8-2003

Enumerating spanning trees

Michelle Renee Brown

To the Graduate Council:

I am submitting herewith a thesis written by Michelle Renee Brown entitled "Enumerating spanning trees." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Mathematics.

Reid Davis, Major Professor

We have read this thesis and recommend its acceptance:

Accepted for the Council:
Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

To the Graduate Council:

I am submitting herewith a thesis written by Michelle R. Brown entitled "Enumerating Spanning Trees." I have examined the final paper copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Mathematics.

Reid Davis

Reid Davis, Major Professor

We have read this thesis and recommend its acceptance:

David F Anderson

Carl G. Wagner

Acceptance for the Council:

Vice Provost and Dean of Graduate Studies

ENUMERATING SPANNING TREES

A Thesis
Presented for the
Master of Science
Degree
The University of Tennessee, Knoxville

Michelle R. Brown

August 2003

DEDICATION

I would like to thank my major professor, Dr. Reid Davis, for his direction in the completion of this document, as well as my other committee members, Dr. David Anderson and Dr. Carl Wagner.

I would also like to thank my parents, Randall and Donna Lane, for their many years of support, and especially my husband, John Brown, for his encouragement and understanding.

Without all of you, I would not have been able to obtain my goals.

ABSTRACT

In 1889 Arthur Cayley stated his well-known and widely used theorem that there are n^{n-2} trees on n labeled vertices **[6, p. 70]**. Since he originally stated it, the theorem has received much attention: people have proved it in many different ways. In this paper we consider three of these proofs. The first is an algebraic result using Kirchhoff's Matrix Tree theorem. The second proof shows a one-to-one correspondence between trees on labeled vertices and sequences known as Prüfer codes. The final proof involves degree sequences and multinomial coefficients. In addition, we extend each of these three proofs to find a result for the number of spanning trees on the complete bipartite graph, and extend the first two results to count the number of spanning trees on the complete tripartite graph. We conclude with a brief generalization to the number of spanning trees on the complete k-partite graph.

TABLE OF CONTENTS

INTRODUCTION

An obvious graph theoretic question arises when one thinks about trees on n labeled vertices. Just how many such trees are there? The answer, n^{n-2}, was first published by Arthur Cayley in 1889 **[6, p. 70]**. Since then, Cayley's theorem has been the subject of many beautiful proofs, using both algebra and combinatorics. There are nine different proofs in Harary **[6, pp. 70 – 78]** alone, as well as more proofs in Aigner and Ziegler **[1, pp. 141 – 146]**. Here we look at three of these proofs.

In the first proof we begin with a complete graph, G, on n labeled vertices $\{1, 2, \ldots, n\}$ and then find the adjacency matrix, A, of G . We obtain the matrix M by subtracting A from the n x n matrix that has entries d_i = degree of vertex i down the diagonal, for all $i \in \{1, 2, \ldots, n\}$ and zeros everywhere else. We then obtain the number of spanning trees of G by using Kirchhoff's Matrix Tree theorem, which states that all cofactors of the matrix M are equal and that this common value is the number of spanning trees of G.

For the second proof, we show that there is a one-to-one correspondence between the number of trees on n labeled vertices, $\{1, 2, \ldots, n\}$, and the set of n – 1 tuples of integers $(a_1, a_2, \ldots, a_{n-1})$ with $1 \le a_i \le n$ for $1 \le i \le n - 2$ and $a_{n-1} = n$. Given a tree, T, on n vertices, such a tuple, which is known as the Prüfer code of T, is easily found using the following procedure. Remove the endpoint of smallest label and the edge incident to it, and record the label of the adjacent vertex. Continue this process with the remaining tree. The process terminates when only one vertex remains. Given such a sequence there also is a procedure for reconstructing the tree. Let $u_1 = \min([n]\backslash\{a_1, a_2, \ldots, a_{n-1}\})$ and connect u_1 to a_1. Now let $u_2 = \min([n]\backslash\{u_1\} \cup \{a_2, \ldots, a_n\})$, and connect u_2 to a_2. Then in general let $u_i = \min([n]\backslash\{u_1, \ldots, u_{i-1}\} \cup \{a_i, \ldots, a_n\})$, and connect u_i to a_i. The process terminates after n – 1 iterations.

