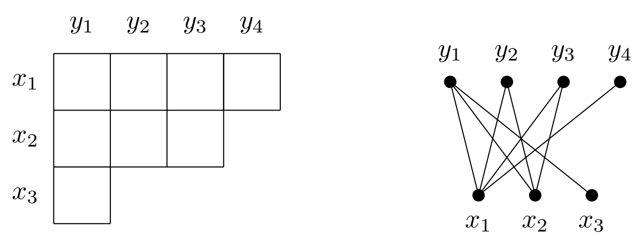
UPPER BOUNDS ON THE NUMBER OF SPANNING TREES IN A BIPARTITE GRAPH

FINTAN GARRETT AND STEVEN KLEE

ABSTRACT. Ferrers graphs are a family of connected bipartite graphs that arise from the Ferrers diagram of a partition. Ehrenborg and van Willigenburg gave a beautiful product formula for the number of spanning trees in a Ferrers graph. In this paper, we use linear algebraic techniques to investigate a conjecture of Ehrenborg stating that a similar product formula gives an upper bound for the number of spanning trees in an arbitrary bipartite graph.

1. Introduction. Suppose we are given an $m \times n$ array of boxes. A subset, $\mathcal{F}$, of these boxes is called a **Ferrers diagram** if, whenever box (z, w) belongs to $\mathcal{F}$, then each box (x, y) with $x \leq z$ and $y \leq w$ also belongs to $\mathcal{F}$. Here, we use matrix notation, so that box $(1, 1)$ is in the upper left hand corner of the array.

A Ferrers diagram gives rise to a Ferrers graph whose vertices are indexed by the rows $x_1, x_2, \ldots, x_n$ and columns $y_1, y_2, \ldots, y_m$ of the array; and the edge $\{x_i, y_j\}$ belongs to the graph if and only if the box in row x_i and column y_j belongs to the Ferrers diagram. For example, the following figure illustrates a Ferrers diagram (left) and its corresponding Ferrers graph (right).



Definition 1.1. *A **Ferrers graph** is a bipartite graph G whose vertices are partitioned as $X \cup Y$ with $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_m\}$ with the property that*

(1) *whenever $\{x_k, y_\ell\}$ is an edge in G, then so is $\{x_i, y_j\}$ for any $i \leq k$ and $j \leq \ell$, and*
(2) *$\{x_1, y_m\}$ and $\{x_n, y_1\}$ are edges in G.*

Ehrenborg and van Willigenburg [**1**] found a beautiful formula counting the number of spanning trees in a Ferrers graph. Here and throughout, we will use $\tau(G)$ to denote the number of spanning trees in a graph G.

Theorem 1.2. [**1,** Theorem 2.1] *Let G be a Ferrers graph whose vertices are partitioned as $V(G) = X \cup Y$. Then*

$$|X| \cdot |Y| \cdot \tau(G) = \prod_{v \in V(G)} \deg(v).$$

Received by the editors December 9, 2015.

This has led Ehrenborg (personal communication) to conjecture the following upper bound formula for the number of spanning trees in an arbitrary bipartite graph.

Conjecture 1.3. *Let G be a simple bipartite graph whose vertex set is partitioned as $V(G) = X \cup Y$. Then*

$$|X| \cdot |Y| \cdot \tau(G) \leq \prod_{v \in V(G)} \deg(v).$$

We begin by motivating Conjecture 1.3 as part of the existing literature. Shifted simplicial complexes play an important role in the study of extremal combinatorics of set systems. Specifically, for a simplicial complex with a fixed number of k-dimensional faces, a shifted simplicial complex supports the maximal possible number of $(k+1)$-dimensional faces as prescribed by the Kruskal-Katona Theorem **[2, 4]**. Further, a Ferrer's graph is a shifted version of a bipartite graph (which is often called a "compressed" bipartite graph). For any simple graph there is a corresponding graphic matroid, which is a simplicial complex whose vertices correspond to the edges of the graph and whose facets correspond to the spanning trees of the graph. Therefore, in a sense, Conjecture 1.3 makes a further claim that the graphic matroids corresponding to shifted bipartite graphs are extremal.

The purpose of this paper is twofold. First, we use linear algebraic techniques reformulate Conjecture 1.3 in terms of purely combinatorial data arising from a given bipartite graph. For any bipartite graph G, we find an associated edge-weighted complete graph $A(G)$ whose weighted spanning tree enumerator counts the number of spanning trees in the original graph G. This allows us to computationally verify Conjecture 1.3 when $|X| \leq 5$. Next, we use this machinery and linear algebraic techniques to give a new proof of Theorem 1.2 for Ferrers graphs. This combinatorial re-formulation seems to give a promising approach to studying Conjecture 1.3 in general.

The remainder of the paper is structured as follows. In Section 2 we review the main linear algebraic tools that will be relevant to our proofs and use these tools to derive the associated edge-weighted complete graph $A(G)$ that arises from a bipartite graph G. In Section 3 we show that Conjecture 1.3 holds when $|X| \leq 5$. Finally, in Section 4 we further study the Laplacian matrix of the graph $A(G)$ to give a new proof of Theorem 1.2.

2. Preliminaries. For the purposes of this paper, we will assume that all graphs are simple, meaning they do not contain loops or multiple edges. We will use $V(G)$ and $E(G)$ to denote the vertex set and edge set of G respectively. We will always assume that $V(G)$ is finite. We will use $\tau(G)$ to denote the number of spanning trees in G. If v is a vertex of G, we will use $\deg(v)$ to denote the degree of v. When it is necessary, we will use the notation $\deg_G(v)$ to emphasize that we are counting the degree of v in G.

Furthermore, the use of subscripts and parentheses can become confusing when describing various operations on graphs and matrices. We will consistently use the following notation throughout the paper in an effort to prevent such confusion.

- If G is a graph, we will use curly braces $\{i, j\}$ to denote an edge in G.
- If G is a graph with weighted edges, we will use subscripts w_{ij} to denote the weight on edge $\{i, j\}$.
- If M is a matrix, we will use round parentheses $M(i, j)$ to denote the entry in row i and column j of M.

