第3卷 第6期
1995年12月

光学 精密工程
OPTICS AND PRECISION ENGINEERING

Vol.3 No.6
December 1995

寻找公共树的一种计算机算法

滕玉鹃
(吉林工学院自动化系,长春 130012)
乐全根
(武汉工学院电子系,武汉 430070)

摘要 用公共树转换的方法求出两个具有相同边集和相同秩的图的所有公共树,同时求出相应的公共树的符号,并给出了实现算法的程序框图。

关键词:公共树;链路图;生成回路集

1 引 言

随着科学技术的发展,图论的发展日新月异,图论的应用日益广泛,尤其是在电网络理论、开关理论及运筹学等方面的应用发展十分迅速。在电网络理论中,随着网络的结构越来越复杂,规模越来越大,它们的分析和设计更多地要依赖于计算机,在这里图论起着重要作用。而在图论中,树是最重要的子图之一。对线性网络进行拓扑分析,多数方法是将问题归结为找相关图中树的集合。完全树分析法就是通过找电流图和电压图(或对其作适当变换后的图)的所有公共树,求出有源网络的节点导纳矩阵的行列式(或一阶代数余子式),进而可以求出线性有源网络 全符号的网络函数(即网络中的全部元件参数都是用符号表示的)。同常用的树值计算法相比,其最大的优点是结果中不含相消项(设元件参数代表符号都相异)。但是,完全树法存在的一个缺点是确定公共树权的符号比较麻烦,这使得完全树法的应用及发展受到了限制。已有的方法存在的缺点是:同一个公共树可能生成多次,而且生成公共树和计算其相应的符号是分开进行的,效率低,且不便于借助计算机进行分析。

本文叙述了在两个具有相同边集和相同秩的图中,如何从一个已知的公共树经公共树转换求出其它所有的公共树,并且同时求出相应的公共树的符号。该方法便于用计算机来实现。每个公共树不会重复出现,且符号问题容易解决。

2 有源网络的拓扑公式

(1)有源网络节点导纳矩阵的行列式 Δn 的拓扑公式为:

$$\Delta n=\sum_{k}\varepsilon_k\times(G_i\text{ 和 }G_v\text{ 的公共树导纳积}) \tag{1}$$

收稿日期:1995年9月30日

其中：ε_k 代表第 k 个公共树的符号

即：有源网络的节点导纳矩阵的行列式等于网络对应的电流图 G_i 和电压图 G_V 的所有公共树的树支导纳积的代数和。

(2)设 Δ_{pq} 为节点导纳矩阵的第 p 行，第 q 列元素的代数余子式，

将电流图中对应于图 G 的 p 点同 j 点短接(j 点为图 G 的参考点)，将电压图中对应于图 G 的 q 点同 j 点短接，设这样得到的图分别是 $G_i(p=j)$ 和 $G_V(q=j)$

则 $\Delta_{pq}=(-1)^{p-q}\sum_{k'}\varepsilon_{k'}\times$〔$G_i(p=j)$ 和 $G_V(q=j)$ 的公共树导纳积〕　　(2)

其中，$\varepsilon_{k'}$ 为对应的第 k' 个公共树的符号

由此可见，求 Δ_n 和 Δ_{pq} 可以归结为寻找两个图中所有公共树及其符号的问题。

3　公共树转换原理

设 G_1，G_2 是具有相同边集和相同秩的两个图。假设 G_1、G_2 至少有一个公共树，从 t_i 到 t_j 的公共树转换是指从 G_1 和 G_2 的任一公共树 t_i 获得新的公共树 t_j 的一种运算。

定义 1：设 t_i 是 G_1 和 G_2 的任一公共树，$\bar{t}_i$ 是对应的公共补树。$B(t_i)$ 是 G_1 关于 t_i 的基本回路矩阵的主要部分，$Q(t_i)$ 是 G_2 关于 t_i 的基本割集矩阵的主要部分，关于 t_i 的加权有向图 $\hat{G}(t_i)$——链路图由下面两种运算来定义：

