Graphs and Their Applications (6)

K.M. Koh*

Department of Mathematics

National University of Singapore, Singapore 117543

F.M. Dong and E.G. Tay

Mathematics and Mathematics Education

National Institute of Education

Nanyang Technological University, Singapore 637616

17 The Number of Spanning Trees

Let G be a connected multigraph of order n. A spanning tree of G is a spanning subgraph of G and is itself a tree. Thus, every spanning tree of G is of size $n-1$. The concept of spanning trees was introduced in [1], and we also discussed therein its applications to the unfolding of 3-D folded structures and the one-way street problem.

Given a connected multigraph G, we know (Theorem 15.1 in [1]) that G contains a spanning tree. Certainly, we are not just content with this answer. You may wish to ask: how many spanning trees could G contain? We shall study this problem in this article.

In what follows, unless otherwise stated, G is a connected multigraph of order n. For convenience, we shall denote by $\tau(G)$ our key quantity, namely, the number of different spanning trees of G. However, before we proceed, we need to clarify what it means by saying that two spanning trees of G are different. Let T_1 and T_2 be two spanning trees of G. We say that T_1 is *different from* T_2 if there exists an edge of G which appears in T_1 but not in T_2 (or vice versa). For instance, of the multigraph G of Figure 17.1,

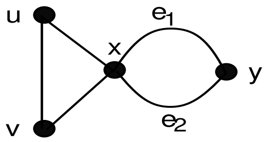
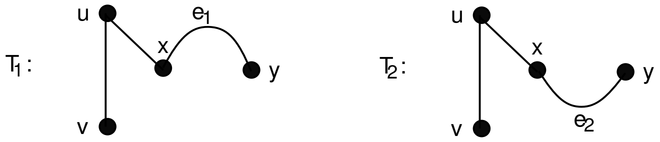


Figure 17.1

*Corresponding author. Email: matkohkm@nus.edu.sg.

the following two spanning trees T_1 and T_2, though isomorphic, are regarded as different, since the edge e_1 of G appears in T_1 but not in T_2.



Indeed, there are exactly six different spanning trees of G (besides T_1 and T_2, the other four are shown in Figure 17.2). Thus, $\tau(G) = 6$.

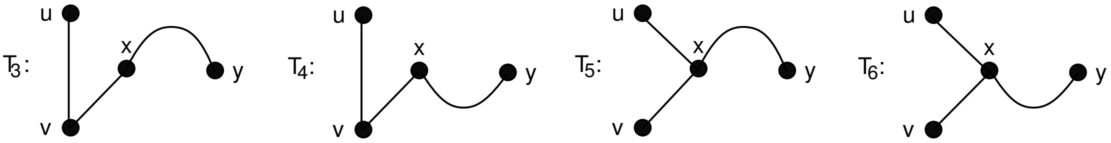


Figure 17.2

To end this section, let us give some observations and examples.

(1) If H is a disconnected multigraph, then $\tau(H) = 0$. Thus, for any multigraph H, $\tau(H) \geq 1$ if and only if H is connected.

(2) If H is a tree, then H itself is its only spanning tree, and so $\tau(H) = 1$. Indeed, let H be a multigraph. Then $\tau(H) = 1$ if and only if H is a tree (see Exercise 17.1).

(3) For the cycle C_n of order n, where $n \geq 2$, $\tau(C_n) = n$ (see Exercise 17.2).

(4) Let H_1 and H_2 be two multigraphs with two specified vertices w_1 and w_2 respectively. Denote by $H_1 \bullet H_2$ the multigraph obtained from H_1 and H_2 by identifying w_1 and w_2 as shown in Figure 17.3. Then (see Exercise 17.4)

$$\tau(H_1 \bullet H_2) = \tau(H_1)\tau(H_2).$$

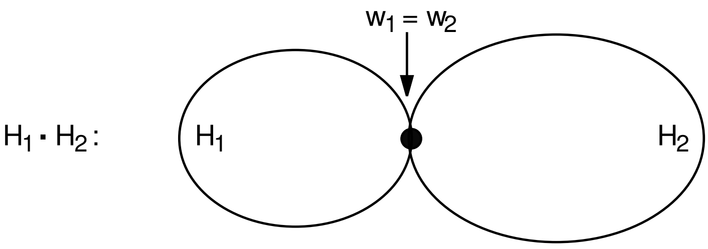
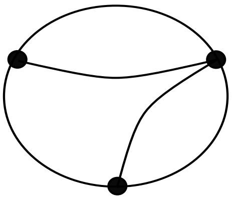


Figure 17.3

Exercise 17.1 *Let H be a multigraph. Show that $\tau(H) = 1$ if and only if H is a tree.*

Exercise 17.2 *Show that $\tau(C_n) = n$ for each $n \geq 2$.*

Exercise 17.3 *Consider the following multigraph G:*



Evaluate $\tau(G)$ by listing all the different spanning trees of G.

