

几种电力网络图的连通路径拓扑算法研究

吕昊, 付立军, 叶志浩, 谢桢

(海军工程大学舰船综合电力技术国防科技重点实验室, 湖北 武汉 430033)

摘要: 网络拓扑分析是电力系统仿真和分析计算的基础。同一图论问题常常有多种算法, 本文用图的形式来描述电力系统模型的结构, 将一个具体的电力系统抽象分解为一些线段及联结这些线段的点的集合。并且列举了图连通路径的搜索方法: 邻接矩阵法、树搜索法、连接矩阵余子式法。这几种方法都可以达到网络拓扑的目的。

关键词: 图论; 开关网络; 拓扑; 电力网络; 连通路径

Some topology algorithm research of power network connectivity path

Lü Hao, FU Li-jun, YE Zhi-hao, XIE Zhen

(National Key Laboratory for Vessel Integrated Power System Technology, Naval University of Engineering, Wuhan 430033, China)

Abstract: Power system simulation and calculation are based on network-topological analysis. There is much algorithm to solve the same of graph problem. This paper describes structure of network model as graph. And then, a material network is abstractly disassembled edge-set and vertex-set. Connectivity path of graph is listed in this paper. Searching methods are adjacency matrix, tree searching and complementary of connection matrix. These methods can also solve topology of networks.

Key words: graph theory; switching network; topology; power system network; connectivity path

中图分类号: TM341 文献标识码: A 文章编号: 1674-3415(2009)21-0082-04

0 引言

图论作为一个新的数学分支, 发展极其迅速。近年来受计算机科学蓬勃发展的刺激, 其应用范围覆盖了从自然科学到社会科学的广阔的领域, 包括: 电力网络、控制论、计算机的程序设计、人工智能、地图着色、情报检索等^[1]。

图是抽象支路和节点的集合, 它反映图中所包含的各支路之间的联结关系, 即节点与支路的关系。在图论中, 图可表示为 $G = (V, E)$, V 表示顶点的集合, E 表示边的集合^[2]。

1 电力网络描述与图的对应

对电力网络简化成数学模型的基本思想是, 将电机、负载、器件等看成一个电气节点; 开关当成连接电气节点的支路, 节点之间由支路相连, 支路是否连通则取决于开关的状态。此处将开关作为边而不是点的优势在于减少节点的数目, 以便于减小邻接矩阵的规模和后续的重构研究中开关操作次数最少的分析。

网络电气连通性分析的主要任务是根据开关状态的变化, 形成网络元件之间的连接关系, 其实质是把母线/开关描述的物理模型转化为节点/支路描述的数学模型。电力网络的主体部分是由各设备通过导线连接所组成的, 在结构上具有明显的点线特征, 用图论方法来分析是最合理的。

图1(a)为所示电气连接, (b)为无向图, (c)为有向图。把电网中的每个电气元件看成图中的顶点, 并依次编号。

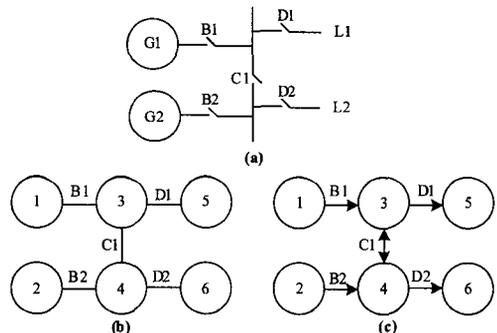


图1 简单电气连接示意图

Fig.1 Simple electric connected graph

2 图的连通性与拓朴结构

2.1 邻接矩阵

图是一种结构复杂的数据结构,表现在顶点之间的逻辑关系错综复杂。对于实际问题,需要根据具体的图结构本身的特点以及所要实施的操作,选择建立合适的存储结构。邻接矩阵是图的最主要存储方式之一,相应的邻接矩阵法优点是分析过程清晰,易于实现,数据的组织比较简单,适应性强,容易判定任意两个顶点之间是否有边相连。矩阵法理论上可以适用于任何复杂的网络结构,且在相同的网络节点规模下,节点之间的接线方式的复杂程度不会影响计算的复杂程度和运算量。但缺点是计算量大,功能单一,存储空间浪费。