- If M is a matrix, we will use subscripts $\hat{M}_{ij}$ to denote the matrix obtained from M by removing its ith row and jth column.

2.1. Counting spanning trees. Let G be a graph with n vertices labeled as $V(G) = \{1, 2, \dots, n\}$. The **Laplacian matrix** of G, denoted $\mathcal{L}(G)$ is an $n \times n$ matrix whose rows and columns are indexed by vertices of G. The entries of the Laplacian matrix are defined by

$$\mathcal{L}(G)(i,j) = \begin{cases} \deg(v) & \text{if } i = j, \\ -1 & \text{if } i \neq j \text{ and } \{i,j\} \in E(G), \\ 0 & \text{otherwise.} \end{cases}$$

Clearly the rows (or columns) of $\mathcal{L}(G)$ sum to 0, so $\mathcal{L}(G)$ is singular. However, the Matrix-Tree Theorem states that if we delete any row and any column of $\mathcal{L}(G)$, then the determinant of the resulting matrix counts the number of spanning trees in G. Specifically, let $\hat{\mathcal{L}}_{ij}(G)$ denote the matrix obtained by deleting row i and column j from $\mathcal{L}(G)$. Let $\tau(G)$ denote the number of spanning trees in G.

Matrix-Tree Theorem. *Let G be a connected graph with $V(G) = \{1, 2, \dots, n\}$. For any vertices $i, j \in V(G)$ (not necessarily distinct),*

$$\tau(G) = (-1)^{i+j} \det \hat{\mathcal{L}}_{ij}(G).$$

More generally, if we assign a weight $w_{ij} = w_{ji}$ to each edge $\{i,j\} \in E(G)$ (and define $w_{ij} = 0$ if $\{i,j\} \notin E(G)$), then we can define a **weighted Laplacian matrix**, $\mathcal{L}(G; w)$ by

$$\mathcal{L}(G;w)(i,j) = \begin{cases} \sum_{k=1}^{n} w_{ik} & \text{if } i = j, \\ -w_{ij} & \text{if } i \neq j. \end{cases}$$

In this case, there is a weighted analogue of the Matrix-Tree Theorem as well.

Weighted Matrix-Tree Theorem. *Let G be a connected graph with $V(G) = \{1, 2, \dots, n\}$. Let $w_{ij} = w_{ji}$ be a weighting on the edges of G. For any vertices $i, j \in V(G)$ (not necessarily distinct),*

$$\sum_{T} \left(\prod_{\{k,\ell\} \in T} w_{k\ell} \right) = (-1)^{i+j} \det \hat{\mathcal{L}}_{ij}(G; w),$$

where the sum is taken over all spanning trees of G and the product is over all edges in a given spanning tree.

The quantity on the lefthand side of the weighted Matrix-Tree Theorem is called the **weighted spanning tree enumerator** for the graph G. We can view a simple graph (whose edges are, a priori, unweighted) as a weighted graph in which $w_{ij} = 1$ for each edge $\{i,j\} \in E(G)$. In this case, the weighted Laplacian matrix is the standard Laplacian matrix, and the weighted spanning tree enumerator for G simplifies to count the number of spanning trees in G. Thus the Weighted Matrix Tree Theorem can be viewed as a generalization of the classical Matrix-Tree Theorem.

2.2. Schur complements in block matrices. In this paper, we will be primarily concerned with counting spanning trees in bipartite graphs with an aim towards developing a better understanding of Conjecture 1.3. If G is a bipartite graph whose vertex set is partitioned as $V(G) = X \cup Y$, then the Laplacian matrix of G can be naturally decomposed into blocks. The method of Schur complements is useful for simplifying calculations of the determinant of a block matrix. First we will review this method.

Let $M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$ be a square block matrix with the property that A is a square $n \times n$ matrix and D is a *nonsingular* square $m \times m$ matrix. The **Schur complement** of the block D in the matrix M is the $n \times n$ matrix $A - BD^{-1}C$. A fundamental result (see **[3]**) is that for such a matrix M,

$$\det(M) = \det(D)\det(A - BD^{-1}C). \tag{1}$$

Notice that if $M = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ is a 2×2 matrix with $d \neq 0$, this result gives the familiar determinant formula $\det(M) = d\left(a - \frac{bc}{d}\right) = ad - bc$.

When M is the Laplacian matrix of a bipartite graph G, the matrices A and D are the degree matrices for the vertices in G. Thus the Schur complement can be used to reduce the size of the matrix whose determinant we will compute in the Matrix-Tree Theorem.

2.3. Spanning trees in bipartite graphs. Let G be a connected bipartite graph whose vertices are partitioned as $V(G) = X \cup Y$. Suppose that $|X| = n$ and $|Y| = m$. Without loss of generality, we can fix a labeling of the vertices in X as $X = \{1, 2, \dots, n\}$. Now we can partition the vertices of Y according to their neighbors in X. For any subset $S \subseteq X$, we define a counting variable

$$x_S := \#\{y \in Y \ : \ N(y) = S\}.$$

Here, $N(y) = \{x \in X \ : \ \{x, y\} \in E(G)\}$ denotes the set of neighbors of y. Throughout, we will treat the vertices in X as being fixed vertices, we treat the vertices in Y and the edges in G as quantities that vary as a function of $\{x_S \ : \ S \subseteq X\}$. Our goal now is to understand (or, in some cases, explicitly compute) the number of spanning trees in G as a function of $\{x_S \ : \ S \subseteq X\}$.

When G is bipartite, its Laplacian matrix is a block matrix of the form

$$\mathcal{L}(G) = \begin{bmatrix} D_X & -Z \\ -Z^T & D_Y \end{bmatrix},$$

where D_X (respectively D_Y) is an $n \times n$ (resp. $m \times m$) diagonal matrix whose diagonal entries encode the degrees of the vertices in X (resp. Y). The matrix Z is an $n \times m$ matrix whose column corresponding to a vertex $y \in Y$ has a 1 in the row indexed by $x \in X$ if $x \in N(y)$ and a 0 otherwise.