Exercise 17.4 *Let H_1 and H_2 be two multigraphs. Show that*

$$\tau(H_1 \bullet H_2) = \tau(H_1)\tau(H_2),$$

where $H_1 \bullet H_2$ is the multigraph defined in (4) above.

Exercise 17.5 *Let G be a connected multigraph. Show that*

(i) if H is a spanning submultigraph of G, then $\tau(H) \leq \tau(G)$; and more generally,

(ii) if H is a submultigraph of G, then $\tau(H) \leq \tau(G)$.

18 A Recursive Formula

Given a multigarph G, one can imagine that it is by no means a simple task to compute $\tau(G)$ if the order or size of G is reasonably big. Is there any systematic way which enables us to enumerate $\tau(G)$ at least step by step? In what follows, we shall introduce a method, which is recursive in nature, to compute $\tau(G)$.

Before we proceed, we need to introduce a way of forming a new multigraph from the given G with a specific edge. Thus suppose that $e = uv$ is a specific edge in G. Let us denote by $G \bullet e$ the multigraph obtained from G by first deleting all edges joining u and v, and then identifying u and v. We note that the order of $G \bullet e$ is one less than that of G and the size of $G \bullet e$ is always less than that of G. An example is shown in Figure 18.1.

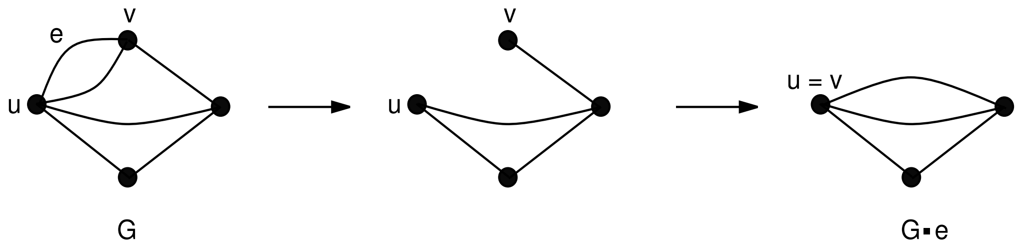
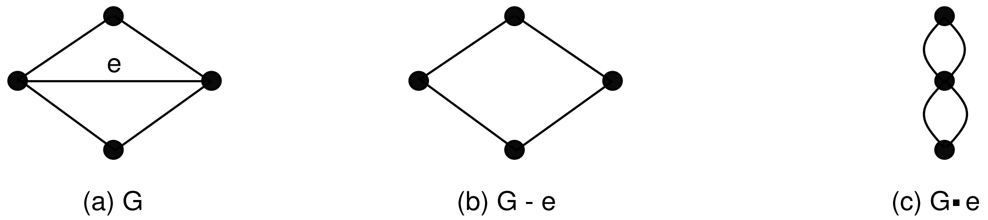


Figure 18.1

Recall that $G - e$ is the multigraph obtained by deleting e from G. Thus, if G is the graph with a specific edge e as shown in (a) of Figure 18.2, then $G - e$ and $G \bullet e$ are shown in (b) and (c) of the figure respectively.



