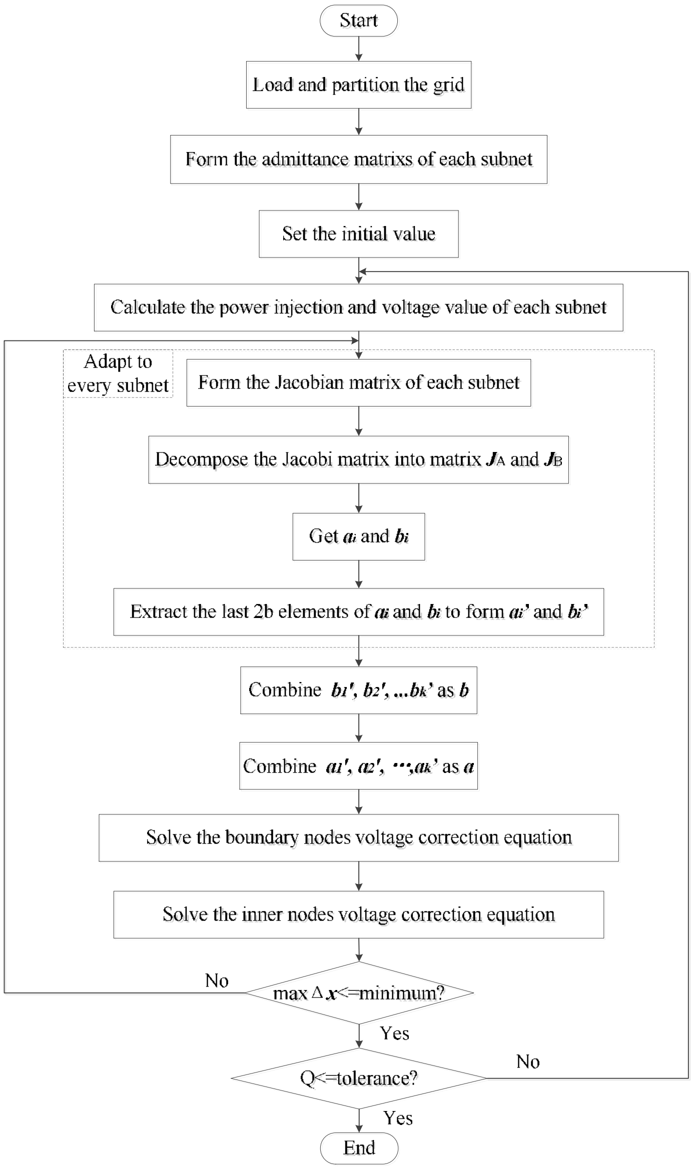

Fig.3. Process of the algorithm

Fig. 3 is the process of the algorithm. 1) Partition the grid into several subnets following the principle mentioned above. 2) Identify and extract three types of nodes in each subnet. Form the admittance matrix according to Eq.1. 3) Set initial voltage and calculate the power injection and voltage value of each subnet. 4) Form the Jacobi matrix and decompose it into $\boldsymbol{J}_{\mathrm{A}}$ and $\boldsymbol{J}_{\mathrm{B}}$. 5) Get $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$. 6) Extract the last 2b elements of $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$ to form $\boldsymbol{a}_i'$ and $\boldsymbol{b}_i'$. 7) Combine $\boldsymbol{b}_1', \boldsymbol{b}_2', ..., \boldsymbol{b}_k'$ as $\boldsymbol{b}$ and combine $\boldsymbol{a}_1', \boldsymbol{a}_2', ..., \boldsymbol{a}_k'$ as $\boldsymbol{a}$. 8) Solve the boundary nodes voltage correction equation. 9) Solve the inner nodes voltage correction equation. 10) Judge whether the voltage correction exceed the setting threshold or not, if exceed, return to step 5). 11) Judge whether the reactive power exceed the setting threshold or not, if exceed, return to step 4), 12) Output the answer.

Case study

To verify the effectiveness of parallel partitioning algorithm mentioned in this paper, the corresponding program is written on MATLAB platform according to the process mentioned last chapter. In this paper, set IEEE30 node system, which contains six generator system with reactive power limit shown in table 1, as an example to verify this algorithm. The types of nodes and the voltage limit are shown in table 2.

