

文章编号: 1000-3673 (2001) 02-0010-03

# 基于关联矩阵的电网拓扑辨识

王湘中<sup>1</sup>, 黎晓兰<sup>2</sup>

(1. 株洲工学院电气系, 湖南省 株洲市 412008; 2. 株洲电业局, 湖南省 株洲市, 412001)

## TOPOLOGY IDENTIFICATION OF POWER NETWORK BASED ON INCIDENCE MATRIX

WANG Xiang-zhong<sup>1</sup>, LI Xiao-lan<sup>2</sup>

(1. Dept. of Electrical Engineering, Zhuzhou Institute of Technology, Zhuzhou 412008, Hunan Province, China;

2. Zhuzhou Electric Power Co. Ltd., Zhuzhou 412001, Hunan Province, China)

**ABSTRACT:** An algorithm of topology identification based on incidence matrix for power networks is presented. This algorithm denotes the basic topology of power network by node-branch incident matrix, defines an "AND-OR" multiplication of two matrices, then use the transmission characteristics of connectivity to implements the topological identification of power network. On this basis, using the node-branch incidence matrix and the symmetry of node-node connectivity matrix, the technique to quicken the calculation speed and its implementation method is put forward. This algorithm can be implemented either by programming in assemble language or high level language, or by hardware such as single chip processor system or ASIC.

**KEY WORDS** topology identification; incidence matrix; connectivity matrix

**摘要:** 提出了基于关联矩阵的电网拓扑辨识算法。该算法使用节点-支路关联矩阵表示电网络的基本拓扑结构, 定义了矩阵的“与-或”乘法运算, 利用连通性的传递性质, 实现对电网络的拓扑辨识。在此基础上, 利用节点-支路关联矩阵和节点-节点连通矩阵的对称性, 提出了加快计算的技术和实现方法, 该算法既可以通过汇编语言或高级语言编程实现, 也可以由单片机系统或 ASIC 等硬件方法实现。

**关键词:** 电网拓扑辨识; 关联矩阵; 连通矩阵

中图分类号: TM732 文献标识码: A

## 1 引言

在电力系统调度员培训仿真系统和电力系统状态估计中, 都需要对电网的网络拓扑结构进行辨识。在各种辨识算法中, 广泛采用搜索算法<sup>[1~3]</sup>, 其中包括广度搜索算法和深度优先搜索算法以及各种变形, 这些搜索算法一般在计算机辨识过程中, 需要完全占用 CPU 的时间。本文提出的基于关联矩阵法的主接线结构辨识算法大部分工作可以由单片机系

统构成的前端机或 ASIC 完成 (也可以完全由软件完成), 节省大量的 CPU 时间, 增强结构辨识算法的实时性。

## 2 主接线拓扑辨识原理

### 2.1 主接线的描述

根据网络拓朴理论, 对于一个任意的拓朴网络, 可以用节点-支路关联矩阵来描述其拓朴结构, 而对于一个电力系统主接线图, 可以把它抽象成为一个拓朴图来描述, 如对图 1(a)所示的主接线, 把主接线的节点作为拓朴图的节点, 把开关元件作为拓朴图的支路 (当开关闭合时该支路连通, 开关断开时该支路断开), 对于单个变电站, 把母线的每一进出线连接点也作为节点, 节点-支路关联矩阵  $A = [a_{ij}]$  的每个元素表示意义如下:

$a_{ij}$  表示节点  $i$  与支路  $j$  的关联值 (联通性), 当节点  $i$  与支路  $j$  相联时  $a_{ij} = 1$ , 否则  $a_{ij} = 0$ 。显然, 当支路  $j$  的开关断开时, 支路  $j$  与原来与其相联的 2 节点不再相联, 因而与这 2 节点的关联值应为 0。

图 1(a)是当所有开关都闭合时的状态, 此时节点-支路关联矩阵 (称为原始节点-支路关联矩阵) 为

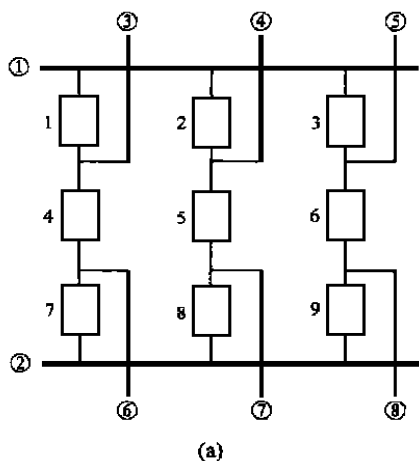
$$A_0 = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