所谓邻接矩阵存储结构,就是用一维数组存储图中顶点的信息,用矩阵表示图中各顶点之间的相邻关系。假设图 $G=(V, E)$ 有 n 个确定的顶点,则表示 G 中各顶点相邻关系的矩阵为一个 $n \times n$ 的矩阵

$$A=(a_{ij})$$

邻接矩阵法是基于邻接矩阵的性质,其邻接矩阵 A 中的元素 a_{ij} 取值“1”或“0”。对角线元素 a_{ii} 的值为“0”,如果两个顶点 v_i 、 v_j ($i \neq j$) 之间有支路直接相连,则 $a_{ij}=1$,称 v_i 、 v_j 为一级连通,否则 $a_{ij}=0$;若 v_i 和 v_j 之间无直接连通的支路,但它们均通过支路与节点 v_k 直接相连,实际上节点 v_i 和 v_j 是通过节点 v_k 连通的,则称 v_i 、 v_j 为二级连通。二级连通可以通过矩阵 A 的两次逻辑自乘求得。而有向连通图与无向图类似,但需要加入方向的判断。

A^m 为矩阵 A 的 m 次逻辑自乘,矩阵元素 $(a_{ij})_m$ 表示节点 v_i 、 v_j 之间从第 1 级到第 m 级的连通关系。对于有 n 个顶点的图来说,任意两个顶点之间最多有 $n-1$ 级连通。所以对邻接矩阵 A 进行 $n-1$ 次自乘运算,然后对 A^{n-1} 进行列扫描(或行扫描),相连的节点对应于线性相关的列(行)矢量,不相连的节点所对应的列(行)矢量的逻辑积为零矢量,从而实现节点间的连通性分析。

2.2 布尔矩阵的运算

如果一个矩阵,其元素的值全部为“1”或“0”,这样的矩阵称为布尔矩阵。邻接矩阵即为布尔矩阵。其运算除了可以进行一般的矩阵运算外,还进行布尔运算,算法如下:

1) 逻辑乘,用 \wedge 表示:

$$0 \wedge 0=0; 0 \wedge 1=0; 1 \wedge 0=0; 1 \wedge 1=1.$$

2) 逻辑加,用 \vee 表示:

$$0 \vee 0=0; 0 \vee 1=1; 1 \vee 0=1; 1 \vee 1=1.$$

3 搜索连通路程过程

图 1 中的邻接矩阵是

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

3.1 连通路程方法一: 矩阵内部的树搜索

方法一的过程为:如果需要查找 v_i 到 v_j 的连通路程,从需要查询的那一行,即第 i 行开始,在该行查到存在“1”的列后,将列标记下来,转化为行标,再在新的行里继续查找存在“1”的列,往复依次记录下标,直至能够搜索到所需要的那个列,即第 j 列的列数为止,以表明 v_i 到 v_j 是连通的。

如需要搜索图 1 中① \rightarrow ⑥的连通路程。只需从 A 的第 1 行向量开始,先调出

$$A_1^{(1)}=(0 \ 0 \ 1 \ 0 \ 0 \ 0)$$

查找 A_1 中的“1”向量,存在第 3 列为“1”。

记下 $a_{13}^{(1)}=1$ 。

$$A_1^{(2)}=A_1^{(1)}A=(0 \ 0 \ 1 \ 0 \ 0 \ 0) \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}=(0 \ 0 \ 0 \ 1 \ 1 \ 0)$$

$A_1^{(2)}$ 表示的是 A 自乘 1 次后 $A^{(2)}$ 的第一行。同理,求出 $A_1^{(2)}$ 的同时,紧接着查找 A 内第 3 行里的“1”元素。 $A_3^{(1)}=(0 \ 0 \ 0 \ 1 \ 1 \ 0)$ 。

结果是第 4 列和第 5 列符合条件,记下 $a_{134}^{(2)}=1$ 和 $a_{135}^{(2)}=1$ 。

$$A_1^{(3)}=A_1^{(2)}A=(0 \ 0 \ 0 \ 1 \ 1 \ 0) \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}=(0 \ 0 \ 1 \ 0 \ 0 \ 1)$$