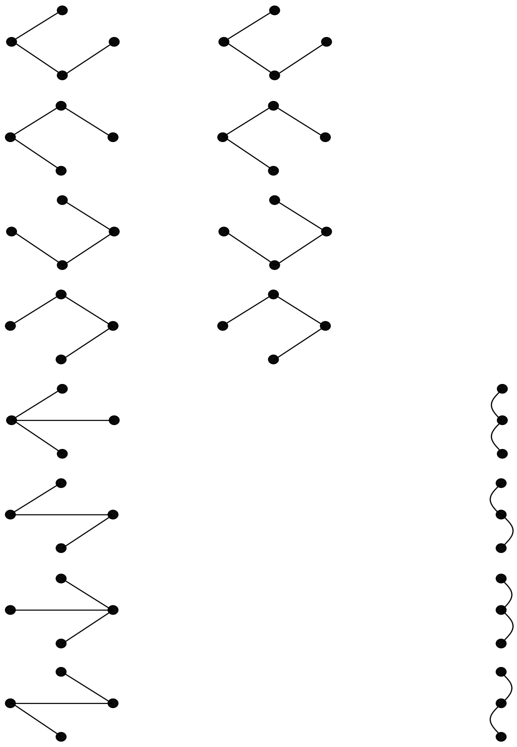
(a) G

(b) G - e

(c) G▪e

Figure 18.2

Let us list all the spanning trees of G, $G - e$ and $G \bullet e$ respectively in the three columns of the following table:



We note that

(1) $\tau(G) = \tau(G - e) + \tau(G \bullet e)$;

(2) the four spanning trees of $G - e$ are same as the first four spanning trees of G;

(3) there is a one-one correspondence between the four spanning trees of $G \bullet e$ and the last four spanning trees of G (see Exercise 18.1).

With this in mind, we are now in a position to establish the following result:

Theorem 18.1 *Let G be any multigraph and e an edge in G. Then*

$$\tau(G) = \tau(G - e) + \tau(G \bullet e).$$

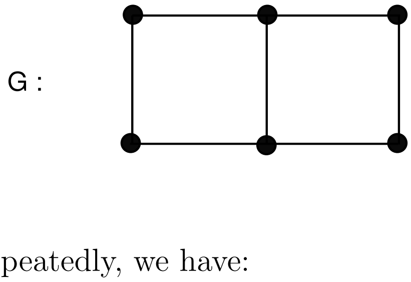
Proof. Let A be the set of spanning trees of G that contain the given edge e and let B be the set of spanning trees of G that do not contain e. Then there is a one-one correspondence between A and the set of spanning trees of $G \bullet e$ (see Exercise 18.1), and, evidently, B is the same as the set of spanning trees of $G - e$. Thus,

$$\tau(G \bullet e) = |A| \quad \text{and} \quad \tau(G - e) = |B|,$$

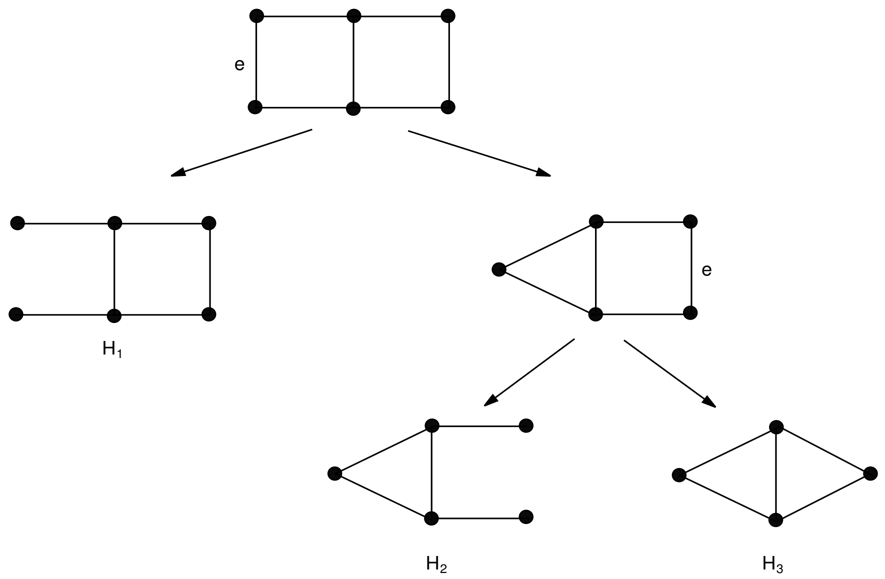
and so $\tau(G) = |A| + |B| = \tau(G \bullet e) + \tau(G - e)$, as was to be shown. □

Given G, both $G - e$ and $G \bullet e$ are either of smaller order or of smaller size, as compared to G. Thus, by applying Theorem 18.1 successively, the enumeration of $\tau(G)$ is reduced to the computations of $\tau(H)$'s, where H's are multigraphs of much smaller order or size, which are certainly much easier. An illustration is given below.

