Determination of Network Topology by Matrix Partial Multiplication

Yubin Yao, Dan Wang and Zhiliang Wu

School of Marine Engineering
Dalian Maritime University
Dalian, China
yaoyubin@dlmu.edu.cn

Abstract—Slowness is a crucial factor to prevent matrix method from its practical use in network topology. The matrix method which gets full connectivity matrix by multiplying the adjacency matrix repeatedly and determines connective sets by comparing or scanning rows of the full connectivity matrix is very time-consuming. Nodes in a connective set have same rows in the full connectivity matrix, the connective set can be determined by the first row of these same rows. Calculation of other rows whose connective set is fixed by the preceding rows is unnecessary. Once a node is certain in an exist connective set, the calculation of the row related to the node can be stopped. Based on the above consideration, a network topology method by matrix partial multiplication is presented. Row comparing or scanning is unnecessary in the presented method, and sparse matrix techniques, connectivity matrix elements immediately updating, optical node numbering are also used in the presented method. All these measures greatly decrease calculation and speed up the network topology. A practical network being analyzed by the proposed method is presented, and the results prove the effectiveness of the proposed method.

Keywords- Network topology; adjacency matrix; connectivity; full connectivity matrix; matrix partial multiplication

I. INTRODUCTION

The power system network topology algorithm is a basic and critical tool which furnishes data for many power system studies such as state estimation, power flow, contingency analysis, and dispatcher training simulator. It establishes the bus/branch model for network analysis from the switch/node model of power systems. Being algorithm about connectivity of graph in graph theory, the network topology algorithm can be accomplished by the search method [1-4], matrix method [5-9], and method of solving equations [10-11]. The matrix method implements the network topology by the operation of the adjacency matrix which represents the relationship of the nodes. The matrix method needs more computer storage and computer time, which hinders its practical usage.

To determinate connectivity the matrix method computes the $n-1$ power to adjacency matrix to get full connectivity matrix. The full connectivity matrix can be obtained by multiplying the adjacency matrix $n-2$ times for systems with n nodes at the worst, which is very time-consuming. The number of matrix multiplication decreases

to maximal $\log_2(n-1)$ times by squaring the connectivity matrix successively [5]. In fact the number of matrix multiplication can be reduced if the computation is ceased as soon as full connectivity matrix is achieved. If the element of the connectivity matrix and its symmetry element are updated and used immediately after the new one is calculated, the full connectivity matrix is achieved by only two times of the matrix multiplication [7], or by only one time of matrix multiplication we can acquire a connectivity matrix which is not a full connectivity matrix but can reveal network topology sufficiently, and network topology is then gained by reverse row sweep [8]. Both the above two methods increase the speed of network topology greatly. The matrix method obtains the connectivity network by multiplication of matrices, the connectivity matrix as one multiplier is dense, the other multiplier being adjacency matrix is sparse, and the sparse matrix techniques can apply to this multiplier, the sparse matrix techniques speed up the multiplication greatly [9].

The nodes in a connective set have identical rows in the full connectivity matrix, so we can find the connective set by the first row of these same rows. Computation of other rows whose connective set is fixed by the preceding rows is unnecessary. Once a node is certain in a connective set which includes preceding nodes, the calculation of the row related to the node can be stopped. Based on the above consideration, a new method network topology by matrix partial multiplication is presented. And many other measures which can greatly speed up network topology such as sparse matrix techniques, connectivity matrix elements immediately updating, optical node numbering are also used in the presented method. A practical network with 7097 physical nodes is analyzed by the proposed method, and the results prove the effectiveness of the proposed method. For this network, the running time is 0.016s for the proposed method.

II. ANALYSIS ON NETWORK TOPOLOGY BY MATRIX METHOD

A. Network Topology

The purpose of network topology is to establish the bus/branch model for other network analysis such as power flow, state estimation. It includes two main sections:

(a) Substation configuration: to form buses from nodes according to the connection of the nodes by closed switches (circuit breaks or disconnectors).

(b) Network configuration: to form islands from buses according to the connection of the buses by live branches (transmission lines, transformers, and etc.).

Although the purpose and the object of substation configuration and network configuration are different, they involve same topic in graph theory named connectivity of graph. Because the procedure of substation configuration is only carried out at the substation voltage levels and the number of the nodes in each substation voltage level is small, tracing in a substation voltage level is very fast. But searching scope for network configuration is entire network.

B. Connectivity and Connectivity Matrix

The connection of nodes in a network can be represented by adjacency matrix. For a network with n nodes, the adjacency matrix A is an $n \times n$ square matrix whose element a_{ij} is one if node i and node j connect directly and a_{ij} is zero if node i and node j do not directly connect. The adjacency matrix represents the first level of the network connectivity.