此时再分两路:

1) $a_{135}^{(2)}=1$,求 $A_1^{(3)}$ 时,查找 A 内第 5 行中“1”元素,结果节点①—③—⑤这条路没有通往节点⑥的路径。 $A_5^{(1)}=(0 \ 0 \ 0 \ 0 \ 0 \ 0)$,因为 A 的第 5 行已无通路,无元素“1”。

2) $a_{134}^{(2)}=1$,同上法则,查到 A 里的第 4 行

第 6 列符合条件。\$A_4^{(1)} = (0 \ 0 \ 1 \ 0 \ 0 \ 1)\$ 那么就搜索到 ① → ③ → ④ → ⑥ 这一路径。

此外如果要求从节点 \$v_i\$ 出发经 \$k\$ 步所能到达的点, 只需求矩阵 \$A^k\$ 的第 \$i\$ 行元素即可。就是说从节点 \$v_i\$ 出发, 所以只需看第 \$i\$ 行的行向量, \$A^k\$ 的第 \$i\$ 行行向量用 \$A_i^{(k)}\$ 表示。求 \$A_i^{(k)}\$ 只需 \$A_i^{(k-1)}\$, 故若只计算 \$A_i^{(k)}\$, 可大大减少计算量。无须计算整个邻接矩阵 \$A^k\$ 的值。

3.2 连通路程方法二: 布尔运算自乘反推

方法二与方法一原理相同, 表述的不同。在方法一里面也有布尔自乘的过程, 但没有真正使用。只是把布尔自乘的结果作为节点连通的标记, 以表明这些点之间是连通的, 并为下一个连通节点做好准备。我们看到, \$A_4^{(3)}\$ 中的 \$a_{16}^{(3)} = 1\$ 由 \$A_4^{(2)}\$ 中的 \$a_{14}^{(2)}\$ 与 \$a_{46}^{(1)}\$ 相乘而得。即从节点 ① 出发走两步到达节点 ④ (\$a_{14}^{(2)} = 1\$), 再走一步到达节点 ⑥, \$a_{14}^{(2)} \to a_{46}^{(1)} \Rightarrow a_{16}^{(3)}\$

$$A_4^{(3)} = A_4^{(2)} A = (0 \ 0 \ 0 \ 1 \ 0 \ 0) \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} = (0 \ 0 \ 0 \ 1 \ 0 \ 0)$$

而 \$a_{14}^{(2)}\$ 是由 \$A_4^{(1)}\$ 中的 \$a_{13}^{(1)}\$ 与 \$a_{34}^{(1)}\$ 相乘而得。即从节点 ① 出发第一步到达节点 ③, 第二步到达节点 ④。\$a_{13}^{(1)} \to a_{34}^{(1)} \Rightarrow a_{14}^{(2)}\$。

$$A_4^{(2)} = A_4^{(1)} A = (0 \ 1 \ 0 \ 0 \ 0 \ 0) \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} = (0 \ 0 \ 0 \ 1 \ 0 \ 0)$$

故由 \$a_{13}^{(1)} \to a_{34}^{(1)} \to a_{46}^{(1)} \Rightarrow a_{16}^{(3)}\$, 也可知这条路径的节点 ① → ③ → ④ → ⑥。如果不强调在矩阵内部搜索过程的话, 可以这样认为: 观察 \$a_{16}^{(k)}\$ 什么时候会出现元素“1”, 经过最少的 \$k\$ 次自乘, 即 \$k\$ 为何值时 \$a_{16}^{(k)} \neq 0\$。(\$k \le n-1\$)。那么 ① → ⑥ 就连通, 且 \$k\$ 为需要走的步数。