Example 18.1 *Consider the following graph.*



By applying Theorem 18.1 repeatedly, we have:

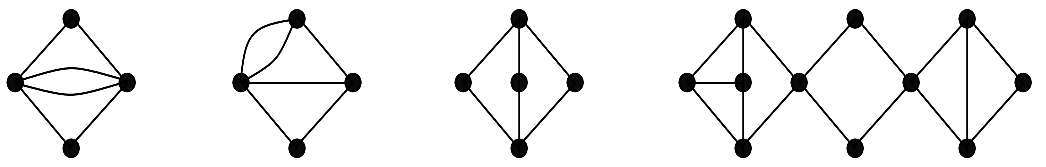


Note that $\tau(H_1) = 4$ and $\tau(H_2) = 3$ (see Exercises 17.2 and 17.3) while $\tau(H_3) = 8$ as shown at the beginning of this section. We thus conclude that $\tau(G) = 4 + 3 + 8 = 15$.

Remark: The value of $\tau(G)$ obtained by applying Theorem 18.1 is independent of the choice of the sequence of edges.

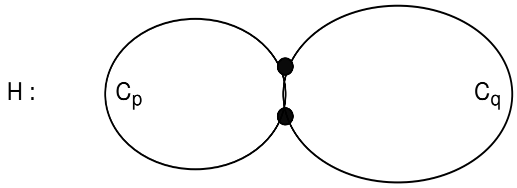
Exercise 18.1 *Let G be a connected multigraph and e an edge in G. Establish a one-one correspondence between the set of spanning trees of G that contain e and the set of spanning trees of $G \bullet e$.*

Exercise 18.2 *Evaluate $\tau(G)$ for each of the following multigraphs:*



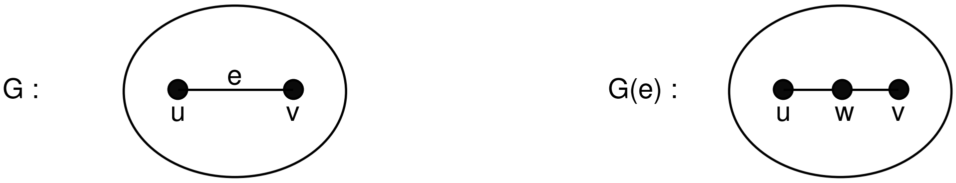
Exercise 18.3 *Let G be an (n, n)-multigraph, where $n \geq 3$. Which G has the largest $\tau(G)$?*

Exercise 18.4 *Consider the following graph H obtained from two cycles C_p and C_q by identifying an edge in C_p with an edge in C_q, where $p, q \geq 3$. Evaluate $\tau(H)$ in terms of p and q.*



Exercise 18.5 *Let G be an $(n, n+1)$-multigraph, where $n \geq 3$. Which G has the largest $\tau(G)$?*

Exercise 18.6 *Let G be a multigraph and e an edge in G. Denote by $G(e)$ the multigraph obtained from G by inserting a new vertex of degree 2 into e as shown below:*



Study the relation between $\tau(G(e))$ and $\tau(G)$.

19 Kirchhoff's Matrix-Tree Theorem

In this section, we shall introduce another way for the enumeration of $\tau(G)$, which makes use of the notion 'matrix'. In what follows, we assume that the readers are familiar with some basic concepts pertaining to it.

Every multigraph can be represented uniquely by a matrix, called its adjacency matrix. In general, let G be a multigraph with vertex set $V(G) = \{v_1, v_2, \cdots, v_n\}$. The *adjacency matrix* of G is the $n \times n$ matrix $A(G) = (a_{i,j})$ in which '$a_{i,j}$' is the number of edges joining the vertices v_i and v_j, where $i, j = 1, 2, \cdots, n$. For instance, if G is the multigraph of Figure 19.1,

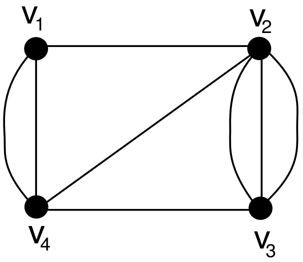


Figure 19.1

then we have $a_{1,1} = a_{2,2} = a_{3,3} = a_{4,4} = 0$, $a_{1,2} = a_{2,1} = 1$, $a_{1,3} = a_{3,1} = 0$, $a_{1,4} = a_{4,1} = 2$, etc and so

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

Notice that the matrix $A(G)$ is symmetric and all the diagonal entries in $A(G)$ are zero.

