On the Parity of Graph Spanning Tree Numbers

David Eppstein*

Department of Information and Computer Science
University of California, Irvine, CA 92717
http://www.ics.uci.edu/∼eppstein/

Tech. Report 96-14

April 30, 1996

Abstract

Any bipartite Eulerian graph, any Eulerian graph with evenly many vertices, and any bipartite graph with evenly many vertices and edges, has an even number of spanning trees. More generally, a graph has evenly many spanning trees if and only if it has an Eulerian edge cut.

*Work supported in part by NSF grant CCR-9258355 and by matching funds from Xerox Corp.

1 Introduction

Smith's Theorem states that every trivalent graph has an even number of Hamiltonian cycles; this is proven by a correspondence between each of these paths with the endpoints of another system of paths, so that each Hamiltonian path has a partner that can be found by following a path from this system. This correspondence leads to interesting questions in computational complexity [6, 7]. Similar problems on the numbers of partitions of a graph into a spanning tree and a matching or Hamiltonian cycle were considered by Chrobak and Poljak [4].

In this paper we investigate a similar problem, the parity of the number of spanning trees of a graph. This number can be calculated as a certain matrix determinant via the well-known *matrix-tree theorem* (e.g. see [3]), and algebraic characterizations of the parity of this number were known [1], but it is not obvious from these results how the number of trees relates to structural properties of the graph. We show that a graph has an even number of spanning trees, if and only if it has an *Eulerian edge cut*, that is, a subgraph $G \cap ((V(G) - S) \times S)$, where S is some vertex subset of G, with the property that all degrees in the subgraph are even. This equivalence between a problem formulation naturally belonging to $\oplus P$ and a formulation naturally belonging to NP is intriguing, as it would be surprising indeed if these two complexity classes were equal. However because of the matrix-tree theorem, the problem represented by the two formulations turns out to be solvable in polynomial time.

We extend one direction of our characterization, to deal with other types of divisibility, and in the other direction we identify several classes of graphs that satisfy our conditions. In particular, every bipartite Eulerian graph has evenly many spanning trees.

The proofs of these results are based on a fascinating new construction by Bacher, de la Harpe, and Nagnibeda [1] of a finite Abelian group, the elements of which are certain equivalence classes of fractional flows, having the same number of elements as the number of spanning trees in the graph. This technique seems to be considerably less constructive than that of Smith's theorem, and in particular does not seem to lead to a natural matching of pairs of spanning trees. However we can use this construction to find an Eulerian edge cut when one exists, in polynomial time. (The existence of an Eulerian edge cut can be tested by the matrix-tree theorem but it is not clear how to use that information to find one.)

2 Elementary Methods

We begin by some explorations of the number of spanning trees that do not require much in the way of algebra. We let m and n denote the number of edges and vertices respectively of a given graph G.

Theorem 1. *Let G be Eulerian, with n even. Then G has evenly many spanning trees.*

Proof: We define a graph G', the vertices of which are the spanning trees of G, with two vertices connected by an edge if the corresponding trees differ by a *swap* (the deletion of a single edge and its replacement by another edge). Deleting an edge from a tree partitions the tree into two components, and the possible replacements are those spanning the gap between the two components. Any tour in G must cross this gap an even number of times. In particular this is true of any particular Euler tour, which of course includes the deleted edge itself; the remaining crossings of this tour are all the possible replacements for the edge. So any edge in any tree is involved in an odd number of swaps. Since n is even, any tree has an odd number of edges and thus the corresponding vertex in G' has odd total degree. But the number of odd-degree vertices in any graph is even, so G' has evenly many vertices and G has evenly many spanning trees. □

This can be dualized in a certain sense to give a similar result:

Theorem 2. *Let G be bipartite, with $m + n$ even. Then G has evenly many spanning trees.*

Proof: We define the same auxiliary graph G'. For any tree T in G, any edge not in T induces a unique cycle with the edges of T which (by bipartiteness) has even length. The swaps involving a given edge are exactly those in which an edge in this cycle is deleted, so any edge not in T is involved in an odd number of swaps. By the assumption that $m + n$ is even, there is an odd number of edges not in any tree, so the total degree of each vertex in G' is odd. The proof follows as before. □

3 The Jacobian of a Graph

According to Dieudonné, "it has been said that when you do not quite understand the properties of new mathematical objects, you should try to put a group structure on them. This seems like a whim, but in fact it has more than once succeeded." [5]

The objects I am interested in are spanning trees; however it is far from obvious that there is any natural way of giving them a group structure. We now describe the method of Bacher et al. [1], which at least finds a group having the same order as the number of spanning trees in the graph. With this construction, we may not be able to study individual spanning trees, but we can get a handle on the total number of them.