如果断开其中的某些开关,如图1(b)所示,此时对应的节点-支路关联矩阵 $A$ (称之为当前节点-支路关联矩阵或简称为节点-支路关联矩阵)为

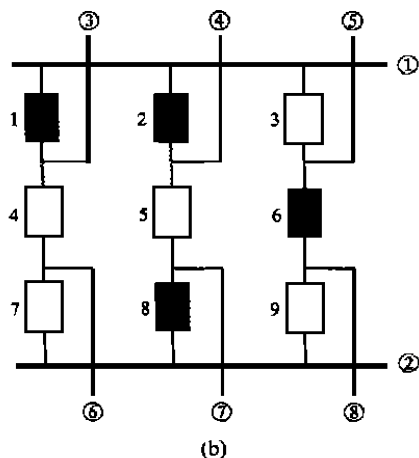
$$A = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

定义开关状态矢量 $S = [s_j]$ ,  $s_j$ 与开关 $j$ 的状态相对应。当开关闭合时,  $s_j = 1$ , 开关断开时,  $s_j = 0$ 。这样,  $S = [0, 0, 1, 1, 1, 0, 1, 0, 1]$ 。  $A$ 就是 $A$ 的每一行与 $S$ 的各个对应元素进行“与”运算后得到的。

用同样的方法,也可以定义支路-节点关联矩阵 $B = [b_{ij}]$ , 显然,  $A = B^T$ 。



(a)



(b)

□表示开关合位 ■表示开关开位

图1 3-断接线型式

Fig. 1 An example of primary network

## 2.2 主接线的拓扑辨识

对电力网络主接线的辨识,实际就是要找出网络中节点间的连通关系。显然,这种连通关系是可以传递的,即

① 如果 $i$ 节点与 $j$ 支路相联,同时,支路 $j$ 又与节点 $k$ 相联,则节点 $i$ 与节点 $k$ 是相联的;

② 如果节点 $i$ 与节点 $k$ 相联,而节点 $k$ 又与节点 $l$ 相联,则节点 $i$ 与节点 $l$ 也是相联的。

定义节点-节点连通矩阵(它与网络图论中的节点-节点关联矩阵有区别) $C = [c_{ij}]$ , 当节点 $i$ 与节点 $j$ 连通时,  $c_{ij} = 1$ , 不连通时  $c_{ij} = 0$ , 显然 $C$ 是对称矩阵。以上连通关系的传递性质可以表示如下:

① 若  $a_{ij} = 1, b_{jk} = 1$ , 则  $a_k = a_{ij} \cap b_k = 1$ ;

② 若  $c_{ik} = 1, c_{kl} = 1$ , 则  $c_{il} = c_{ik} \cap c_{kl} = 1$ 。

对于具有 $m$ 个节点, $n$ 条支路的网络,定义以下矩阵乘法运算:

$$C = A \cdot B \quad (1)$$

$$\text{其中 } a_{ij} = \bigcup_{k=1}^n (a_{ik} \cap b_{kj}) \quad (2)$$

$\cap$  表示“与”运算;  $\cup$  表示“或”运算。

那么当 $A$ 为节点-支路关联矩阵, $B$ 为支路-节点关联矩阵时,  $a_{ij}$ 将表示节点 $i$ 与节点 $j$ 通过任一支路的关联情况。当然,只要节点 $i$ 与节点 $j$ 有一条支路相联,则  $a_{ij} = 1$ 。  $C = [c_{ij}]$ 表示了节点与节点之间的连通性,称为节点-节点连通矩阵。这时的节点-节点连通矩阵仅仅表示了节点之间的直接连通性质,把它称为1级节点-节点连通矩阵,并记为 $C^{(1)}$ 。由于连通性的传递性质,可以通过用1级节点-节点连通矩阵 $C^{(1)}$ 进行上面定义的矩阵乘法运算,得到2级节点-节点连通矩阵

$$C^{(2)} = C^{(1)} \cdot C^{(1)} \quad (3)$$

2级节点-节点连通矩阵 $C^{(2)}$ 在1级节点-节点连通矩阵 $C^{(1)}$ 的基础上,运用连通关系的传递性,把节点之间的部分间接连通关系也表示出来了,用 $C^{(2)}$ 再自乘得到 $C^{(3)}$ , ..., 直到 $C^{(n)} = C^{(n-1)} \cdot C^{(n-1)}$ 。这时,通过传递,所有连通的节点之间的关联值都是1了,而不连通的节点间关联值都为0。