Since G is connected it has no isolated vertices and hence the matrices D_X and D_Y are nonsingular. Thus we may compute the Schur complement of the block D_Y in the matrix $\mathcal{L}(G)$, which is the $n \times n$ matrix

$$\mathcal{S}(G) := D_X - ZD_Y^{-1}Z^T.$$

Based on the structure of $\mathcal{L}(G)$, we can compute the entries of $\mathcal{S}(G)$. For any $i, j \in X$ (not necessarily distinct) and any $y \in Y$, the entry $Z(i, y)$ is equal to 1 if $\{i, y\} \in E(G)$ and is equal to 0 otherwise. Similarly, the entry $D_Y^{-1}Z^T(y, j)$ is equal to $\frac{1}{\deg(y)}$ if $\{y, j\} \in E(G)$ and is equal to 0 otherwise. So when we compute the entry of the matrix $ZD_Y^{-1}Z^T$ in position (i, j) as the dot product of the ith row of Z with the jth column of $D_Y^{-1}Z^T$, there will be a nonzero contribution to the sum for each vertex $y \in Y$ that is a common neighbor of both i and j. Therefore,

$$ZD_Y^{-1}Z^T(i,j) = \sum_{\{y\in Y\ :\ i,j\in N(y)\}} \frac{1}{\deg(y)} = \sum_{\substack{S\subseteq X\\ i,j\in S}} \frac{x_S}{|S|}.$$

Similarly, for any $i \in X$, the degree of vertex i can be written as $\deg(i) = \sum_{\substack{S\subseteq X\\ i\in S}} x_S$. Thus

$$\mathcal{S}(G)(i,j) = \begin{cases} \displaystyle\sum_{\substack{S\subseteq X\\ i\in S}} \left(x_S - \frac{x_S}{|S|}\right) & \text{if } i = j \\ \displaystyle -\sum_{\substack{S\subseteq X\\ i,j\in S}} \frac{x_S}{|S|} & \text{if } i \neq j. \end{cases}$$

Now we observe that the matrix $\mathcal{S}(G)$ can be viewed as the weighted Laplacian matrix of a complete graph on n vertices in which the edge $\{i,j\}$ receives a weight $w_{ij} = \sum_{\substack{S\subseteq X\\ i,j\in S}} \frac{x_S}{|S|}$. Indeed, for any $i \in \{1,2,\ldots,n\}$, we compute

$$\sum_{j\neq i} w_{ij} = \sum_{j\neq i} \sum_{\substack{S\subseteq X\\ i,j\in S}} \frac{x_S}{|S|} = (|S|-1) \sum_{\substack{S\subseteq X\\ i\in S}} \frac{x_S}{|S|} = \sum_{\substack{S\subseteq X\\ i\in S}} \left(x_S - \frac{x_S}{|S|}\right),$$

so $\mathcal{S}(G)(i,i) = \sum_{j\neq i} w_{ij}$. This motivates the following definition.

Definition 2.1. *Let G be a simple bipartite graph whose vertex set is partitioned as $V(G) = X \cup Y$ with $X = \{1,2,\ldots,n\}$. Define an **associated weighted complete graph** $A(G)$ on vertex set X in which the edge $\{i,j\}$ has weight $w_{ij} = w_{ji} := \sum_{\substack{S\subseteq X\\ i,j\in S}} \frac{x_S}{|S|}$.*

From these observations, we are able to realize that the number of spanning trees in a bipartite graph G can be computed in terms of the weighted spanning tree enumerator from its associated weighted complete graph $A(G)$.

Theorem 2.2. *Let G be a simple connected bipartite graph whose vertex set is partitioned as $V(G) = X \cup Y$ with $X = \{1,2,\ldots,n\}$. Let $A(G)$ be the associated weighted complete graph to G and let $\mathcal{S}(G) = \mathcal{L}(A(G);w)$ be its weighted Laplacian matrix. For any $i,j \in X$ (not necessarily distinct), we have*

$$\tau(G) = \prod_{y\in Y} \deg(y) \cdot (-1)^{i+j} \det \hat{\mathcal{S}}_{ij}(G),$$

where $\hat{\mathcal{S}}_{ij}(G)$ is the matrix obtained by removing the ith row and jth column from $\mathcal{S}(G)$.

Proof: Let $\mathcal{L}(G)$ be the (unweighted) Laplacian matrix of G. As above, write $\mathcal{L}(G) = \begin{bmatrix} D_X & -Z \\ -Z^T & D_Y \end{bmatrix}$. Let $\mathcal{S}(G)$ be the Schur complement of the block D_Y in the matrix $\mathcal{L}(G)$, which can be realized as the weighted Laplacian matrix of $A(G)$. The only subtlety that arises now is that we know $\tau(G) = (-1)^{i+j} \det \hat{L}_{ij}(G)$ is the determinant of an $(m+n-1) \times (m+n-1)$ matrix, while $\hat{\mathcal{S}}_{ij}(G)$ is an $(n-1) \times (n-1)$ matrix.

Let $M = \hat{\mathcal{L}}_{ij}(G)$ be the reduced Laplacian matrix of G obtained by deleting the row corresponding to vertex i and the column corresponding to vertex j from $\mathcal{L}(G)$. Then M is a block matrix of the form $M = \begin{bmatrix} \widetilde{D} & -B \\ -C & D_Y \end{bmatrix}$, where

(1) $\widetilde{D}$ is obtained from D_X by removing the row corresponding to vertex i and the column corresponding to vertex j (and hence $\widetilde{D}$ is a diagonal matrix whose entries are the degrees of the vertices in X except for i and j,
(2) B is obtained from Z by removing the row corresponding to vertex i, and
(3) C is obtained from Z^T by removing the column corresponding to vertex j.