The remaining proof uses a lemma which states that the number of trees on {1, 2, …, n} with degree sequence $(d_1, d_2, \ldots, d_n)$ – i.e., vertex i has degree d_i – is $\binom{n-2}{d_1-1, d_2-1, \ldots, d_n-1}$. Given this lemma, we can then count the number of trees on {1, 2, …, n} by summing over all possible degree sequences.

With these three proofs of Cayley's theorem under our belt, we then extend each of them to find the number of spanning trees on the complete bipartite graph. We then take the first two results and extend them to find the number of spanning trees on the complete tripartite graph. Finally we conclude with an extension to the number of spanning trees on the complete k-partite graph.

CHAPTER ONE
DEFINITIONS AND PRELIMINARY RESULTS

1.1 Definitions

The following common definitions were compiled primarily using Johnsonbaugh **[7]**, Krishnamurthy **[8]**, Wagner **[12]**, and Wilson **[14]**.

For the purposes of our work here let $\mathbb{P} = \{1, 2, \ldots\}$, the set of *positive integers*, and let $\mathbb{N} = \{0, 1, 2, \ldots\}$, the set of *nonnegative integers.* For $n \in \mathbb{N}$, define $[n] = \{1, 2, \ldots, n\}$ with $[0] = \varnothing$.

A *graph* G = (V, E) consists of a set V of elements called vertices, and a set E of edges, which consists of unordered pairs of elements from V. If $\{u, v\} \in E$, then vertices u and v are *adjacent* and edge {u, v} is *incident* with both of them. The *degree* of a vertex is the number of edges incident with it. A vertex of degree one is an *endpoint.* A sequence of positive integers $(d_1, d_2, \ldots, d_n)$ is a *degree sequence* on the graph G = ([n], E) if vertex i has degree d_i.

A *path* from v_0 to v_n is an alternating sequence of adjacent vertices and their shared edges beginning with vertex v_0 and ending with vertex v_n, $(v_0, e_1, v_1, e_2, \ldots, v_{n-1}, e_n, v_n)$, with no repeated edges (some sources refer to this as a *simple path*). Not allowing two edges to be associated with the same vertex set

$\{v_i, v_j,\}$, we may remove the edges from the sequence denoting the path as $(v_0, v_1, \ldots, v_n)$. A graph G is *connected* if there exists a path joining each vertex to every other. A *cycle* is a path that joins a vertex to itself. A graph with no cycles is called *acyclic*. A *tree* is an acyclic, connected graph.

A *loop* is a cycle in which an edge is incident with only one vertex. In other words, $\{u, v\}$ is a loop if $u = v$. A graph that has no loops is a *simple graph*. From this point forward, we consider only simple graphs.

The *complete graph on n vertices*, denoted K_n, is a graph in which every pair of distinct vertices is adjacent. The *complete bipartite graph on n and m vertices*, denoted $K_{n,m}$, is a graph whose set of vertices is partitioned into two sets, U, which has n vertices, and V, which has m vertices, such that vertices u and v are adjacent if and only if $u \in U$ and $v \in V$. For the purpose of this paper, we let $U = [n]$ and $V = \{n + 1, \ldots, n + m\}$. The *complete tripartite graph on p, q and r vertices*, denoted $K_{p,q,r}$, is a simple graph whose set of vertices is partitioned into three sets, U, which has p vertices, V, which has q vertices, and W, which has r vertices, such that u and v are adjacent if and only if they are not in the same set. For the purposes of this paper, we let $U = [p]$, $V = \{p + 1, \ldots, p + q\}$, and $W = \{p + q + 1, \ldots, p + q + r\}$. Similarly the *complete k-partite graph on $n_1, \ldots, n_k$ vertices*, denoted $K_{n_1,n_2,\ldots,n_k}$, is a graph whose set of vertices is partitioned into k sets, $V_1, V_2, \ldots, V_k$, such that vertex set V_i has n_i vertices and such that u is adjacent to v if and only if they are not in the same vertex set.

A *subgraph* of a graph $G = (V, E)$, is a graph with vertex set V' and edge set E', such that $V' \subseteq V$ and $E' \subseteq E$. A *spanning tree* of a graph G is a tree that is a subgraph of G containing all vertices of G.