(1)对应于 $B(t_i)=[b_{jk}]$ 的每一行和每一列的点分别叫作 G_1 的补树边点和树边点；对应于 $Q(t_i)=[q_{jk}]$ 的每一行和每一列的点分别叫作 G_2 的树边点和补树边点。树边点用双环表示，补树边点用单环表示。当且仅当 $B(t_i)$ 的元素 b_{jk} 是非零的，存在一条有向边，从 G_1 中对应于 $B(t_i)$ 的第 k 列的树边点指向 G_1 中对应于 $B(t_i)$ 的第 j 行的补树边点，边的权为 b_{jk}，称这种有向边为 B 边；当且仅当 $Q(t_i)$ 的元素 q_{jk} 是非零的，存在一条有向边，从 G_2 中对应于 $Q(t_i)$ 的第 k 列的补树边点指向 G_2 中对应于 $Q(t_i)$ 的第 j 行的树边点，边的权为 q_{jk}，称这种有向边为 Q 边。

(2)从 G_2 的树边点到对应的 G_1 的树边点加一条有向边；从 G_1 的补树边点到对应的 G_2 的补树边点加一条有向边，边的权为 1，称这种有向边为 C 边。

在 $G\hat{G}(t_i)$ 中，B 边、Q 边和 C 边分别用实线、虚线和点划线来表示。

定义 2：设 $g\hat{G}_{ii}(E_1,E_2)$ 是 $G\hat{G}(t_i)$ 的子图，E_1，E_2 是子图中的点集(对应于 G_1 和 G_2 的边的子集)，其中 $E_1\subseteq t_i$，$E_2\subseteq\bar{t}_i$，且 $|E_1|=|E_2|\neq 0$($|X|$ 表示集合 X 中元素的个数)。子图 $g_{ii}(E_1,E_2)$ 的生成回路集是 $g_{ii}(E_1,E_2)$ 中包含 $g_{ii}(E_1,E_2)$ 的所有点的点分离的有向回路的并集。

定义 3：设 E 是 $G_1(G_2)$ 的边集，$Ea=Eb(Eas/Ebs)$ 表示一个子集，它是指在 G_1，G_2 中，从边集 $Eb\subseteq E$ 中删去边集 Ebs，再增加边集 $Eas\subseteq E-Eb$ 而获得的。其中：$|Eas|=|Ebs|\neq 0$

公共树转换的充分必要条件：

设 $G\hat{G}(t_i)$ 是 G_1，G_2 关于任意一个公共树 t_i 的链路图，$g_{ii}(E_1,E_2)$ 是 $G\hat{G}(t_i)$ 的子图，其中：$E_1\subseteq t_i$，$E_2\subseteq\bar{t}_i$，且 $|E_1|=|E_2|\neq 0$。那么，当且仅当 $g\hat{G}_{ii}(E_1,E_2)$ 中存在奇数个生成回路集时，$t_j=t_i(E_2/E_1)$ 是 G_1，G_2 的一个新的公共树。