On the one hand, we know that $\tau(G) = (-1)^{i+j}\det(M)$ by the Matrix-Tree Theorem. Now we claim that the Schur complement of the block D_Y in M is the same as the matrix $\hat{\mathcal{S}}_{ij}(G)$ obtained from $\mathcal{S}(G)$ by removing its ith row and jth column, which is to say $\widetilde{D} - BD_Y^{-1}C = \hat{\mathcal{S}}_{ij}(G)$. Assuming that this is true, the proof will be complete since by Equation (1),

$$\det(M) = \det(D_Y)\det(\widetilde{D} - BD_Y^{-1}C) = \det(D_Y)\det\hat{\mathcal{S}}_{ij}(G) = \prod_{y\in Y}\deg(y)\cdot\det\hat{\mathcal{S}}_{ij}(G).$$

Choose $k, \ell \in X$ (not necessarily distinct) such that $k \neq i$ and $\ell \neq j$. Since the kth row of B is equal to the kth row of $-Z$ and the ℓth column of C is equal to the ℓth row of $-Z^T$, it follows that $BD_Y^{-1}C(k,\ell) = ZD_Y^{-1}Z^T(k,\ell)$. From here it follows that $\widetilde{D} - BD_Y^{-1}C = \hat{\mathcal{S}}_{ij}(G)$, as desired. $\square$

As a corollary, we obtain a nice formula for the number of spanning trees in a connected bipartite graph from the Weighted Matrix-Tree Theorem.

Corollary 2.3. *Let G be a connected, simple bipartite graph whose vertices are partitioned as $V(G) = X \cup Y$ with $|X| = n$. Then*

$$\tau(G) = \prod_{y\in Y}\deg(Y)\cdot\sum_{T \text{ spanning in } K_n}\left(\prod_{\{i,j\}\in E(T)}\left(\sum_{\substack{S\subseteq X\\ i,j\in S}}\frac{x_S}{|S|}\right)\right).$$

Observe that $|Y| = \sum_{S\subseteq X} x_S$ and recall that for any $i \in X$, $\deg(i) = \sum_{\substack{S\subseteq X\\ i\in S}} x_S$. This leads us to the following restatement of the main Conjecture 1.3 that depends only on the variables $\{x_S : S \subseteq X\}$.

Conjecture 2.4. *Let G be a simple, connected bipartite graph whose vertices are partitioned as $V(G) = X \cup Y$ with $|X| = n$. For any $1 \leq i, j \leq n$, we have*

$$n\cdot\left(\sum_{S\subseteq X} x_S\right)\cdot(-1)^{i+j}\det\hat{\mathcal{S}}_{ij}(G) \leq \prod_{i=1}^{n}\left(\sum_{\substack{S\subseteq X\\ i\in S}} x_S\right). \tag{2}$$

3. Evidence for the main conjecture.

Theorem 3.1. *Let G be a connected, simple bipartite graph for which Conjecture 1.3 holds. Let u be a new vertex not in $V(G)$, and let v be a vertex in $V(G)$. Let G' be the graph obtained by adding the edge $\{u,v\}$ to G. Then Conjecture 1.3 holds for G' as well.*

Proof: Suppose the vertices of G are partitioned as $V(G) = X \cup Y$, and assume without loss of generality that $v \in X$. Then the vertices of G' are partitioned as $X \cup (Y \cup \{u\})$.

Since $\deg_{G'}(u) = 1$, our goal is to prove that

$$\tau(G') \leq \frac{(\deg_G(v)+1) \cdot \prod_{\substack{w \in V(G) \\ w \neq v}} \deg_G(w)}{|X| \cdot (|Y|+1)}.$$

First, since u has only one neighbor in G', any spanning tree in G' must contain the edge $\{u,v\}$. So $\tau(G) = \tau(G')$. Thus it is sufficient to prove that

$$\frac{\prod_{w \in V(G)} \deg_G(w)}{|X| \cdot |Y|} \leq \frac{(\deg_G(v)+1) \cdot \prod_{\substack{w \in V(G) \\ w \neq v}} \deg_G(w)}{|X| \cdot (|Y|+1)}, \tag{3}$$

by our hypothesis that $\tau(G') = \tau(G)$ is bounded from above by the quotient on the lefthand side of Equation (3).

We can compute that

$$\begin{aligned} |X| \cdot (|Y|+1) \cdot \deg_G(v) &= |X| \cdot |Y| \cdot \deg_G(v) + |X| \cdot \deg_G(v) \\ &\leq |X| \cdot |Y| \cdot \deg_G(v) + |X| \cdot |Y| \\ &= |X| \cdot |Y| \cdot (\deg_G(v)+1), \end{aligned}$$

since $v \in X$ and hence $\deg_G(v) \leq |Y|$. Thus

$$\frac{\deg_G(v)}{|X| \cdot |Y|} \leq \frac{\deg_G(v)+1}{|X| \cdot (|Y|+1)},$$

and Equation (3) follows by multiplying both sides by $\prod_{\substack{w \in V(G) \\ w \neq v}} \deg_G(w)$.

□

Corollary 3.2. *Conjecture 1.3 holds when G is a tree.*

Proof: Conjecture 1.3 clearly holds when G consists of a single edge, and any tree can be obtained from a single edge by inductively adding leaf vertices. □

Therefore we need only consider bipartite graphs in which $x_S = 0$ for all $|S| \leq 1$, and we will only consider such graphs for the remainder of the paper.

Corollary 3.3. *Conjecture 1.3 holds when $|X| = 2$.*

Proof: Let G be a bipartite graph with $|X| = 2$. By Theorem 3.1, it suffices to prove that Conjecture 1.3 holds if $\deg(y) \geq 2$ for all $y \in Y$; i.e, in the case that G is the complete bipartite graph $K_{2,m}$. The result is clear in this case. □

Theorem 3.4. *Conjecture 1.3 holds when $|X| \leq 5$.*

Proof: First suppose $|X| = 3$. As in Section 2, let $w_{ij} = \sum_{\substack{S \subseteq X \\ i,j \in S}} \frac{x_S}{|S|}$. By Equation (2), we need to show that

$$\begin{aligned} & 3 \cdot (x_{\{1,2\}} + x_{\{1,3\}} + x_{\{2,3\}}) \cdot (w_{1,2} \cdot w_{1,3} + w_{1,2} \cdot w_{2,3} + w_{1,3} \cdot w_{2,3}) \\ \leq \quad & (x_{\{1,2\}} + x_{\{1,3\}} + x_{\{1,2,3\}}) \cdot (x_{\{1,2\}} + x_{\{2,3\}} + x_{\{1,2,3\}}) \cdot (x_{\{1,3\}} + x_{\{2,3\}} + x_{\{1,2,3\}}) \end{aligned}$$