The *adjacency matrix* of a graph on n labeled vertices is an n x n matrix $A = (a_{i,j})$ such that $a_{i,j} = 0$ if vertices i and j are not adjacent, and $a_{i,j} = 1$ if vertices i and j are adjacent.

As is well known, the following are equivalent definitions of the *multinomial coefficient* $\binom{n}{n_1, n_2, \ldots, n_k}$, where $n, n_1, \ldots, n_k \in \mathbb{P}$ and $n_1 + \ldots + n_k = n$.

(i) $\binom{n}{n_1, n_2, \ldots, n_k} = \binom{n}{n_1}\binom{n-n_1}{n_2}\cdots\binom{n-n_1-n_2-\ldots-n_{k-1}}{n_k} = \frac{n!}{n_1!n_2!\ldots n_k!}$

(ii) $\binom{n}{n_1, n_2, \ldots, n_k} = |\{f:[n] \to [k] \text{ such that } |f^{\leftarrow}(j)| = n_j, j = 1, \ldots, k\}|$, where

$f^{\leftarrow}(j) = \{i \in [n] \mid f(i) = j\}$.

1.2 Preliminary Results

Our first lemma is a well known result of introductory graph theory and can be found in Johnsonbaugh [7, p. 323].

Lemma 1.2.1

Let G = ([n], E) be a graph with |E| = m, and degree sequence $(d_1, d_2, \ldots, d_n)$. Then

$$\sum_{i=1}^{n} d_i = 2m.$$

Proof.

Summing over the degrees of all vertices we count each edge twice. ■

Lemma 1.2.2

For $n \geq 2$, every tree on n vertices has at least two vertices of degree 1 – i.e., every tree has at least two endpoints.

Proof.

Suppose we have a tree on n labeled vertices. Start at a vertex, say v_1, and move along one of the edges from v_1 to, say, v_2. If the degree of v_2 is one then we are done. If not, there is an edge from v_2 different from $\{v_1, v_2\}$, say to v_3. If the degree if v_3 is one, we are done. If not, then there is an edge from v_3, different than $\{v_2, v_3\}$. Note that this edge also does not connect v_3 to v_1, since a tree has no cycles. Continue this process. At each point we either encounter a vertex of degree one or we continue along to a new vertex. Since our tree is on n points, the process must terminate. Therefore there is a vertex of degree one.

Now suppose that we have a tree on n labeled vertices. From above, we know the tree has one endpoint. Start at that endpoint and follow the same procedure as above. We will similarly find another vertex of degree one. Hence the tree has two vertices of degree one **[12]**. ■

The following theorem is essential to our work and can be seen in Johnsonbaugh **[7, pp. 387 – 389]**.

Theorem 1.2.1

Let T be a graph with n vertices. Then the following are equivalent.

(i) T is a tree. That is, T is connected and acyclic.

(ii) T is connected and has n – 1 edges.

(iii) T is acyclic and has n – 1 edges.

Proof.

(i) ⇒ (ii). Suppose that T is connected and acyclic. We need to show that T has n – 1 edges. This can be proved by induction on n. Let n = 1, then T has one vertex and therefore no edges, thus the theorem is true for n = 1. Suppose it is true for an acylic, connected graph with n vertices. Let T be a connected, acyclic graph with n + 1 vertices. We know that T has an endpoint. Remove the endpoint and the incident edge. The remaining is a tree on n vertices. Thus by induction it has n – 1 edges. Hence T has n edges.

(ii) ⇒ (iii). Suppose that T is connected with n – 1 edges. We need to show that T is acyclic. Suppose not. Then T contains at least one cycle. Remove edges (but not vertices) from the cycles of T until the resulting graph, T*, is acylic. Note that removing an edge from a cycle does not disconnect a graph. Thus T* is also connected. Hence T* has n – 1 edges. However, we removed edges from T. Thus T has at least n edges, which is a contradiction. Hence T is acyclic.