可以总结出方法二, 如果需要查找 \$v_i\$ 到 \$v_j\$ 的连通路程, 也需要把 \$A_i^{(k)}\$ 做完。不同之处在于先把 \$A_i^{(k)}\$ 全部求出, 如果 \$a_{ij}^{(k)}\$ 有元素“1”, 则说明其中是有连通路程的。接着就可以从 \$A_i^{(k)} = A_i^{(k-1)} A\$ 这个等式里查找可能的连通路程, 得到表达式 \$a_{ik}^{(k-1)} \to a_{kj}^{(1)} \Rightarrow a_{ij}^{(k)}\$。然后是 \$A_i^{(k-1)} = A_i^{(k-2)} A\$, 一步一步直至从 \$A_i^{(2)} = A_i^{(1)} A\$ 中 \$a_{ii}^{(1)} \to a_{ii}^{(1)} \Rightarrow a_{ii}^{(2)}\$ 反

推得出结论。最终的连通路程为: 点 \$v_i \to v_l \to v_l \to \dots \to v_k \to v_j\$。

3.3 连通路程之方法三: 连接矩阵余子式

一个具有 \$n\$ 个节点的开关网络的连接矩阵是一个 \$n \times n\$ 阶的矩阵 \$C\$, 其中

$$c_{ij} = \begin{cases} c_{ij} & \text{顶点 } v_i \text{ 与 } v_j \text{ 之间有连接} \\ 0 & \text{顶点 } v_i \text{ 与 } v_j \text{ 之间没有连接} \end{cases}$$

\$c_{ij}\$ 表示为布尔变量, 是开关的连接标记, 顶点 \$v_i\$ 到 \$v_j\$ 的边权; \$c_{ii} = 0\$。连接矩阵 \$C\$ 与邻接矩阵 \$A\$ 本质是相同的。表述不同之处在于引入了开关布尔变量的概念, 将原来邻接矩阵里表示为有连接时的 \$a_{ij} = 1\$, 改为连接矩阵里的布尔变量 \$c_{ij}\$。

方法三步骤: 假设无向图 \$G\$ 的所有边权为布尔变量, 连接矩阵为 \$C\$。

1) 将 \$C\$ 与单位矩阵 \$I_{n \times n}\$ 逻辑和运算, 得到 \$(C+I)\$。

2) 划去第 \$j\$ 行和第 \$i\$ 列, 得到的子矩阵。

3) 再求其子矩阵的行列式, 并且展开。

所得子矩阵行列式的展开表达式, 即为节点 \$v_i\$ 和 \$v_j\$ 的开关函数。开关函数为路径乘积之和, 正好可以代表 \$v_i\$ 与 \$v_j\$ 之间所有可连通的路径表达式。换句话说, 欲求 \$v_i\$ 与 \$v_j\$ 的开关函数, 即它们之间所有可连通的路径表达式, 只需求 \$(C+I)\$ 的余子式 \$\Delta_{ji}\$。

由于无向图的连接矩阵是对称的, 即 \$c_{ij} = c_{ji}\$, 划去第 \$j\$ 行、第 \$i\$ 列与划去第 \$i\$ 行、第 \$j\$ 列等价, 最后所得结果是相同的。

无向图 1(c) 的连接矩阵

$$C = \begin{pmatrix} 0 & 0 & c_{13} & 0 & 0 & 0 \\ 0 & 0 & 0 & c_{24} & 0 & 0 \\ c_{31} & 0 & 0 & c_{34} & c_{35} & 0 \\ 0 & c_{42} & c_{43} & 0 & 0 & c_{46} \\ 0 & 0 & c_{53} & 0 & 0 & 0 \\ 0 & 0 & 0 & c_{64} & 0 & 0 \end{pmatrix}$$

$$C+I = \begin{pmatrix} 1 & 0 & c_{13} & 0 & 0 & 0 \\ 0 & 1 & 0 & c_{24} & 0 & 0 \\ c_{31} & 0 & 1 & c_{34} & c_{35} & 0 \\ 0 & c_{42} & c_{43} & 1 & 0 & c_{46} \\ 0 & 0 & c_{53} & 0 & 1 & 0 \\ 0 & 0 & 0 & c_{64} & 0 & 1 \end{pmatrix}$$