The difference between the righthand side and lefthand side of this inequality expands as

$$\begin{aligned} & \frac{1}{4}x_{\{1,2\}}^2 x_{\{1,3\}} + \frac{1}{4}x_{\{1,2\}} x_{\{1,3\}}^2 + \frac{1}{4}x_{\{1,2\}}^2 x_{\{2,3\}} + \frac{1}{4}x_{\{1,2\}} x_{\{2,3\}}^2 + \frac{1}{4}x_{\{1,3\}} x_{\{2,3\}}^2 + \frac{1}{4}x_{\{1,3\}}^2 x_{\{2,3\}} \\ & + \frac{1}{4}x_{\{1,2\}} x_{\{1,3\}} x_{\{1,2,3\}} + \frac{1}{4}x_{\{1,2\}} x_{\{2,3\}} x_{\{1,2,3\}} + \frac{1}{4}x_{\{1,3\}} x_{\{2,3\}} x_{\{1,2,3\}} - \frac{1}{4}x_{\{1,2\}} x_{\{1,3\}} x_{\{2,3\}}. \end{aligned}$$

Without loss of generality assume that $x_{\{1,2\}} \geq x_{\{1,3\}}$ so that $x_{\{1,2\}}^2 x_{\{2,3\}} \geq x_{\{1,2\}} x_{\{1,3\}} x_{\{2,3\}}$. This proves that the desired inequality is nonnegative since $x_{\{1,2\}}^2 x_{\{2,3\}} - x_{\{1,2\}} x_{\{1,3\}} x_{\{2,3\}} \geq 0$ and all other terms are nonnegative.

When $|X| \geq 3$, the equations on the left and right side of Equation (2) become quite unwieldy. We wrote code in Sage **[5]** to carry out these calculations for us. The code is included in the appendix.

When $|X| = 4$, the calculations showed that there are 751 terms with positive coefficients and only four terms with negative coefficients when we compute the difference between the righthand side and the lefthand side of Equation (2). Twenty of the 755 terms that appear in the expansion are

$$\begin{aligned} & \frac{1}{4}x_{\{1,2\}}^2 x_{\{1,3\}} x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{1,2\}} x_{\{1,3\}}^2 x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{1,2\}}^2 x_{\{2,3\}} x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{1,2\}} x_{\{2,3\}}^2 x_{\{1,2,3,4\}} \\ & + \frac{1}{4}x_{\{1,3\}}^2 x_{\{2,3\}} x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{1,3\}} x_{\{2,3\}}^2 x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{1,2\}}^2 x_{\{1,4\}} x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{1,2\}} x_{\{1,4\}}^2 x_{\{1,2,3,4\}} \\ & + \frac{1}{4}x_{\{1,2\}}^2 x_{\{2,4\}} x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{1,2\}} x_{\{2,4\}}^2 x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{1,4\}}^2 x_{\{2,4\}} x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{1,4\}} x_{\{2,4\}}^2 x_{\{1,2,3,4\}} \\ & + \frac{1}{4}x_{\{1,3\}}^2 x_{\{1,4\}} x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{1,3\}} x_{\{1,4\}}^2 x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{1,3\}}^2 x_{\{3,4\}} x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{1,3\}} x_{\{3,4\}}^2 x_{\{1,2,3,4\}} \\ & + \frac{1}{4}x_{\{1,4\}}^2 x_{\{3,4\}} x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{1,4\}} x_{\{3,4\}}^2 x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{2,3\}}^2 x_{\{2,4\}} x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{2,3\}} x_{\{2,4\}}^2 x_{\{1,2,3,4\}} \\ & + \frac{1}{4}x_{\{2,3\}}^2 x_{\{3,4\}} x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{2,3\}} x_{\{3,4\}}^2 x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{2,4\}}^2 x_{\{3,4\}} x_{\{1,2,3,4\}} + \frac{1}{4}x_{\{2,4\}} x_{\{3,4\}}^2 x_{\{1,2,3,4\}} \\ & - \frac{1}{4}x_{\{1,2\}} x_{\{1,3\}} x_{\{2,3\}} x_{\{1,2,3,4\}} - \frac{1}{4}x_{\{1,2\}} x_{\{1,4\}} x_{\{2,4\}} x_{\{1,2,3,4\}} - \frac{1}{4}x_{\{1,3\}} x_{\{1,4\}} x_{\{3,4\}} x_{\{1,2,3,4\}} \\ & - \frac{1}{4}x_{\{2,3\}} x_{\{2,4\}} x_{\{3,4\}} x_{\{1,2,3,4\}}. \end{aligned}$$

Here notice that each negative term has the form $x_{\{a,b\}} x_{\{a,c\}} x_{\{b,c\}} x_{\{1,2,3,4\}}$ for some choice of distinct $a, b, c \in \{1, 2, 3, 4\}$. For each such term, we can assume without loss of generality that $x_{\{a,b\}} \geq x_{\{a,c\}}$ so that

$\frac{1}{4}x^2_{\{a,b\}}x_{\{b,c\}} - \frac{1}{4}x_{\{a,b\}}x_{\{a,c\}}x_{\{b,c\}} \geq 0$. Since this assumption can be made independently for all four choices of distinct a, b, c and all other terms in the above array are nonnegative, it follows that the above quantity is nonnegative. In addition to these 20 terms, the difference between the righthand side and the lefthand side of Equation (2) contains 735 more monomial terms, each of which has a positive coefficient. Therefore, Conjecture 2.4 holds when $|X| = 4$.

When $|X| = 5$, our calculations showed that there are 123,435 terms with positive coefficients and only 140 terms with negative coefficients when we compute the difference between the righthand side and the lefthand side of Equation (2). It seems a significant waste of paper to write out all of these negative terms, but they all fall into one of five families. As in the previous two cases, we will exhibit the general form of these five families of terms with negative coefficients, together with five corresponding families of terms with positive coefficients whose value will exceed the negative contributions.