Given an undirected graph G, we form a directed graph by orienting its edges arbitrarily. A *preflow* on G is then an assignment of real-valued flow values to each oriented edge; a *flow* is a preflow with the property that the total incoming flow equals the total outgoing flow at each vertex. Bacher, de la Harpe, and Nagnibeda [1] interpret the set of all preflows as a vector space, with a dot product of two preflows formed by multiplying the two flow values on each edge and summing the results. The set of all flows is then a linear subspace of this vector space, and the set of all *integer-valued* flows forms a *lattice* (discrete full-dimensional subgroup) $\Lambda(G)$ in this subspace. The *dual lattice* $\Lambda^{\#}(G)$ consists of the fractional flows having the property that their dot products with all integer flows are integers. For instance, in a cycle C_k, any integer flow sends some integer i units of flow around the cycle, and the dot product of this flow with a fractional flow sending j/k flow units around the cycle would be the integer ij. So these fractional flows are members of $\Lambda^{\#}(G)$. It is clear that none of these spaces depends on the orientation of G (actually, following Serre, Bacher et al. use the directed graph with both orientations of each edge, and require that the flows in each direction are negations of each other).

Bacher, de la Harpe, and Nagnibeda then form a finite Abelian group, the *Jacobian* of G, by taking the quotient of these two lattices. Miraculously, the order of this group (the number of its elements) is the same as the number of spanning trees of G. (Both quantities can be computed as a certain determinant of a matrix derived from G; the tree side of this equality is the matrix-tree theorem.)

Theorem 3 (Bacher, de la Harpe, and Nagnibeda). *For any graph G, $\Lambda(G)$ is a normal subgroup of $\Lambda^{\#}(G)$, and the quotient $\mathcal{J}(G) = \Lambda^{\#}(G)/\Lambda(G)$ is a finite Abelian group with order $\kappa(G)$ equal to the number of spanning trees of G.*

We will use this result to test the divisibility properties of $\kappa(G)$ by finding elements of the appropriate orders in $\mathcal{J}(G)$. From now on, we will not refer to spanning trees at all, except via the quantity $\kappa(G)$.

4 Divisibility from Cuts

We now prove that information from the cuts of a graph can be used to find divisors of $\kappa(G)$. The method is to construct a preflow from the cut that, if it were a flow, would correspond to an element of the appropriate order in $\mathcal{J}(G)$. By adjusting the preflow by integer amounts on the edges of a spanning tree of G, we get a flow with the same behavior and hence a divisor of $\kappa(G)$. Note that Bacher et al. define a similar process for transforming preflows into flows, that is more canonical in that it does not require us to specify a spanning tree (it is just the orthogonal projection in the corresponding vector spaces). However their process does not preserve the fractional values of the preflow and is unsuitable for our purposes.

Theorem 4. *Let C be an edge cut subgraph of a graph G. Then the gcd of vertex degrees in C divides $\kappa(G)$.*

Proof: Let U and V be the vertex sets on each side of the cut, and let g denote the gcd of vertex degrees in C. Form a preflow by assigning each edge in C a flow value of $1/g$ in the direction from U to V. This is not a flow, but the *excess* (difference between incoming and outgoing flow) at each vertex is an integer. By adjusting the preflow by integer amounts on the edges of a spanning tree of G, one can form a flow F in which each edge in C is given a value $1/g$ plus an integer, and each remaining edge is given an integer value.

We now show that F is in $\Lambda^{\#}(G)$. In other words, the dot product of F with any integer flow is an integer. Since any integer flow can be reduced to a sum of cycles, we need only prove that the dot product of F with any cycle is zero. But any cycle must cross from U to V exactly the same number of times as it crosses from V to U, so the $1/g$ fractions of flow in F cancel in the dot product.

Thus F corresponds to an element in $\mathcal{J}(G)$, which clearly has order g since g times F is an integer flow (in $\Lambda(G)$) while any smaller multuple of F has non-integer flow values on the edges of C. $\square$

We can also deduce divisibility from cycles in Eulerian graphs, as a partial dual to the above theorem. (More precisely, it is dual to the case of Theorem 4 in which $C = G$ and so G is bipartite.)

Theorem 5. *Let G be Eulerian. Then the gcd of cycle lengths in G divides $\kappa(G)$.*

Proof: Let g denote the gcd of cycle lengths in G. We send $1/g$ units of flow around an Eulerian cycle of G. This flow has integer dot product with any cycle, and hence with any integer flow in G, so it is an element of $\Lambda^{\#}(G)$ It clearly corresponds to an element of order g in $\mathcal{J}(G)$. $\square$

The following is a special case of both of these theorems.

Corollary 1. *Any bipartite Eulerian graph has an even number of spanning trees.*

In this connection, it is curious to note that bipartiteness and Eulerianness are planar duals of each other, so the property of being bipartite Eulerian is self-dual. Also, a graph is bipartite if and only if every edge is in an odd number of bonds, and Eulerian if and only if every edge is in an odd number of cycles [8].

5 Cuts from Parity

Theorem 4 tells us that a cut with certain properties gives rise to a divisor of $\kappa(G)$. The reverse is not true in general; for instance the triangle (for which $\kappa(G) = 3$) has no cut with vertex degree gcd more than one. However we can find a converse for the case when this gcd is two. In this case, we are looking for *Eulerian cuts*, that is, cuts that form subgraphs in which all vertex degrees are even. (We do not constrain the cut edges to form a connected subgraph, so the cut may not actually have an Euler tour.)