(iii) ⇒ (i). Suppose that T is acylic with n – 1 edges. We need to show that T is a tree. T does not contain any loops and T cannot contain distinct edges e_1 and e_2 incident to the same set of vertices, as that would create a cycle. So T is a simple graph. Suppose that T is not connected. Let $T_1, T_2, \ldots, T_k$ be the components of T. As T is not connected, $k > 1$. Suppose further that T_i has n_i vertices. Each T_i is connected and acyclic, so T_i has $n_i - 1$ edges. However, this is impossible, as we would then have the following: The number of edges of T = n – 1 = The sum of the edges of each

$T_i = (n_1 - 1) + (n_2 - 1) + \ldots + (n_k - 1) < (n_1 + n_2 + \ldots + n_k) - 1$ (as $k > 1$) $= n - 1$.

Thus T is connected. Hence T is a tree. ■

Corollary 1.2.1

Given a tree T = ([n], E) with degree sequence $(d_1, d_2, \ldots, d_n)$, then $\sum_{i=1}^{n} d_i = 2n - 2$.

Proof.

By Theorem 1.2.1, T has n – 1 edges. Then by Lemma 1.2.1, $\sum_{i=1}^{n} d_i = 2n - 2$. ■

Corollary 1.2.2

Given a spanning tree T of $K_{n,m}$ with degree sequence

$(d_1, d_2, \ldots, d_n; d_{n+1}, d_{n+2}, \ldots, d_{n+m})$, then $\sum_{i=1}^{n+m} d_i = 2(n + m - 1) = 2n + 2m - 2$.

Proof.

By Theorem 1.2.1, as T is a tree, T has n + m − 1 edges. Then by Lemma 1.2.1,

$\sum_{i=1}^{n+m} d_i = 2(n + m - 1) = 2n + 2m - 2$. ■

Lemma 1.2.3

Given a spanning tree T of $K_{n,m}$ with vertex sets U and V and degree sequence

$(d_1, d_2, \ldots, d_n; d_{n+1}, d_{n+2}, \ldots, d_{n+m})$, then $\sum_{i=1}^{n} d_i = \sum_{i=n+1}^{n+m} d_i = m + n - 1$.

Proof.

Every edge of T connects an element from U to an element of V. ■

The last two results involving multinomial coefficients can be found in Krishnamurthy **[8, p. 69]**.

Lemma 1.2.4

$$\sum_{\substack{n_1 + \ldots + n_k = n \\ n_i \text{ nonnegative}}} \binom{n}{n_1, \ldots, n_k} = k^n.$$

Proof.

By definition (ii) of the multinomial coefficient, $\binom{n}{n_1, n_2, \ldots, n_k}$ counts the number of functions $f:[n] \to [k]$ such that $|f^{\leftarrow}(j)| = n_j$, $j = 1, \ldots, k$. Then summing over all $n_1 + \ldots + n_k = n$ we get all functions $f:[n] \to [k]$. Thus $\sum_{\substack{n_1 + \ldots + n_k = n \\ n_i \text{ nonnegative}}} \binom{n}{n_1, \ldots, n_k} = k^n$. ■

Lemma 1.2.5 (Multinomial Recurrence)

For all $n, k \in \mathbb{P}$, and $n_1, \ldots, n_k \in \mathbb{N}$ with $n_1 + \ldots + n_k = n$, then

$$\binom{n}{n_1, \ldots, n_k} = \binom{n-1}{n_1 - 1, \ldots, n_k} + \binom{n-1}{n_1, n_2 - 1, \ldots, n_k} + \ldots + \binom{n-1}{n_1, \ldots, n_{k-1}, n_k - 1}.$$

Proof.

Count the number of functions $f:[n] \longrightarrow [k]$ such that $|f^{\leftarrow}(j)| = n_j$, $j = 1, \ldots, k$ according to the values of $f(n)$. For instance, if $f(n) = 1$, there are $\binom{n-1}{n_1 - 1, \ldots, n_k}$ ways to map the remaining $n - 1$ elements. Then summing over the other possible values of $f(n)$, we get the formula. If any $n_j = 0$, then $\binom{n-1}{n_1, \ldots, n_j - 1, \ldots, n_k} = 0$, as is appropriate since there are then no functions of the type being counted for which n is mapped to j. ■

CHAPTER TWO
TREES ON n LABELED VERTICES

In this chapter we will be proving Cayley's Theorem using the three proofs outlined in the introduction.

Theorem 2.1 – Cayley's Theorem

Let $n \in \mathbb{P}$. There are n^{n-2} trees on n labeled vertices.