(1) Choose distinct $a, b \in \{1,2,3,4,5\}$ and let $\{c,d,e\} = \{1,2,3,4,5\} \setminus \{a,b\}$. There are 10 ways to make such a choice. The difference between the right and lefthand side contains the following terms:

$$\frac{1}{16}x^2_{\{a,b\}}x^2_{\{a,b,c,d\}}x_{\{a,b,c,e\}} + \frac{1}{16}x^2_{\{a,b\}}x_{\{a,b,c,d\}}x^2_{\{a,b,c,e\}} + \frac{1}{16}x^2_{\{a,b\}}x^2_{\{a,b,c,d\}}x_{\{a,b,d,e\}}$$
$$+\frac{1}{16}x^2_{\{a,b\}}x_{\{a,b,c,d\}}x^2_{\{a,b,d,e\}} + \frac{1}{16}x^2_{\{a,b\}}x^2_{\{a,b,c,e\}}x_{\{a,b,d,e\}} + \frac{1}{16}x^2_{\{a,b\}}x_{\{a,b,c,e\}}x^2_{\{a,b,d,e\}}$$
$$-\frac{1}{32}x^2_{\{a,b\}}x_{\{a,b,c,d\}}x_{\{a,b,c,e\}}x_{\{a,b,d,e\}}.$$

(2) Choose distinct $a, b, c \in \{1,2,3,4,5\}$. There are 10 ways to make such a choice. The difference between the right and lefthand side contains the following terms:

$$\frac{1}{4}x^2_{\{a,b\}}x_{\{a,c\}}x^2_{\{1,2,3,4,5\}} + \frac{1}{4}x_{\{a,b\}}x^2_{\{a,c\}}x^2_{\{1,2,3,4,5\}} + \frac{1}{4}x^2_{\{a,b\}}x_{\{b,c\}}x^2_{\{1,2,3,4,5\}}$$
$$+\frac{1}{4}x_{\{a,b\}}x^2_{\{b,c\}}x^2_{\{1,2,3,4,5\}} + \frac{1}{4}x^2_{\{a,c\}}x_{\{b,c\}}x^2_{\{1,2,3,4,5\}} + \frac{1}{4}x^2_{\{a,c\}}x_{\{b,c\}}x^2_{\{1,2,3,4,5\}}$$
$$-\frac{1}{4}x_{\{a,b\}}x_{\{a,c\}}x_{\{b,c\}}x^2_{\{1,2,3,4,5\}}$$

(3) Choose distinct $a, b, c \in \{1,2,3,4,5\}$ and choose $d \neq a, b, c$. There are 20 ways to make such a choice. The difference between the right and lefthand side contains the following terms:

$$\frac{1}{4}x^2_{\{a,b\}}x_{\{a,c\}}x_{\{a,b,c,d\}}x_{\{1,2,3,4,5\}} + \frac{1}{4}x_{\{a,b\}}x^2_{\{a,c\}}x_{\{a,b,c,d\}}x_{\{1,2,3,4,5\}}$$
$$+\frac{1}{4}x^2_{\{a,b\}}x_{\{b,c\}}x_{\{a,b,c,d\}}x_{\{1,2,3,4,5\}} + \frac{1}{4}x_{\{a,b\}}x^2_{\{b,c\}}x_{\{a,b,c,d\}}x_{\{1,2,3,4,5\}}$$
$$+\frac{1}{4}x^2_{\{a,c\}}x_{\{b,c\}}x_{\{a,b,c,d\}}x_{\{1,2,3,4,5\}} + \frac{1}{4}x_{\{a,c\}}x^2_{\{b,c\}}x_{\{a,b,c,d\}}x_{\{1,2,3,4,5\}}$$
$$-\frac{1}{4}x_{\{a,b\}}x_{\{a,c\}}x_{\{b,c\}}x_{\{a,b,c,d\}}x_{\{1,2,3,4,5\}}$$

(4) Choose distinct $a, b \in \{1,2,3,4,5\}$ and choose $c \neq a, b$. Let $\{d,e\} = \{1,2,3,4,5\} \setminus \{a,b,c\}$. There are 30 ways to make such a choice. The difference between the right and lefthand side contains the

following terms:

$$\begin{aligned}
&\frac{5}{8}x^2_{\{a,b\}}x_{\{a,c\}}x_{\{a,b,d,e\}}x_{\{1,2,3,4,5\}}+\frac{5}{8}x_{\{a,b\}}x^2_{\{a,c\}}x_{\{a,b,d,e\}}x_{\{1,2,3,4,5\}}\\
+&\frac{5}{8}x^2_{\{a,b\}}x_{\{b,c\}}x_{\{a,b,d,e\}}x_{\{1,2,3,4,5\}}+\frac{5}{8}x_{\{a,b\}}x^2_{\{b,c\}}x_{\{a,b,d,e\}}x_{\{1,2,3,4,5\}}\\
+&\frac{5}{8}x^2_{\{a,c\}}x_{\{b,c\}}x_{\{a,b,d,e\}}x_{\{1,2,3,4,5\}}+\frac{5}{8}x_{\{a,c\}}x^2_{\{b,c\}}x_{\{a,b,d,e\}}x_{\{1,2,3,4,5\}}\\
-&\frac{1}{8}x_{\{a,b\}}x_{\{a,c\}}x_{\{b,c\}}x_{\{a,b,d,e\}}x_{\{1,2,3,4,5\}}
\end{aligned}$$

(5) Choose distinct $a, b, c \in \{1, 2, 3, 4, 5\}$ and choose distinct $S, T \subseteq \{1, 2, 3, 4, 5\}$ with $|S| = |T| = 4$ such that either $\{a, b, c\} \subseteq S$ or $\{a, b, c\} \subseteq T$ (or both). There are 70 ways to make such a choice. The difference between the right and lefthand side contains the following terms:

$$\begin{aligned}
&\frac{19}{64}x^2_{\{a,b\}}x_{\{a,c\}}x_Sx_T+\frac{19}{64}x_{\{a,b\}}x^2_{\{a,c\}}x_Sx_T+\frac{19}{64}x^2_{\{a,b\}}x_{\{b,c\}}x_Sx_T+\frac{19}{64}x_{\{a,b\}}x^2_{\{b,c\}}x_Sx_T\\
+&\frac{19}{64}x^2_{\{a,c\}}x_{\{b,c\}}x_Sx_T+\frac{19}{64}x_{\{a,c\}}x^2_{\{b,c\}}x_Sx_T-\frac{7}{64}x_{\{a,b\}}x_{\{a,c\}}x_{\{b,c\}}x_Sx_T
\end{aligned}$$

In each of these cases, one of the monomials with a positive coefficient is at least as large as the monomial with negative coefficient. Therefore the difference between the right and lefthand sides of Equation (2) is nonnegative when $|X| = 5$. ☐

The following corollary is immediate because one of the vertex partitions in a bipartite graph with 11 or fewer vertices has size at most 5.