Table 1. Reactive power limit of generator node

Node Number	Lower limit of reactive power(p.u.)	Upper limit of reactive powe(p.u.)
2	-0.2	0.6
5	-0.15	0.625
8	-0.15	0.5
11	-0.1	0.4
13	-0.15	0.45

Table 2. Types of nodes and the voltage limit

Type of node	Node number	Lower limit of voltage（p.u.）	Upper limit of voltage（p.u.）
Balance node	1	0.95	1.05
PV	2、5、8、11、13	0.95	1.1
PQ	3、4、6、7、9、10、12、14-30	0.95	1.05

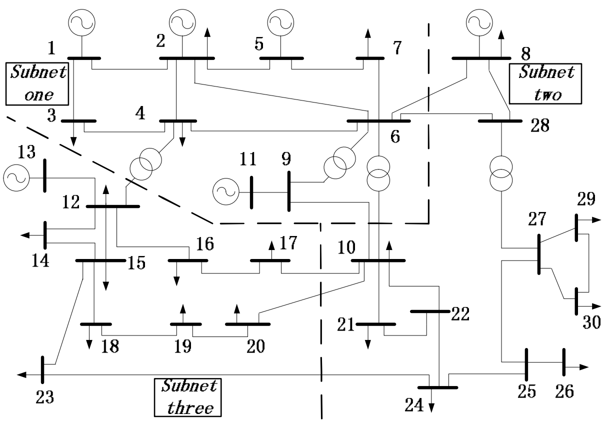

Fig. 4.The partition method of IEEE 30 nodes system

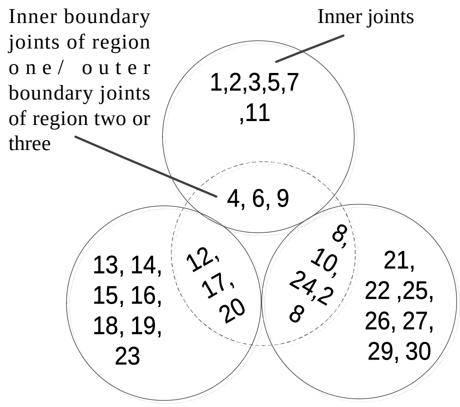

Fig. 5. Node types of IEEE 30 nodes system

Partition the net like Fig.4. Subnet one contains nine nodes, network nodes contains 11 nodes and subnet three contains 10 nodes. Every node connecting with the cutting line is PQ node, so the cutting method can meet the partition principle.

Make the convergence precision in program equals 1E-12. Run the program and the results are shown in table 3. It is known from table 3 that the program reach convergent after 5 iterations. To compare the parallel partition algorithm with traditional serial algorithm, extract the max voltage correction when get every iterative result and calculate the absolute error.

Table 3 Result of simulation

Iterations	Voltage correction (parallel. p. u.)	Voltage correction (serial. p. u.)	Absolute deviation (p. u.)
1	0.436608	0.436608	0
2	0.125896	0.125896	0
3	0.004562	0.004562	0
4	7.35641e-009	4.32652e-009	3.02989e-009
5	4.32680e-013	2.15553e-013	2.17127e-013

It's obvious that the parallel partition algorithm and the traditional serial algorithm reach almost the same results. It can be proved that the convergence ability of the proposed parallel algorithm compares to the traditional ones. In addition, the traditional algorithm reaches convergence after 3.2 ms, and the the parallel partition algorithm reaches convergence after 2.6 ms. It's proved that the parallel algorithm can significantly improve the convergence speed. Predictably, when the number of nodes increases, the parallel algorithm can save a lot of time.

Conclusions

The algorithm in this paper is based on Newton-Raphson power flow algorithm and two layer tree structure. After reasonable network partition, in accordance with the thinking of "upward substitution" and "downward calculation", solve the voltage correction of nodes. The parallel partition algorithm mentioned above just conducts linear transformation in the primary Jacobi matrix and does not involve simplification and approximation calculation, nor to introduce extra variables like virtual current. So the results are the same as traditional algorithm whether the grid is complex or not. According to the simulation results, convergence speed of the parallel partition algorithm is superior to the traditional algorithm.

Acknowledgements

This work was financially supported by National Natural Science Foundation (51277067) and Central University Foundation (2015XS03).

References

[1] Tinney W F, Hart C E: *Power flow solution by Newton's method*. IEEE Transactions on Power Apparatus and Systems, 86(196), p. 1449-1460.

[2] Yao Y, Wang D, Chen Y, et al: *The effect of small impedance branches on the convergence of the Newton Raphson power flow*//The 3rd International Conference on Electric Utility Deregulation and Restructuring and Power Technologies. Nanjing(China) : Southeast University, 2008, p. 1141-1146.

[3] Braz L M C, Castro C A, Murari C A F: *A critical evaluation of step size optimization based load flow methods*. IEEE Transactions on Power Systems, 15(2000, p. 202-207.

[4] Jean-Jemeau R, Chiang H D: *Parameterizations of the load flow equations for eliminating ill-conditioning load flow solutions*. IEEE Transactions on Power Systems, 8(1993), p. 1004-1012.

[5] Xue Wei, Shu Ji wu, et al: *Cluster-based parallel simulation for power system transient stability analysis*. Proceedings of the CSEE, 23(2003), p. 38-43.

[6] Yang Fenghong, Tang Yun, Luo Ping: Solving load flow equations with block weighted inexact Newton method. Tsinghua Science and Technology, 43(2003), p. 1695-1698.

[7] Huang Yanquan, Xiao Jian, Liu lan: *A coordinational parallel algorithm for power flow calculation based on branch cutting*. Power system Technology, 30(2006), p. 21-25