Proof 1 of Theorem 2.1.

For our first proof of Cayley's theorem we consider an approach used by Gustav Kirchhoff. To proceed, we use without proof the following theorem attributed to Kirchhoff, which is in many sources, one of which is Chartrand and Lesniak, [5].

Theorem 2.2 – Kirchoff's Matrix Tree Theorem

Given the adjacency matrix, A, of a connected graph G on n labeled vertices, and

$$M = -A + \begin{bmatrix} d_1 & 0 & \cdots & 0 \\ 0 & d_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & d_n \end{bmatrix},$$

where d_i = degree of vertex i, then all cofactors of M are equal, and their common value is the number of spanning trees of G.

Now, as the set of all trees on n labeled vertices is the same as the set of spanning trees of the complete graph K_n, we may use Kirchhoff's result to find the number of such trees.

We have

$$A = \begin{bmatrix} 0 & 1 & 1 & \cdots & 1 & 1 \\ 1 & 0 & 1 & \cdots & 1 & 1 \\ 1 & 1 & 0 & \cdots & 1 & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & 1 & \cdots & 0 & 1 \\ 1 & 1 & 1 & \cdots & 1 & 0 \end{bmatrix} \text{ and therefore}$$

$$M = -A + \begin{bmatrix} n-1 & 0 & \cdots & 0 \\ 0 & n-1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & n-1 \end{bmatrix} = \begin{bmatrix} n-1 & -1 & \cdots & -1 \\ -1 & n-1 & \cdots & -1 \\ \vdots & \vdots & \ddots & \vdots \\ -1 & -1 & \cdots & n-1 \end{bmatrix}.$$

By Kirchhoff's theorem, the number of trees on n labeled vertices is simply a cofactor of M. Using the cofactor associated with the first row and column of M we get that

$$M_{11} = (-1)^{1+1} \det \begin{bmatrix} n-1 & -1 & \cdots & -1 & -1 \\ -1 & n-1 & \cdots & -1 & -1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -1 & -1 & \cdots & n-1 & -1 \\ -1 & -1 & \cdots & -1 & n-1 \end{bmatrix} =^{1} \det \begin{bmatrix} n & 0 & \cdots & 0 & -n \\ 0 & n & \cdots & 0 & -n \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & n & -n \\ -1 & -1 & \cdots & -1 & n-1 \end{bmatrix}$$

$$=^{2} \det \begin{bmatrix} n & 0 & \cdots & 0 & -n \\ 0 & n & \cdots & 0 & -n \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & n & -n \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} = n^{n-2}.$$

Thus the number of trees on n labeled vertices is n^{n-2}. ■

---

[1] Subtract the last row from each of the first n – 2 rows.
[2] Add (1/n) times each of the first n – 2 rows to the last row.

This next proof of Cayley's theorem uses the Prüfer code of a tree, defined by Heinz Prüfer. Many versions of this proof have been published, one of which is Lovász **[9, pp. 34 and 348 – 249]**.

Proof 2 of Theorem 2.1.

It suffices to show there is a one-to-one correspondence between the set of trees on n labeled vertices, say [n], and the set of ordered n – 1 tuples of integers $(a_1, a_2, \ldots, a_{n-1})$ with $1 \leq a_i \leq n$ for $1 \leq i \leq n - 2$ and $a_{n-1} = n$.

First, we need to show that associated with each tree is such an n – 1 tuple. Let T be a tree on the vertices [n]. Remove the endpoint of smallest label and the edge incident to it and record the label of the adjacent vertex. Repeat this process with the remaining tree. The process terminates when only one vertex remains. Since there are always at least two endpoints, the final vertex remaining is n, so $a_{n-1} = n$. This procedure creates a sequence of n – 1 numbers associated with the tree we began with. This sequence is known as the Prüfer code of T. More formally, we are doing the following.