Corollary 3.5. *Conjecture 1.3 holds for any connected bipartite graph with* 11 *or fewer vertices.*

4. Weighted spanning tree enumerators for Ferrers graphs. Let G be a bipartite graph whose vertex set is partitioned as $V(G) = X \cup Y$ with $|X| = n$. Consider the associated weighted complete graph $A(G)$ and, as before, let $\mathcal{S}(G) = \mathcal{L}(A(G))$ be its Laplacian matrix. Let $\Sigma(A(G))$ denote the weighted spanning tree enumerator for $A(G)$; i.e, $\Sigma(A(G)) = (-1)^{i+j} \det \hat{\mathcal{S}}_{ij}(G)$. The quantity on the lefthand side of Conjecture 2.4 can be written as

$$n \cdot \left(\sum_{S\subseteq X} x_S\right) \cdot \Sigma(A(G)). \tag{4}$$

Our goal now is to gain a better understanding of this quantity. We define a new $n\times n$ matrix $E'' = E''(G)$ with the property that for all $1 \le i, j \le n$,

$$E''(i,j) = \begin{cases} \displaystyle\sum_{\substack{S\subseteq X\\ i\in S}} x_S & \text{if } i = j,\\ \displaystyle\sum_{\substack{S\subseteq X\\ i\notin S, j\in S}} \frac{x_S}{|S|} & \text{if } i \neq j.\end{cases}$$

Proposition 4.1. *Let G be a bipartite graph whose vertex set is partitioned as $V(G) = X \cup Y$ with $|X| = n$.*

Then

$$\det(E'') = n \cdot \left(\sum_{S \subseteq X} x_S \right) \cdot \Sigma(A(G)).$$

Proof:

First consider the $n \times n$ matrix E that is obtained from $\mathcal{S}(G)$ by replacing its first column with the $n \times 1$ vector of all ones. If we compute the determinant of E by cofactor expansion along the first column, we see that

$$\det(E) = \sum_{i=1}^{n} (-1)^{i+1} \det\left(\hat{\mathcal{S}}_{i,1}(G)\right) = n \cdot \Sigma(A(G)), \tag{5}$$

by the Matrix-Tree Theorem.

Let $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_n$ be the column vectors of E. Let E' be the matrix with columns $\mathbf{c}_1', \mathbf{c}_2', \ldots, \mathbf{c}_n'$ that is obtained from E by the column operations

$$\mathbf{c}_j' = \left(\sum_{\substack{S \subseteq X \\ j \in S}} \frac{x_S}{|S|} \right) \mathbf{c}_1 + \mathbf{c}_j \qquad \text{for all } 2 \le j \le n,$$

and $\mathbf{c}_1' = \mathbf{c}_1$. Thus for any $2 \le j \le n$,

$$E'(i,j) = \begin{cases} \displaystyle\sum_{\substack{S \subseteq X \\ i \in S}} x_S & \text{if } i = j, \\ \displaystyle\sum_{\substack{S \subseteq X \\ i \notin S, j \in S}} \frac{x_S}{|S|} & \text{if } i \neq j. \end{cases}$$

For any $2 \le i \le n$, consider the ith partial row sum $\sum_{j=2}^{n} E'(i,j)$. For a given term x_S with $i \notin S$, the term $\frac{x_S}{|S|}$ appears in this sum $|S|$ times if $1 \notin S$, and $|S| - 1$ times if $1 \in S$. If $i \in S$, the term x_S appears exactly once. Therefore,

$$\sum_{j=2}^{n} E'(i,j) = \sum_{S \subseteq X} x_S - \sum_{\substack{S \subseteq X \\ 1 \in S, i \notin S}} \frac{x_S}{|S|}.$$

Similarly, in the first partial row sum $\sum_{j=2}^{n} E'(1,j)$ the only terms that appear are the terms $\frac{x_S}{|S|}$ with $1 \notin S$, each of which appears $|S|$ times. Thus $\sum_{j=2}^{n} E'(1,j) = \sum_{\substack{S \subseteq X \\ 1 \notin S}} x_S$.

Therefore for all $2 \le j \le n$, the jth column of E' is the same as the jth column of E''. We can obtain E'' from E' by replacing the first column of E' with the vector

$$\mathbf{c}_1'' = \left(\sum_{S \subseteq X} x_S \right) \mathbf{c}_1' - \sum_{j=2}^{n} \mathbf{c}_j'.$$

Since E' was obtained from E by elementary column operations, $\det(E) = \det(E')$. Since E'' was obtained from E' by multiplying its first column by $\sum_{S \subseteq X} x_S$ and then performing an elementary column operation,

$\det(E'') = \left(\sum_{S \subseteq X} x_S\right) \det(E')$. Therefore, by Equation (5),

$$\det(E'') = n \cdot \left(\sum_{S \subseteq X} x_S \right) \cdot \Sigma(A(G)).$$

$\square$

Corollary 4.2. *If G is a Ferrers graph, then* $\det(E'') = \prod_{i=1}^{n} \left(\sum_{\substack{S \subseteq X \\ i \in S}} x_S \right)$.

Proof: When G is a Ferrers graph, x_S is nonzero only when S has the form $\{1, 2, \ldots, k\}$ for some k. Therefore, $x_S = 0$ if there exist integers $i < j$ with $j \in S$ and $i \notin S$, meaning E'' is upper triangular. Therefore, $\det(E'')$ is the product of its diagonal entries, which is the product of the degrees of the vertices in X. $\square$

The matrix E'' seems quite appealing for the purposes of studying Conjecture 2.4 because the lefthand side of the inequality in that conjecture is exactly $\det(E'')$ and the product on the righthand side of the inequality is the product of the diagonal entries of E''. Unfortunately, E'' is not positive semidefinite in general (E'' can have negative eigenvalues even when $|X| = 3$), so results such as Hadamard's inequality (which continues to hold for asymmetric PSD matrices) cannot be immediately applied to directly prove Conjecture 2.4.