如如图1(b)

$$C^{(1)} = A \cdot B = A \cdot A^T =$$

$$C^{(1)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$C^{(2)} = C^{(1)} \cdot C^{(1)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

$$C^{(3)} = C^{(2)} \cdot C^{(2)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

再重复进行以上运算,将发现  $C$  不再变化,这时已把网络中的所有连通关系都表示出来了。由  $C$  可见,节点连接成以下几组:

节点 1 5 为一组;节点 2 3 6 8 为一组;节点 4 7 为一组

### 3 算法的简化与加速

#### (1) 利用 $B = A^T$ 化简

在式 (1) 中由于  $B = A^T$ , 所以式 (2) 可以转化为

$$a_{ij} = \bigcup_{k=1}^n (a_{ik} \cap b_{kj}) = \bigcup_{k=1}^n (a_{ik} \cap a_{jk}) \quad (4)$$

即  $a_{ij}$  就是  $A$  的第  $i$  行与第  $j$  行按位求“与”后的值,如果其中有一位为 1,则  $a_{ij} = 1$ ;

#### (2) 利用 $C$ 的对称性

$C^{(2)} = C^{(1)} \cdot C^{(1)}$  等运算过程,实际上就是节点间连通关系的传递过程。由于  $C$  的对称性,

$$a_{ij} = \bigcup_{k=1}^m (a_{ik} \cap c_{kj}) = \bigcup_{k=1}^m (a_{ik} \cap c_{jk}) \quad (5)$$

即  $a_{ij}$  可由  $C$  的第  $i$  行与第  $j$  行按位求“与”后再“或”,如果其中有一位为 1,则  $a_{ij} = 1$ ;

同时,由于在节点连通关系的传递过程中,如果节点  $i$  与节点  $j$  在传递前是连通的,则在传递运算后仍然是连通的。因此,如果  $c_{ij}$  在传递运算前是 1,则

$a_{ij}$  无需再进行传递运算

为了充分利用前一步连通性传递结果,更进一步加快传递速度,在对  $C$  的自乘过程中,可以直接对  $C$  中为 0 的元素进行运算,并用运算结果直接修改该元素。

实际上,在求连通关系的过程中,只要进行以下过程即可

$$c_{ij} = \bigcup_{k=1}^n (a_{ik} \cap a_{jk}) \quad i = 1, \dots, m; j = 1, \dots, m$$

$$c_{ij} = \bigcup_{k=1}^m (a_{ik} \cap c_{jk}) \quad i = 1, \dots, m; j = 1, \dots, m$$

本算法的框图如图 2 所示。

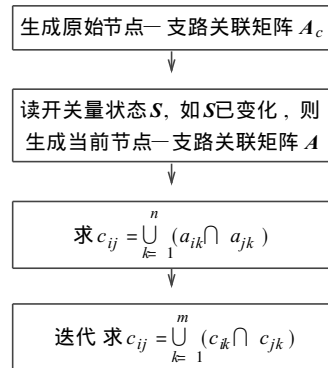


图 2 算法框图

Fig. 2 Flow chart of the algorithm

#### (3) 节点编号的优化

为了加快连通关系的传递速度,应把连接开关较多的节点编号在前,如母线节点。因为,这种传递运算实际是从低节点号开始的,编号较小的节点参加传递运算的次数较多,所以,把连接支路较多的节点编号在前有利于加快传递速度

#### (4) 消去中间节点和开关支路

对网络进行拓扑辨识中,不需要“过多关心”的中间节点和开关支路可以消去,从而减少节点和支路数目,如图 3 所示。图 3 的网络实际已经把隔离开关支路和中间节点消去了。

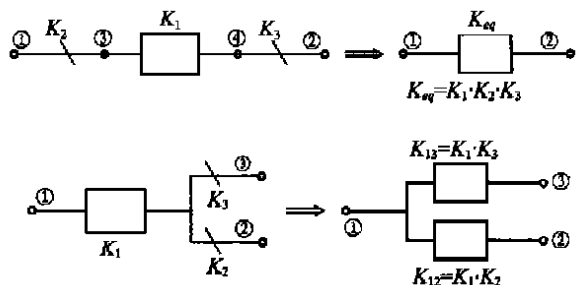


图 3 消去中间节点和支路

Fig. 3 Removing of middle nodes and branches

(下转第 16 页 continued on page 16)

以改善稳定性,而且较线性最优控制器更显著。但在实现上,模糊神经网络控制器比线性最优控制器要困难些。

## 6 结论

可控制电阻的电阻值可以随时连续调节,为改善电力系统稳定性提供了新的手段。本文应用一种改进的模糊神经网络控制方法,设计了可控制电阻的控制器。仿真计算结果表明,可控制电阻的模糊神经网络控制对静态稳定性和暂态稳定性均具有良好效果。