Let T = ([n], E) be a tree. Let $\varepsilon(T) = \{\text{endpoints of } T\}$. Define $u_1 = \min \varepsilon(T)$. Then there is a unique a_1 such that $\{u_1, a_1\} \in E$. Define $V_1 = [n]\backslash\{u_1\}$, $E_1 = E\backslash\{\{u_1, a_1\}\}$, and let $T_1 = (V_1, E_1)$. Clearly T_1 is also a tree. Now let $u_2 = \min \varepsilon(T_1)$. Then there exists a unique a_2 such that $\{u_2, a_2\} \in E_1$. Define $V_2 = [n]\backslash\{u_1, u_2\}$,
$E_2 = E\backslash\{\{u_1, a_1\},\{u_2, a_2\}\} = E_1\backslash\{\{u_2, a_2\}\}$, and $T_2 = (V_2, E_2)$. Again, T_2 is clearly a tree. Continue repeating the procedure. In general, we have that
$V_{i-1} = [n]\backslash\{u_1, \ldots, u_{i-1}\}$, $E_{i-1} = E\backslash\{\{u_j, a_j\} \mid 1 \leq j \leq i - 1\}$. Then $T_{i-1} = (V_{i-1}, E_{i-1})$ and $u_i = \min \varepsilon(T_{i-1})$. At the final step the tree $T_{n-2} = (V_{n-2}, E_{n-2})$ has two vertices joined by an edge $\{u_{n-1}, a_{n-1}\}$, where u_{n-1} is the smaller of the two vertices in V_{n-2}.

We have produced a sequence, $(u_1, u_2, \ldots, u_{n-1})$ which we will call the minimum endpoint sequence of T. We have also produced the sequence $(a_1, a_2, \ldots, a_{n-1})$ which is the Prüfer code of T. Note that $T = ([n], \{\{u_i, a_i\} | 1 \le i \le n-1\})$. By construction, the u_i are all distinct. Further, since every tree has at least two endpoints and each u_i is the smallest endpoint of a tree, no $u_i = n$. Thus the minimum endpoint sequence is a permutation of $[n-1]$ and $a_{n-1} = n$.

Now we need to show that the Prüfer code is unique – i.e., given two trees on n vertices with the same Prüfer code, the trees are identical. Using the notation from above, assume that $S = ([n], E)$ with minimum endpoint sequence $(s_1, s_2, \ldots, s_{n-1})$, that $T = ([n], F)$ with minimum endpoint sequence $(t_1, t_2, \ldots, t_{n-1})$, and that S and T have the same Prüfer code $(a_1, \ldots, a_{n-1})$.

Lemma 2.1

The endpoints of S are the elements of [n] which do not appear in $\{a_1, \ldots, a_{n-2}\}$. Hence $\varepsilon(S) = [n] \backslash \{a_1, \ldots, a_{n-2}\}$.

Proof.

First we need to show that for all $c \in \{a_1, \ldots, a_{n-2}\}$, c is not an endpoint of S. Suppose $c \in \{a_1, \ldots, a_{n-2}\}$ with $c \neq n$. Then we have that for some i, $c = a_i$ and therefore $\{u_i, c\} \in E$. Also, $c = u_j$ for some j, so $\{c, a_j\} \in E$. Then at the ith iteration, u_i and $\{u_i, c\}$ are removed, in which case c is a part of the resulting tree. Therefore, c could not have been removed prior to this iteration. Hence $c \neq u_j$ for $j < i$. Thus $c = u_j$ for some $j > i$, so it must be that $\{u_i, c\} \neq \{c, u_j\}$, and c is not an endpoint.

Now we need to show that $c \in [n] \backslash \{a_1, \ldots, a_{n-2}\}$ is an endpoint of S. For some i, we have that $c = u_i$. Thus $\{c, a_i\} \in E$. Now suppose that $\{c, x\} \in E$ for some $x \neq a_i$. Then $\{c, x\}$ must have been removed during an earlier iteration as otherwise $c \neq u_i$.

Thus $\{c, x\} = \{u_j, a_j\}$ for some $j < i$, which is a contradiction, since neither c nor x can be removed at an earlier iteration. Hence c is an endpoint, and the proof of the lemma is complete.