还是求 \$v_1\$ 到 \$v_6\$ 的所有路径, 即开关函数 \$F_{16}\$, 可以先将连接矩阵 \$C\$ 中的第 6 行和第 1 列划去, 得到一个子矩阵

$$C_1 = \begin{pmatrix} 0 & c_{13} & 0 & 0 & 0 \\ 1 & 0 & c_{24} & 0 & 0 \\ 0 & 1 & c_{34} & c_{35} & 0 \\ c_{42} & c_{43} & 1 & 0 & c_{46} \\ 0 & c_{53} & 0 & 1 & 0 \end{pmatrix}$$

$$\det(C_1) = -c_{13} \times \begin{vmatrix} 1 & c_{24} & 0 & 0 \\ 0 & c_{34} & c_{35} & 0 \\ c_{42} & 1 & 0 & c_{46} \\ 0 & 0 & 1 & 0 \end{vmatrix} =$$

$$-c_{13} \times \begin{vmatrix} c_{34} & c_{35} & 0 \\ 1 & 0 & c_{46} \\ 0 & 1 & 0 \end{vmatrix} - c_{13} \times (-c_{24}) \times \begin{vmatrix} 0 & c_{35} & 0 \\ c_{42} & 0 & c_{46} \\ 0 & 1 & 0 \end{vmatrix} =$$

$$c_{13} \times c_{34} \times c_{46} = F_{16}$$

返回图 1 中, $c_{13} - c_{34} - c_{46}$ 的连接顺序相对应的路径即为 $1 \rightarrow 3 \rightarrow 4 \rightarrow 6$ 。此方法同样适用于有向图 D , 由于加入了方向, 必须执行划去的是第 j 行、第 i 列规定。

方法三相比较繁琐的方法一和方法二, 行列式的程序编写也简单了很多, 可以将所有的连通路径都能快速表达出来, 具有普遍性和直观性。

4 结论

电力网络重构要确定一组开关的状态组合, 使网络在该组组合下运行状态最佳, 因此离不开电力网络的拓朴模型。这就需要拓朴模型能精确描述电网拓朴结构, 对拓朴模型有以下要求: 1) 可以准确地体现电力系统网络的联结关系及顶点之间的关系。2) 可以根据开关状态的变化方便地描述其运行电力网络的状态量。3) 便于计算机处理。本文介绍了三种图论中的矩阵方法来描述寻找图的连通路经, 分别为: 邻接矩阵法、树干搜索法和连接矩阵余子式法, 以达到拓朴的目的。该算法的实现, 可以为将来的电力系统保护与重构研究打下基础。

参考文献

[1] 陈树柏. 网络图论及其应用[M].北京: 科学出版社, 1982.

CHAN Shu-bai. Network Graph Theory with Applications[M]. Beijing: Science Press, 1982.

[2] GONG Yan-feng, HUANG Yan. Integrated Protection System Design for Shipboard Power System[A]. in: IEEE Electric Ship Technologies Symposium[C]. Philadelphia(USA): 2005. 237-243.

[3] 徐俊明. 图论及其应用[M].合肥: 中国科学技术大学出版社, 2004.

XU Jun-ming. Graph Theory with Applications[M]. Hefei: University of Science and Technology of China Press, 2004.

[4] Butler K L, Sarma N D R. Network Reconfiguration for Service Restoration in Shipboard Power Distribution Systems[J]. IEEE Trans on Power Systems, 2001, 16(4): 653-661.

[5] 张伯明, 陈寿孙. 高等电力网络分析[M]. 北京: 清华大学出版社, 1996.

ZHANG Bo-ming, CHEN Shou-sun. Advanced Electric Power Network Analysis[M]. Beijing: Tsinghua University Press, 1996.

[6] 卢开澄, 卢华明. 图论及其应用[M].北京: 清华大学出版社, 1995.

LU Kai-cheng, LU Hua-ming. Graph Theory with Applications[M]. Beijing: Tsinghua University Press, 1995.

[7] 蔡丽娟, 张建成. 面向对象技术的电力网络拓朴分析[J]. 电气时代, 2004, (4): 94-95.

CAI Li-juan, ZHANG Jian-cheng. Power System Network Topology Analysis Using Object-Oriented Technology[J]. Electric Age, 2004, (4): 94-95.

收稿日期: 2008-10-29; 修回日期: 2009-07-16

作者简介:

吕昊(1979-), 男, 海军工程大学电气工程系博士生, 目前主要研究电力系统重构算法。E-mail: lvhao79@163.com