Acknowledgments. This research was conducted as part of the Seattle University College of Science & Engineering Summer Research Program. We gratefully acknowledge support from the Washington NASA Space Grant, as well as a generous donation from Judith and William Doyle.

REFERENCES

1. Richard Ehrenborg and Stephanie van Willigenburg. Enumerative properties of Ferrers graphs. *Discrete Comput. Geom.*, 32(4):481–492, 2004.

2. P. Erdős, C. Ko, and R. Rado. Intersection theorems for systems of finite sets. *Quart. J. Math. Oxford Ser. (2)*, 12:313–320, 1961.

3. Roger A. Horn and Charles R. Johnson. *Matrix analysis*. Cambridge University Press, Cambridge, second edition, 2013.

4. R. P. Stanley. *Combinatorics and Commutative Algebra*. Birkhäuser, Boston, 2. edition, 1996.

5. W. A. Stein et al. *Sage Mathematics Software (Version 6.2)*. The Sage Development Team, 2014. `http://www.sagemath.org`.

A.

B. . . Code

Given a bipartite graph G, our first goal is to write code to populate the weighted Laplacian matrix of the associated edge-weighted complete graph $A(G)$. This requires several small functions before the main code shown below in the function `makeKnLaplacian(n)`. It is easier to reindex the set X so that its elements are labeled as $\{0, 1, \ldots, n-1\}$. Furthermore, for a set $S \subseteq X$, the variable x_S is written by concatenating x with the elements of S. For example, if $S = \{1, 3, 4, 5\}$, the variable x_S is stored as `x1345` in Sage.

```
def listToVariableDictInput(L):
############################
### Input:  A list, L, of subsets of [0,...,n-1]
### Output:  A dictionary in which
### keys are variable names formatted as strings
### values are variable names formatted as variables
############################

       partDict = {}
       s = 'x'
       for i in L:
            s += stringToVar(i)
            partDict[s] = var(s)
            s = 'x'
       return partDict

def createElementInput(x,y,L,dic):
############################
### Input:
### L = a list of sets
### x, y = elements of [0,...,n-1]
### dic = a dictionary
### Output:  Weighted sum of variables x_S over S in L
### indexed by row x and column y
############################

       element = []
       toBeChecked = [x,y]
       for i in L:
            if set(toBeChecked).issubset(set(i)):
                 element.append(i)
       if x==y:
            elementSum = sumDiagonalInput(element,dic)
       else:
            elementSum = sumNonDiagonalInput(element,dic)
       return elementSum

def sumDiagonalInput(L,dic):
############################
### Input:  L = a list of sets
### Output:  \sum_{S \in L} \left(x_S - \frac{x_S}{|S|}\right)
############################

       value = 0
       for i in L:
            value += (dic['x'+stringToVar(i)]*(len(i)-1)/len(i))
       return value

def sumNonDiagonalInput(L,dic):
############################
### Input:  L = a list of sets
```

```
### Output:  −∑_{S∈L} x_S/|S|
############################

       value = 0
       for i in L:
           value += (dic['x'+stringToVar(i)]*(-1)/len(i))
       return value

 def stringToVar(L):
############################
### Input:  A subset of [0...n-1] formatted as a list, L,
### whose elements are separated by commas
### Output:  A string that concatenates the elements of L without commas
############################

       s = ''
       for i in L:
           s += str(i)
       return s
```

This is the main code that is used to populate the weighted Laplacian matrix of $A(G)$ as a function of the variables $\{x_S \ : \ S \subseteq X\}$.

```
 def makeKnLaplacian(n):
############################
### Input:  An integer n
### Output:  Laplacian of the edge-weighted complete graph on n vertices
### as a function of the variables x_S.
############################

       ### listCombos stores subsets of [0...n-1] as lists
       ### and only includes those subsets with at least 2 elements
       listCombos = list(Combinations(range(n)))
       listCombos = filter(lambda x:  len(x) > 1, listCombos)
       i = 0 # row index
       j = 0 # column index
       ### create dictionary indexed by variables x_S for each set S in listCombos
       partitionDict = listToVariableDictInput(listCombos)
       ### create columns of Laplacian matrix
       LapMatCols = []
       while i < n:
           column = []
           while j < n:
                column.append(createElementInput(i,j,listCombos,partitionDict))
                j += 1
           LapMatCols.append(column)
           j = 0
           i += 1
       ### return weighted Laplacian matrix with entires in a symbolic ring
       return matrix(SR,LapMatCols)
```

```
 def makeDegrees(n):
############################
### Input:  An integer n
### Output:  Product of vertex degrees in X
############################

       listCombos = list(Combinations(range(n)))
       listCombos = filter(lambda x:  len(x) > 1,listCombos)
       degrees = 1
       i = 0
       while i < n:
           deg = 0
           for x in listCombos:
                s = 'x'
                if i in x:
                    s += stringToVar(x)
                    deg += var(s)
           degrees *= deg
           i += 1
       return degrees

 def generateEquation(n):
############################
### Input:  An integer n
### Output:  The difference between the righthand side and lefthand side
### of the equation in Conjecture 2.4
############################

       listCombos = list(Combinations(range(n)))
       listCombos = filter(lambda x:  len(x) > 1, listCombos)
       ySize = 0
       for x in listCombos:
           s = 'x'
           s += stringToVar(x)
           ySize += var(s)
       M = makeKnLaplacian(n)
       LHS = n * ySize * det(M[[0..(n-2)],[0..(n-2)]])
       RHS = makeDegrees(n)
       eq = expand(RHS) - expand(LHS)
       return expand(eq)
```

Seattle University, Department of Mathematics, 901 12th Avenue, Seattle, WA 98122
Email address: **garrettf@seattleu.edu**

Seattle University, Department of Mathematics, 901 12th Avenue, Seattle, WA 98122
Email address: **klees@seattleu.edu**