Therefore $\varepsilon(S) = [n]\backslash\{a_1, \ldots, a_{n-2}\} = \varepsilon(T)$. Thus by definition of the minimum endpoint sequence $s_1 = \min \varepsilon(S) = \min \varepsilon(T) = t_1$. Then $S_1 = ([n]\backslash\{s_1\}, E\backslash\{\{s_1, a_1\}\}$ and $T_1 = ([n]\backslash\{t_1\}, F\backslash\{\{t_1, a_1\}\}$. From above we see that $\varepsilon(S_1) = [n]\backslash\{s_1, a_2, \ldots, a_{n-2}\} =$ $= [n]\backslash\{t_1, a_2, \ldots, a_{n-2}\} = \varepsilon(T_1)$, and therefore $s_2 = \min \varepsilon(S_1) = \min \varepsilon(T_1) = t_2$. In general, $S_{i-1} = ([n]\backslash\{s_1, \ldots, s_{i-1}\}, E\backslash\{\{s_j, a_j\} \mid 1 \le j \le i-1\})$, and $T_{i-1} = ([n]\backslash\{t_1, \ldots, t_{i-1}\}, E\backslash\{\{t_j, a_j\} \mid 1 \le j \le i-1\})$. Then as in Lemma 2.1, the endpoints of S_i are the elements of $[n]\backslash\{s_1, \ldots, s_{i-1}\}$ which do not appear in the remainder of the Prüfer code, $\{a_i, \ldots, a_{n-2}\}$, and again similarly for T. Thus $\varepsilon(S_{i-1}) = [n]\backslash\{s_1, \ldots, s_i, a_{i+1}, \ldots, a_{n-2}\} = [n]\backslash\{t_1, \ldots, t_i, a_{i+1}, \ldots, a_{n-2}\} = \varepsilon(T_{i-1})$ and hence $s_i = \min \varepsilon(S_{i-1}) = \varepsilon(T_{i-1}) = t_i$ for all $1 \le i \le n-2$. Then $(s_1, s_2, \ldots, s_{n-1}) =$ $= (t_1, t_2, \ldots, t_{n-1})$, and therefore $S = T$. Note that in general

$$u_i = \min ([n]\backslash\{u_1, \ldots, u_{i-1}, a_i, \ldots, a_{n-1}\}). \qquad (1)$$

This constructively defines the tree from $(a_1, \ldots, a_{n-1})$, so if we know the Prüfer code we know T.

We have shown how to find the Prüfer code of a tree and that trees with the same Prüfer code are equal. Now we need to show that given an n-1 tuple of integers $(a_1, a_2, \ldots, a_{n-1})$ with $1 \le a_i \le n$ for $1 \le i \le n-2$ and $a_{n-1} = n$ there is a tree (which is unique from above) with this sequence as its Prüfer code. Let $(a_1, \ldots, a_{n-1})$ be such a sequence. Define u_i recursively by

$u_1 = \min ([n]\backslash\{ a_1, a_2, \ldots, a_{n-1}\})$, and $u_i = \min ([n]\backslash\{u_1, \ldots, u_{i-1}, a_i, \ldots, a_{n-1}\})$
for $i = 2, \ldots, n-1$

(note that $[n]\backslash\{u_1, \ldots, u_{i-1}, a_i, \ldots, a_{n-1}\}$ is never empty as there are n elements in [n], and at most n-1 elements in $\{u_1, \ldots, u_{i-1}, a_i, \ldots, a_{n-1}\}$). Once again, this definition compels $(u_1, \ldots, u_{i-1})$ to be a permutation of $[n-1]$. Define a graph $T = ([n],E)$ where $E = \{\{u_i, a_i\} | 1 \leq i \leq n\}$. Also, for $1 \leq i \leq n$, let $V_i = [n]\backslash\{u_1, \ldots, u_{i-1}\}$, $E_i = E\backslash\{\{u_j, a_j\} | 1 \leq j < i\}$ and $T_i = (V_i, E_i)$. We claim that the resulting graph T is a tree with Prüfer code $(a_1, a_2, \ldots, a_{n-1})$. We need only to show that u_i is an endpoint of the graph T_i and that no vertex with smaller label is an endpoint. Since u_i is adjacent to a_i in T_i and $a_i = u_j$ for some $j > i$ (that is, a_i is an endpoint of a later T_j), we can trace a path from each u_i to $a_{n-1} = n$, showing connectedness of T and each T_i.