经公共树转换，由 t_i(符号为 ε_i)获得的新的公共树 t_j 的符号为：

$$\varepsilon_j=\varepsilon_i\times\sum(-1)^n\varepsilon_0 \tag{3}$$

其中：$\sum$ 表示对子图 $g\hat{G}_{ii}(E_1,E_2)$ 的所有生成回路集求和

n 是子图中每个生成回路集的重数

ε_0 是每个生成回路集中所有 B 边和 Q 边的权系数乘积

4 计算机算法及程序框图

以图 1 为例

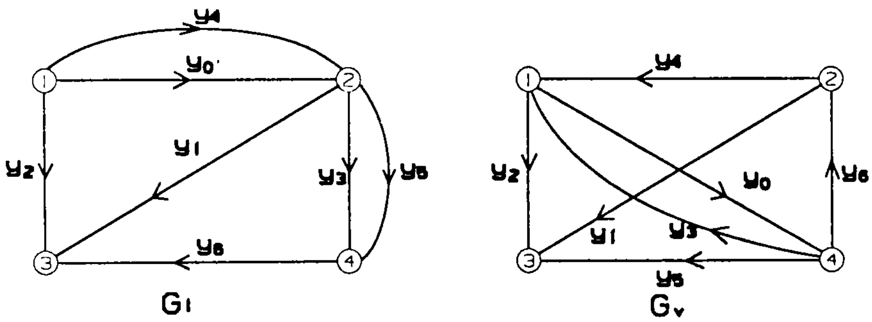

图 1 网络的电流图和电压图

G_i 和 G_V 两图的关联矩阵分别为：

$$A_i=\begin{array}{c}\begin{matrix} Y_0 & Y_1 & Y_2 & Y_3 & Y_4 & Y_5 & Y_6 \end{matrix}\\ \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 0\\ -1 & 1 & 0 & 1 & -1 & 1 & 0\\ 0 & -1 & -1 & 0 & 0 & 0 & -1 \end{bmatrix}\end{array}$$

$$A_V=\begin{array}{c}\begin{matrix} Y_0 & Y_1 & Y_2 & Y_3 & Y_4 & Y_5 & Y_6 \end{matrix}\\ \begin{bmatrix} 1 & 0 & 1 & -1 & -1 & 0 & 0\\ 0 & 1 & 0 & 0 & 1 & 0 & -1\\ 0 & -1 & -1 & 0 & 0 & -1 & 0 \end{bmatrix}\end{array}$$

$$A'_i=\begin{array}{c}\begin{matrix} Y_0 & Y_1 & Y_2 & Y_3 & Y_4 & Y_5 & Y_6 \end{matrix}\\ \left[\begin{array}{cccc|ccc} 1 & 0 & 1 & 0 & 1 & 0 & 0\\ 0 & 1 & 1 & 1 & 0 & 1 & 0\\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{array}\right]\end{array}$$

$$A'_V=\begin{array}{c}\begin{matrix} Y_0 & Y_1 & Y_2 & Y_3 & Y_4 & Y_5 & Y_6 \end{matrix}\\ \left[\begin{array}{cccc|ccc} -1 & 0 & -1 & 1 & 1 & 0 & 0\\ 0 & 1 & 1 & 0 & 0 & 1 & 0\\ -1 & -1 & -1 & 1 & 0 & 0 & 1 \end{array}\right]\end{array}$$

显然，Y_4, Y_5, Y_6 是 G_i 和 G_V 的一个公共树，设为 $t_i=\{Y_4, Y_5, Y_6\}$，其符号为：$\varepsilon_i=(-1)^{N_i-N_V}=(-1)^{1+4}=-1$

其中 N_i 和 N_V 分别是将 A_i 和 A_V 化为 A'_i 和 A'_V 所需的矩阵基本运算的次数。

为方便起见,将图中的边分别用数字表示,树枝 Y_4,Y_5,Y_6 分别用 1,2,3 表示,连枝 Y_0,Y_1,Y_2,Y_3 用 4,5,6,7 表示,作出 G_i 和 G_V 关于 t_i 的链路图 $G(t_i)$ 如图 2 所示。为了简化,已将 $G(t_i)$ 中所有的 C 边短路。

我们定义一个邻接矩阵 A,它反映了 $G\hat{G}(t_i)$ 中树边点和补树边点的邻接情况。如果 i 与 j 两点间只有一条从 i 到 j 的边,则 $Aij=+1$;如果只有一条从 j 到 i 的边,则 $Aij=-1$;若既有从 i 到 j 的边,又有从 j 到 i 的边,$Aij=2$;若两点间无边存在,$Aij=0$。由此得出 $G(t_i)$ 的邻接矩阵为:

$$A=\begin{array}{c}\\1\\2\\3\\4\\5\\6\\7\end{array}\begin{array}{c}\begin{array}{ccccccc}1&2&3&4&5&6&7\end{array}\\\left[\begin{array}{ccccccc}0&0&0&2&0&2&-1\\0&0&0&0&2&2&+1\\0&0&0&-1&2&2&-1\\2&0&+1&0&0&0&0\\0&2&2&0&0&0&0\\2&2&2&0&0&0&0\\+1&-1&+1&0&0&0&0\end{array}\right]\end{array}$$