If the adjacency matrix is multiplied by itself, the resulting matrix T which is named as the connectivity matrix represents the second level of the network connectivity, which the element t_{ij} being one indicates that node i and node j are directly connected or connected via another node. We can obtain at best the $(n-1)$th level of the network connectivity for a network with n nodes, which is called as full connectivity matrix.

C. The Matrix Method

The matrix method is described as follows:

$$T^{(k+1)} = T^{(k)} \cdot A \qquad (1)$$

where A is the adjacency matrix, T is the connectivity matrix, the superscript (k) denotes the kth level connectivity matrix.

The adjacency matrix represents the first level of the network connectivity, so $T^{(1)} = A$.

Equation (1) is repeated until the adjacency matrix to the power $(n-1)$. By squaring the connectivity matrix, we can reach the $(n-1)$th power of the adjacency matrix rapidly, so the following equation is also used to obtain the full connectivity matrix:

$$T^{(2k)} = T^{(k)} \cdot T^{(k)} \qquad (2)$$

At the worst we need multiply $n-2$ times to get the full connectivity matrix with (1) or $\log_2(n-1)$ times with (2). In fact it does not need so much times of matrix multiplication, since when the successive connectivity matrices are same, i. e. the element of the connectivity matrix does not change any more, the full connectivity matrix is reached.

The connectivity matrix resulting from multiplication of the matrices is dense one, and the degree of the density increased rapidly along with the number of matrix multiplication. New connectivity matrix is obtained by (1) or (2). The multipliers are both dense matrices in (2), and one of the two multipliers is dense matrix in (1), such the sparse matrix techniques are not applied to the matrix method.

D. The Sparse Matrix Method

Multiplication of two dense matrices is very time-consuming, therefore the running time of the matrix method is very long and unbearable. In (1) the connectivity matrix as one multiplier is dense, but the adjacency matrix as the other multiplier is sparse and unchanged, thus the sparse matrix techniques can apply to this multiplier.

In (1), when the adjacency matrix is stored in compact form, we can apply the sparse matrix techniques to matrix method.

a) Compact Storage of the Adjacency Matrix

For compact storage of the adjacency matrix, we can use the following two arrays:
(a) AC is the column number of each non-zero element.
(b) AR is the location in array AC for the first non-zero element in each row of the adjacency matrix.

Compact storage of the adjacency matrix saves much computer storage.

b) Computation of the Connectivity Matrix

When the connectivity matrix is calculated by (1), and considering symmetry of the adjacency matrix, the element of the connectivity matrix is as follows:

$$t_{ij}^{(k+1)} = \sum_{m=1}^{n} t_{im}^{(k)} a_{jm} \qquad (3)$$

The elements of the connectivity matrix are stored with an $n \times n$ array, and the adjacency matrix is stored by compact storage.

Since the adjacency matrix is sparse one, lookup all elements in a given row is unwanted, we only need pay attention to a few $t_{im}^{(k)}$ whose column number is same as that of a_{jm}. If any of these $t_{im}^{(k)}$ is one, and then $t_{ij}^{(k+1)}$ will be one and further computation is needless. The flow chart of the computation of the element $t_{ij}^{(k+1)}$ of the connectivity matrix is shown in Fig .1.
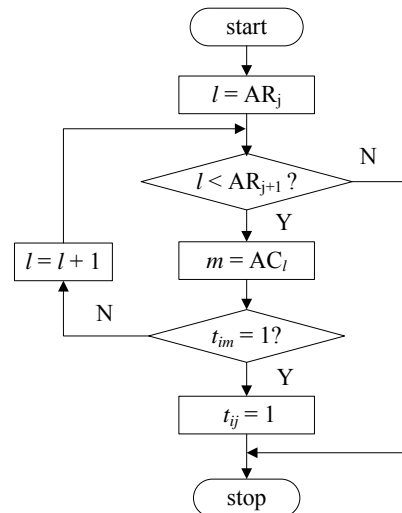

Figure 1.   Flow chart of formation of the element of the connectivity matrix

III. Determinate Connectivity by Matrix Partial Multiplication

The matrix method which gets full connectivity matrix by multiplication of the matrices is very time-consuming.

The rows of the full connectivity matrix belong to the same connective set are identical, so we can find a connective set by just sweeping its first row rather than all the rows.

From (3), it needs only the old elements in the ith row of the connectivity matrix to calculate new elements in the ith row of the connectivity matrix, the other rows have nothing to do with formation of the elements in ith row. So the elements of every row of the connectivity matrix can be calculated independently.

During multiplication of the matrices, once a row is certain to belong to same connective set with the preceding row, calculation of the elements of the row is unnecessary and can be stopped. So calculation of many rows can be cancel by the presented method. Because many rows of the full connectivity matrix are incomplete, the real full connectivity matrix is not achieved. We call the resulting matrix as quasi full connectivity matrix.