Theorem 6. *Graph G has an Eulerian edge cut if and only if $\kappa(G)$ is even.*

Proof: In one direction, Theorem 4 tells us that any G with an Eulerian cut has $\kappa(G)$ even. In the other direction, if $\kappa(G) = |\mathcal{J}(G)|$ is even, $J(G)$ must have an element e of order two. Then clearly, e corresponds to a family of fractional flows in which all flow values are integers or half-integers, and some nonempty set of flow values are half-integers. Let S be the edges with half-integer flow values in some flow e' corresponding to e. Then S must have even vertex degrees in order to satisfly the flow constraints. We now show that S is an edge cut in G.

Choose some vertex $v \in G$, and form two vertex sets: set A consists of those vertices that can be connected to v by a path using an odd number of edges in S, and set B consists of those vertices that can be connected to v by a path using an even number of edges in S. We can assume without loss of generality that G is connected, so A and B together cover G. Further, no vertex can be in both sets, for by combining the corresponding two paths we could get a cycle (not necessarily simple) in G involving an odd number of edges of S. Sending one unit of flow around this cycle would give a flow having a half-integer dot product with e', contradicting the assumption that e' is in $\Lambda^{\#}(G)$. Thus A and B are disjoint. Each edge in S is itself a path (with an odd number of edges in S) and so connects one vertex in A with one in B). Further, each other edge in G is a path with evenly many edges in S and so does not cross from A to B. Thus S must be the cut consisting of the edges connecting A and B. $\Box$

From this we can deduce the presence of Eulerian cuts in graphs for which we know the number of spanning trees to be even, even though there may be no obvious construction of an explicit Eulerian cut.

Corollary 2. *Any Eulerian graph with n even, and any bipartite graph with $m + n$ even, has an Eulerian cut.*

6 Finding Eulerian cuts

Because of Theorem 6, we can test whether a graph has an Eulerian edge cut in polynomial time, by using the matrix-tree theorem to count the spanning trees in the graph. However it is not clear how to use this information to find such a cut. Instead we can find an Eulerian cut more directly, by interpreting the constraints defining the (fractional part of) the corresponding fractional flow as being a system of linear equations over the finite field $GF(2)$.

Theorem 7. *We can find an Eulerian edge cut in any graph that has one, in polynomial time.*

Proof: We define a system of variables and equations in $GF(2)$, one variable per edge of G. We create two classes of equations, one enforcing the condition that the total parity of edge variables at each vertex of G is even. The other class of equations enforces, for each cycle in a cycle basis of G, the condition that the parity of edge variables for each edge of the cycle is even. There are $m + 1$ equations in m unknowns, but we can ignore one of the vertex parity equations since it is a consequence of the $n - 1$ others. The result is just a system of m linear equations in m unknowns, which we can solve (if a solution exists) in polynomial time.

Then if G has an Eulerian edge cut, this system of equations has a solution found by setting the variables corresponding to cut edges to one and the other variables to zero. Conversely, if the equations have a solution, one can find a preflow with flow values 1/2 on edges corresponding to the one values of the solution and integer values elsewhere; the vertex parity constraints cause the excess at each vertex to be an integer, so the technique used in Theorem 4 of adjusting by integer flow amounts on spanning tree edges can be used to find a flow f with half-integer flow amounts in the same places. Then since each integer flow in G is a sum of cycles in the cycle basis, the cycle basis constraints cause f to be in $\Lambda^{\#}(G)$, and the proof of Theorem 6 shows that the half-integer-flow edges in f form an Eulerian edge cut. □

References

[1] R. Bacher, P. de la Harpe, and T. Nagnibeda. The lattice of integral flows and the lattice of integral coboundaries on a finite graph. Preliminary version, 19 July 1995. Available on the Amer. Math. Soc. preprint server at URL http://www.ams.org/preprints/05/199509/199509-05-002/199509-05-002.html

[2] K. A. Berman. Bicycles and spanning trees. *SIAM J. Alg. Disc. Meth.* 7 (1986).

[3] J. A. Bondy and U. S. R. Murty. Graph Theory with Applications. North Holland, 1976, pp. 218–219.

[4] M. Chrobak and S. Poljak. On common edges in optimal solutions to travelling salesman and other optimization problems. Discrete Applied Math. 20 (1988) 101–111.

[5] J. Dieudonné. Mathematics—the Music of Reason. Springer, 1992, p. 154.

[6] Adam Krawczyk. A note on the complexity of finding a second Hamiltonian cycle in cubic graphs. Manuscript, 1995.

[7] C. H. Papadimitriou. On graph-theoretic lemmata and complexity classes. 31st IEEE Symp. Foundations of Computer Science (1990) 794–801.

[8] D. R. Woodall. A proof of McKee's Eulerian-bipartite characterization. Discrete Math. 84 (1990) 217–220.