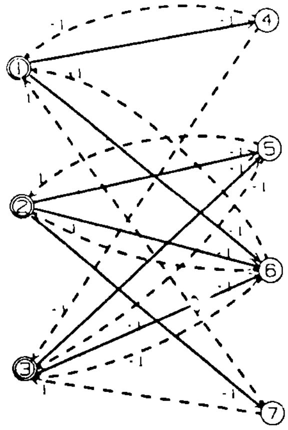

图 2　链路图

寻找 $G\hat{G}(t_i)$ 中所有可能的子图需要将树边点和补树边点分别进行组合,然后两者之间再进行组合。子图 $g_{ii}(E_1,E_2)$ 中的树边点和补树边点的个数是相同的(因为 $|E_1|=|E_2|\neq 0$)。若 $|E_1|=1$,称子图为一阶子图,若 $|E_1|=2$,称子图为二阶子图,以下类推。每个子图对应着 A 的一个子方阵,我们通过这个子方阵,采用双向搜寻的方法,逐层产生生成回路集拟树(S-拟树),从而求出每个子图的全部生成回路集。

假设子图 $g\hat{G}_{ii}(E_1,E_2)$ 中 $E_1=\{1,2,3\}$,$E_2=\{5,6,7\}$,对应的邻接矩阵为:

$$A'=\begin{array}{c}\\1\\2\\3\\5\\6\\7\end{array}\begin{array}{c}\begin{array}{cccccc}1&2&3&5&6&7\end{array}\\\left[\begin{array}{cccccc}0&0&0&0&2&-1\\0&0&0&2&2&+1\\0&0&0&2&2&-1\\0&2&2&0&0&0\\2&2&2&0&0&0\\+1&-1&+1&0&0&0\end{array}\right]\end{array}$$

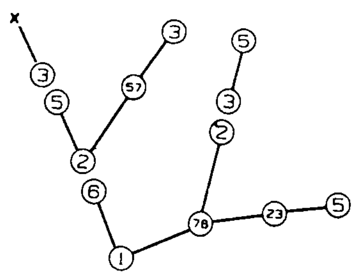

图 3　生成回路集拟树

选择一个树边点作为搜寻的出发点,即作为 S-拟树的树根,例如选点①为树根,划去 A' 中第①列(指点①对应的列,并不一定是第 1 列,以下同)。从 A' 的第①行可知,流入①的节点有⑥和⑦,流出①的节点为⑥,它们的组合为66和76,由此得 S-拟树的第一层为⑥和76。将66简写为⑥,表明流入和流出①的点均为⑥,点①和点⑥之间已形成一个小回路,然而此时回路尚未包含子图中所有的点,所以再从剩余的树边点中选择一个点例点②作为出发点,划去 A' 的第⑥、第②列。从②行知,流入②的点为⑤,流出②的点有⑤和⑦,组合为⑤和57,则又形成一层拟树,如图 3 所示。②与⑤之间又形成一个小回路,再选择点③为出发点,划去第⑤、第③列,从第③行可知,从③出发的点已经没有了,此路不通,停止搜寻,恢复③、⑤两列,这一层还有一个组合57,从⑤、⑦两行可知,进入⑤的点为③,