A. Formation of the Quasi Full Connectivity Matrix

The Flow chart of the formation of the quasi full connectivity matrix is showed as Fig .2. To form a new connectivity matrix, we want know if an element can be changed to 1, so we just calculate the elements whose old value is 0. The elements whose old value is 1 already need not be calculated.

When the elements in one row of the connectivity matrix is being formed, only the elements right to the principal diagonal need be calculated, that is we just need determinate whether one node connects to the nodes whose number is great than that of the node, the connection with the nodes whose number is less has be justified when the preceding rows is calculated. So only the elements right to the principal diagonal need be calculated and updated immediately.

In Fig .2, $Change$ recodes the variation of the connectivity matrix. If the elements of the connectivity matrix do not change, then $change = 0$, which means the quasi full connectivity matrix is obtained.

Array Group[] records the numbers of the groups nodes belong to. Group[i] = k ($k \neq 0$) means that node i belongs to the kth group; Group[i] = 0 means that node i does not belong to any group yet, a new group which includes node i will be created. During first multiplications, a connective set may include many groups, groups merge along with the procedure, and at last there is only a group in a connective set when the quasi full connectivity matrix is obtained.

B. Determination of Connective Set

When the quasi full connectivity matrix is obtained, Group[] are the numbers of the connective sets nodes belong to. If Group[i] = k ($k \neq 0$), node i belongs to the kth connective set. So after the matrix partial multiplication is done, the connective sets are also decided, the row sweep is unnecessary to the presented method.
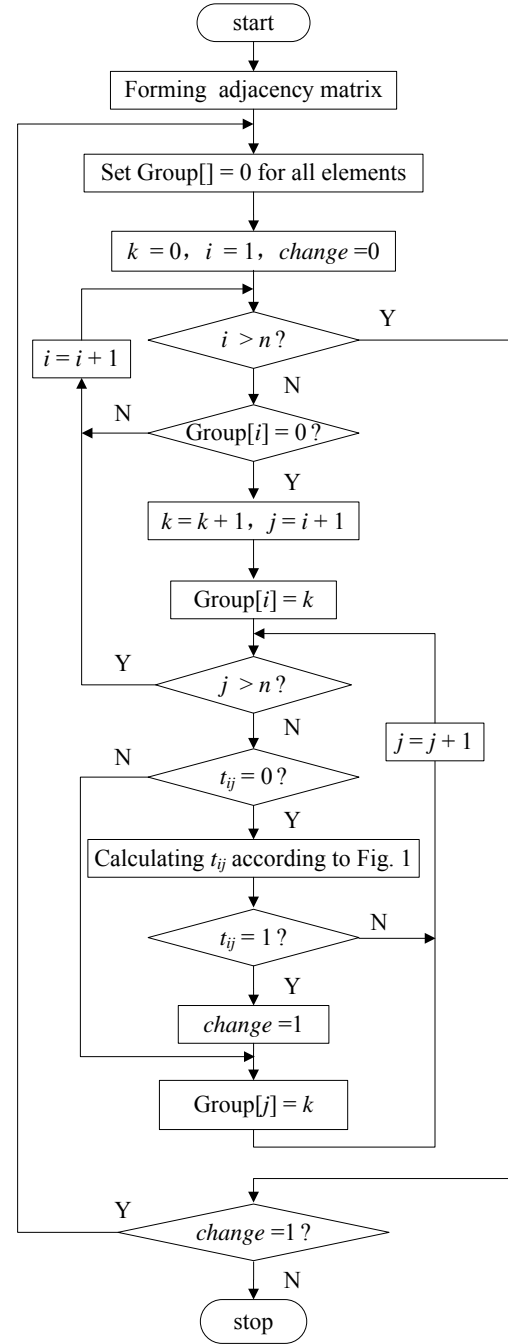

Figure 2. Flow chart of the formation of the quasi full connectivity matrix

IV. Case Study

A real network named Hangzhou network in China is used to testify our conclusions. There are 187 stations, 715 busbars, 7329 switches, 318 transmission lines, 127 two-winding transformers, 123 three-winding transformers, 11 series reactors, 232 shunt capacitors, and 27 shunt reactors in the network. There are total 7097 nodes, and 825 branches including transmission lines, series reactors and transformers (3 branches for a three-winding transformer).

The study is fulfilled on a personal computer with an Intel Pentium process of 1.10 GHz. The procedure of substation configuration is limited at the substation voltage

levels, while searching scope for network configuration is entire network. 957 buses and 49 islands are found. Only one island of these islands is live. There are 704 buses in the live island, and 122 buses in dead islands, the other buses are isolated ones.