## 参考文献:

- [1] Ellis H M, Hardy J E, Blythe A L, *et al.* Dynamic stability of the Peace River transmission system [J]. IEEE Trans on PAS, 1966, 85(6).
- [2] Shelton M L, Mittelstadt W A, Winkelman P F, *et al.* Bonneville power administration 1400 MW braking resistor [J]. IEEE Trans on PAS, 1975, 94(1).
- [3] 韩英铎. 电力系统中的三项前沿课题——柔性交流输电技术,

智能控制. 基于GPS的动态安全分析与监测系统[J]. 清华大学学报, 1997, 37(7).

- [4] 李海峰, 李乃湖. FACTS装置用于电力系统稳定控制的综述[J]. 电力系统自动化, 1998, 22(9).
- [5] 姜齐荣, 闵勇, 韩英铎. 基于人工神经网络的励磁调节、快速汽门和电阻制动的协调控制[J]. 清华大学学报, 1998, 38(7).
- [6] Rahim A H M A, Almgir D A H. A closed-loop quasi-optimal dynamic braking resistor and shunt reactor control strategy for transient stability [J]. IEEE Trans on PWRs, 1988, 3(3).
- [7] 李士勇. 模糊控制、神经控制和智能控制论[M]. 哈尔滨: 哈尔滨工业大学出版社, 1996.
- [8] 鞠平, 付蓉, 倪辉. 可控串联补偿的模糊神经网络控制[J]. 电力系统自动化, 2000, 24(2).

收稿日期: 2000-04-27; 改回日期: 2000-07-31

作者简介:

付蓉, 硕士研究生;

韩敬东 (1962-), 男, 工程师, 从事电力系统动态模拟工作;

鞠平 (1962-), 男, 博士, 教授, 从事电力系统模拟与控制方面的研究工作;

倪辉, 硕士研究生。

(编辑 陈定保)

(上接第12页 continued from page 12)

## 4 算法的实现

### (1) 用C语言实现

用软件编程实现以上算法时, 虽然矩阵  $A$  和  $C$  的存储量分别是  $m \times n$  和  $m \times m$ , 但用C语言编程时可以充分利用C语言的位运算功能, 把一个字与16位的位结构作为共用体, 这16位与16条支路(或节点)对应, 一个  $m \times n$  矩阵可以用  $m \times (n/16 + 1)$  个字表示; 另外, 矩阵的行间“与”运算用字节按位“与”运算, 这样运算速度可以大大加快, 使用32位或64位计算机效果更好。

### (2) 用汇编语言实现

由于本算法很简单, 用汇编语言编程实现也很方便, 笔者已用 MCS-8051 单片机开发了能辨识 64 节点  $\times$  64 支路的专用辨识模块。

### (3) 用硬件实现

本算法的最大特点是能用简单的 ASIC 实现。

## 5 分布式拓扑辨识法

电力网络是由许多发电厂和变电站组成的一个多电压等级网络, 在每一个发电厂和变电站安装一辨识模块, 再在主站安装系统辨识模块, 可以实现分

## 层分布式辨识

## 6 结论

基于关联矩阵的电网拓扑辨识算法有以下特点

(1) 算法简单, 速度快, 可以用多种方法实现。在用硬件实现时, 很少占用 CPU 的时间。

(2) 可以很方便地实现分层分布式的拓扑辨识, 特别适用于电力系统的实时辨识。

## 参考文献:

- [1] 于尔铿. 电力系统状态估计 [M]. 北京: 水利电力出版社, 1985.
- [2] 万华, 李乃湖, 陈珩. 用于培训仿真的快速网络拓扑形成与跟踪、修正算法 [J]. 电力系统自动化, 1995, 19(5).
- [3] 朱文东, 刘广一, 于尔铿, 等. 电力网络局部拓扑的快速算法 [J]. 电网技术, 1996, 24(3).
- [4] 秦红霞, 董张卓, 孙启宏, 等. 基于面向对象技术的变电站故障诊断及恢复专家处理系统 (二) 故障诊断及恢复处理 [J]. 电力系统自动化, 1997, 21(2).

收稿日期: 2000-04-24; 改回日期: 2000-08-31

作者简介:

王湘中 (1966-), 男, 硕士, 讲师, 从事电力系统仿真与变电站自动化系统的研究与开发;

黎小兰 (1964-), 女, 高级工程师, 从事变电站运行管理工作。

(编辑 陈树勇)