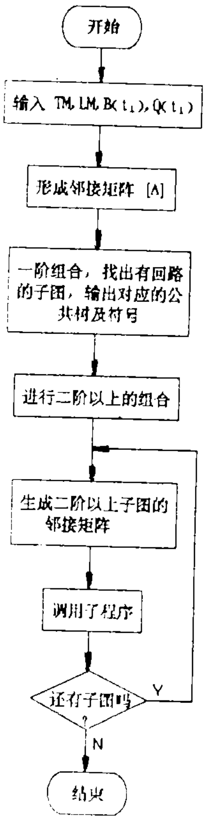

图4 寻找公共树主程序框图

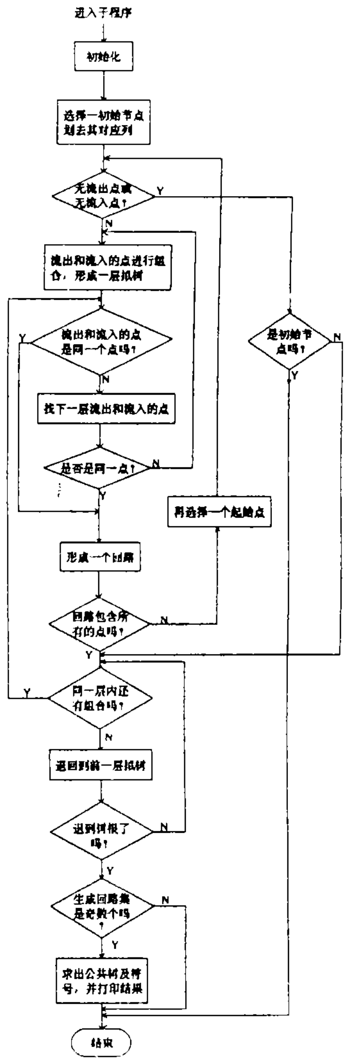

图5 寻找公共树子程序框图

流出⑦的点也为③，组合为③，又形成一个回路，此时回路已包含了子图的所有点，形成一个生成回路集：①—⑥—① ②—⑦—③—⑤—②。拟树的第二层还有一个组合⑦⑥，按上述方法继续搜寻，可得出另两个生成回路集为：

①—⑥—②—⑦—① ③—⑤—③

①—⑥—③—⑤—②—⑦—①

因为子图中有奇数个生成回路集，所以 $t_j=t_i(E_2/E_1)=\{5,6,7\}$ 是一个新的公共树，其符号 ε_j 可由式(3)求出：$\varepsilon_j=-1$。

对于二阶以上的子图，采用上述方法，找出每个子图的生成回路集。对含有奇数个生成回路集的子图进行公共树转换，并求出相应的符号。

对于一阶子图，由于只有一个树边点和一个补树边点，若有生成回路集，只能有一个，而且是一重的，它由一条 B 边和一条 Q 边构成。因此，从邻接矩阵 A 即可看出，若 A 中元素为 $A_{ij}=2$，则 i,j 两点构成的子图有一个一重的生成回路集。例 A 中元素 $A_{14}=2$，①，④两点构成的

子图有一个生成回路集:①—④—①。则4,2,3对应的边一定是一个公共树。

根据上述算法,给出计算机程序框图如图4、图5所示。主程序主要用来生成二阶以上的子图,子程序则找出每个子图的生成回路集,并实现公共树转换。

其中:TM,LM分别为树枝和连枝的总个数。

5 结束语

用完全树法求节点导纳矩阵的行列式,边Y_1,Y_2,……,Y_m若构成一个公共树,结果中就会出现一项$Y_1Y_2\cdots\cdots Y_m$。如果网络中的元件参数用不同的符号来表示,结果中就不存在也是由Y_1,Y_2,……,Y_m构成的其它公共树,所以结果中不会出现相消项。

本算法原理简单,整个算法基于数码访问和组合运算,易于用计算机实现。由于采用节点对的双向搜寻方法,每次分层都使邻接矩阵以二阶递减,收缩很快,节省了程序运行时间。本程序不仅可以求有源网络的电流图和电压图的全部公共树及符号,而且可用来求任意两个具有相同边集和相同秩的图的全部公共树,当两图完全相同时,相当于求出一个图的全部树。

参考文献

〔1〕Tadashi Matsumoto, Mikio Kitai, and Yoji Kajitani. On Distance Between The Common Trees in Two Graphs. proceedings of ISCAS 85

〔2〕A. C. Davies, Topological Solution of Networks Cantaining Nullators and Norators. Elect. Lett. 1966, 2 (3): 90—92

〔3〕M. N. S. Swamy, and K. Thulasiraman, Graphs, Networks, and Algorithms. Wiley Interscience, New York. 1981

〔4〕滕玉鹃,硕士学位论文. 武汉工学院,1991

A Kind of Computer Algorithm for Finding Common Trees

Teng Yujuan
(*Department of Automation, Jilin Institute of Technology, Changchun* 130012)
Le Quangen
(*Department of Electronics, Wuhan Institute of Technology, Wuhan* 430070)

Abstract

The method of common tree transformation is used to find out all the common trees of two graphs with the same edge set and the same rank. The associated signs of common trees can be obtained at the same time. The flow-chart for realizing the algorithm is given.

Key words: Common tree, Chain path graph, Spanning loop set