The network topology on this network was carried out by four different methods:

Method 1: The full connectivity matrix is determined by multiplying the adjacency matrix repeatedly.

Method 2: The *full* connectivity matrix is determined by squaring the connectivity matrix repeatedly.

Method 3: The full connectivity *matrix* is determined by multiplying the adjacency matrix repeatedly, and sparse matrix techniques are applied.

Method 4: The presented *method*.

A. Comparison of the Number of Matrix Multiplication

The number of matrix multiplication required by different methods in the network configuration is listed in Table I. The number of matrix multiplication includes one more time for detecting whether the full connectivity matrix is reached.

TABLE I. The Number of Matrix Multiplication in Network Configuration

Methods	Number of matrix multiplication
Method 1	23
Method 2	6
Method 3	6
Method 4	9

The presented method needs more times of matrix multiplication than the method of squaring the connectivity matrix repeatedly or the method with sparse matrix techniques, but greatly less than the method of multiplying the adjacency matrix repeatedly.

B. Comparison of Running Time for Different Methods

The running time required by different methods is listed in Table II.

TABLE II. Computer Time Required by Different Methods

Methods	Running time (s)		
	substation configuration	network configuration	Total running time
Method 1	0.546	81.938	82.484
Method 2	0.329	24.359	24.688
Method 3	0.016	0.078	0.094
Method 4	0.008	0.008	0.016

As showed in Table II, the presented method is greatly fast than the method implements the full connectivity matrix by multiplying the adjacency matrix repeatedly or the method implements the full connectivity matrix by squaring the connectivity matrix repeatedly. It is also fast than the method with sparse matrix techniques, although

needs little more times of matrix multiplication to reach the quasi full connectivity matrix.

For all these method, most of the running time is consumed in network configuration because the matrices in network configuration are large while those in substation configuration are small.

V. Conclusion

The matrix methods which form full connectivity matrix by multiplication of the matrices are very time-consuming. The nodes belong to same connective set have identical rows in full connectivity matrix; the connective set can be found by sweep its first row in the full connectivity matrix. The first row of a connective set in the full connectivity matrix is enough, calculation of any of the other rows belong to the same connective set can be cancel if it has already found that the row belong to the connective set determined by receding row.

So network topology can be implemented by matrix partial multiplication. Study on a large practical network shows that the proposed method is very fast and suitable for real-time operation.

References

[1] R. J. Trudeau. Introduction to graph theory. New York: Dover Publications, Inc. 1993.

[2] M. Prais and A. Bose. "A topology processor that tracks network modifications over time," IEEE Trans. on Power Systems, vol. 3, no. 3, Aug. 1988, pp. 992-998.

[3] P. D. Yehsakul and I. Dabbaghchi. "A topology-based algorithm for tracking network connectivity," IEEE Trans. on Power Systems, vol. 10, no. 1, Feb. 1995, pp. 339-345.

[4] Yubin Yao, Wenzhuan Jin and Li Jin. "Fast network topology method for a distribution network," (in Chinese), Relay, vol. 33, no. 19, Oct. 2005, pp. 31-35.

[5] F. Goderya, A. A. Metwally and O. Mansour. "Fast detection and identification of islands in power networks," IEEE Trans. on Power Apparatus and Systems, vol. 99, no. 1, Jan. / Feb. 1980, pp. 217-221.

[6] Wei Zheng, Yuncheng Zhou, Tongyu Xu and Yingli Cao. "Electric connectivity analyzing for 10kV distribution network based on GIS," Proceedings of the APPEEC2010, March, 2010, pp. 1-5.

[7] Yubin Yao, Jian Xuan, Na Yu, Dan Wang and Zhiliang Wu. "Determination of network topology by quasi-square of the connectivity matrix," (in Chinese). Power System Protection and Control, vol. 39, no. 5, March, 2011, pp. 31-34, 40.

[8] Yubin Yao. "Determination of network topology by fast quasi-square of the adjacency matrix," (in Chinese). Power System Protection and Control, vol. 40, no. 6, March, 2012, pp. 17-21, 29.

[9] Yubin Yao, Shuangli Ye, Zhiliang Wu and Dan Wang. "Determination of network topology by the matrix method with sparse matrix techniques," (in Chinese). Power System Protection and Control, vol. 39, no. 23, Dec. 2011, pp.1-5, 10.

[10] Yubin Yao, Dan Wang, Zhiliang Wu and Weike Xu. "Network topology analysis by solving equations," (in Chinese), Electric Power Automation Equipment, vol. 30, no. 1, Jan. 2010, pp. 79-83.

[11] Yubin Yao, Guoshun Zhou, Hong Li and Dan Wang. "A network topology method by solving logic equations," Proceedings of the SUPERGEN2009, April, 2009, pp. 1-6.