Let us define another matrix associated with G, called its degree matrix. In general, the *degree matrix* of G is the matrix $D(G) = (d_{i,j})_{n\times n}$, where

$$d_{i,j} = \begin{cases} \text{the degree of } v_i, & \text{if } j = i, \\ 0, & \text{otherwise.} \end{cases}$$

Thus, $D(G)$ is a diagonal matrix in which the diagonal entries are the degrees of the corresponding vertices in G. For instance, for the multigraph G of Figure 19.1,

$$D(G) = \begin{pmatrix} 3 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix}.$$

Observe that

$$D(G) - A(G) = \begin{pmatrix} 3 & -1 & 0 & -2 \\ -1 & 5 & -3 & -1 \\ 0 & -3 & 4 & -1 \\ -2 & -1 & -1 & 4 \end{pmatrix}.$$

We shall see that this resulting matrix $D(G) - A(G)$ plays a prominent role in evaluating $\tau(G)$.

Let M be an $n \times n$ matrix. For $i, j = 1, 2, \cdots, n$, the *cofactor* of the (i, j)-entry in M is defined as $(-1)^{i+j}$ times the determinant of the $(n-1) \times (n-1)$ matrix obtained from M by deleting the i-th row and j-th column in M.

For instance, if M is the 4×4 matrix $D(G) - A(G)$ shown above, then

$$\text{the cofactor of the } (1,1)\text{-entry in } M \text{ is given by} \quad (-1)^{1+1} \begin{vmatrix} 5 & -3 & -1 \\ -3 & 4 & -1 \\ -1 & -1 & 4 \end{vmatrix} = 29$$

and

$$\text{the cofactor of the } (3,2)\text{-entry in } M \text{ is given by} \quad (-1)^{3+2} \begin{vmatrix} 3 & 0 & -2 \\ -1 & -3 & -1 \\ -2 & -1 & 4 \end{vmatrix} = 29.$$

We are now ready to state without proof the following beautiful and surprising result, known as the Matrix-Tree theorem, which is implicit in the work of Gustav Kirchhoff (1824 - 1887).

Theorem 19.1 *For any multigraph G, $\tau(G)$ is equal to the cofactor of any entry in $D(G) - A(G)$.*

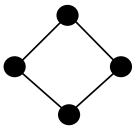
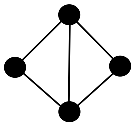
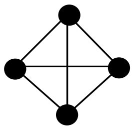
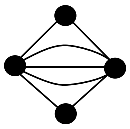
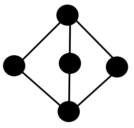
Remarks.

(1) In general, given a square matrix M, the cofactors of different entries in M need not be the same. However, due to the special feature of $D(G) - A(G)$ (note that the sum of the entries in each row or column is zero), the cofactors of any two different entries in $D(G) - A(G)$ are the same. Thus, as highlighted in Theorem 19.1, the value of $\tau(G)$ is independent of the choice of the entry in $D(G) - A(G)$. (See the example preceding Theorem 19.1. Note that $\tau(G) = 29$ in this case.)

(2) As far as computing $\tau(G)$ is concerned, it seems that Theorem 18.1 is more user friendly than Theorem 19.1. In fact, if the order and size of G are large, Theorem 18.1 is impractical. On the other hand, there are efficient ways available to compute the determinant of a square matrix. Thus, one can compute $\tau(G)$ efficiently by Theorem 19.1.

(3) The reader may refer to [2] for the proof of Theorem 19.1 and its generalizations.

Exercise 19.1 *For each of the following multigraphs G, compute $\tau(G)$ by Theorem 19.1.*

References

[1] K. M. Koh, F. M. Dong and E. G. Tay, Graphs and their applications (5), *Mathematical Medley* **31**(2) (2004) 77-87.

[2] D. B. West, *Introduction to Graph Theory*, Prentice-Hall (2001).