Note that $\{u_i, a_i\}$ is an edge of T_i. Therefore u_i has a neighbor in T_i, namely a_i. Also, u_i cannot be adjacent to any other vertex of T_i, since if it were, then for some j, $\{a_j, u_j\}$ is also an edge of T_i that includes u_i. Then $j > i$ as u_j is a vertex of T_i. Also, we have that either $u_i = u_j$ or $u_i = a_j$. However, $u_i \neq u_j$ since $u_i \notin T_j$, and $u_i \neq a_j$ by definition of u_i. Thus u_i is an endpoint of T_i. Hence we have T and all T_i's are trees. Thus as T is a tree, it has a Prüfer code. Note that we have just defined each u_i exactly as it was defined in (1). Therefore u_i is the vertex of smallest label in T_i. So $(u_1, \ldots, u_{n-1})$ is the minimum endpoint sequence of T and $(a_1, \ldots, a_{n-1})$ is the Prüfer code of T.

We have now shown a one-to-one correspondence between the set of trees on n labeled vertices, and the set of ordered $n-1$ tuples $(a_1, a_2, \ldots, a_{n-1})$ with $1 \leq a_i \leq n$ for $1 \leq i \leq n-2$ and $a_{n-1} = n$. As the number of such sequences is n^{n-2}, we have that the number of trees on n labeled vertices is n^{n-2}. ■

Interesting properties of the Prüfer code.

Given a tree, T, with Prüfer code $(a_1, a_2, \ldots, a_{n-1})$ with $1 \le a_i \le n$ for $1 \le i \le n-2$ and $a_{n-1} = n$, the following properties are true. The number of times $i \in [n]$ occurs in the Prüfer code tells us the degree of i. For $i \in [n-1]$, the degree of i is one more than the number of times it occurs in the Prüfer code. The degree of n is the number of times it occurs in $(a_1, \ldots, a_{n-1})$. With that in mind, we can see that the endpoints of T are $[n] \setminus \{a_1, a_2, \ldots, a_{n-2}\}$ since each $a_i \in \{a_1, a_2, \ldots, a_{n-2}\}$ has at least degree two. The above proof also gives us a mechanical way to generate all trees on [n] using (1). For the final proof of Cayley's theorem, we first need the following lemma, which can be found in Wilf **[13, p. 163]**.

Lemma 2.2

For $n \ge 2$, given a sequence of positive integers $(d_1, \ldots, d_n)$ with $\sum_{i=1}^{n} d_i = 2n - 2$, the number of trees on [n] with degree sequence $(d_1, \ldots, d_n)$, denoted $T(d_1, \ldots, d_n)$, is $\binom{n-2}{d_1 - 1, d_2 - 1, \ldots, d_n - 1}$.

Proof.

This can be shown by induction on n.

For n = 2, we need positive integers d_1 and d_2 with $d_1 + d_2 = 2$. Thus $d_1 = d_2 = 1$ and therefore $\binom{0}{d_1 - 1, d_2 - 1} = \binom{0}{0,0} = 1$. The theorem holds in this case since the only tree on [2] is ([2],{{1,2}}).

Suppose the theorem is true for n – 1. We need to show it is true for $n \ge 3$. Note that some $d_i = 1$. If $\{d_1, \ldots, d_n\} = \{e_1, \ldots, e_n\}$ as multisets then $T(d_1, \ldots, d_n) = T(e_1, \ldots, e_n)$, since the same number of trees would occur, one tree being just a relabeling of the other tree. Thus we can say that without loss of generality $d_n = 1$. Then

$$T(d_1, \ldots, d_{n-1}, 1) = T(d_1 - 1, d_2, \ldots, d_{n-1}) + T(d_1, d_2 - 1, \ldots, d_{n-1}) + + \ldots + T(d_1, d_2, \ldots, d_{n-1} - 1). \quad (1)$$

Equation (1) follows from categorizing the trees enumerated by $T(d_1, \ldots, d_{n-1}, 1)$ according to the vertex adjacent to n (note that if some $d_i = 1$, the term on the RHS of (1) containing $d_i - 1$ is zero, as is appropriate, since two vertices, each of degree 1, cannot be adjacent in a tree with 3 or more vertices).

Now applying the induction hypothesis to the RHS of (1) and invoking the multinomial recurrence yields

$$T(d_1, \ldots, d_{n-1}, 1) = \binom{n-3}{d_1-2, d_2-1, \ldots, d_{n-1}-1} + \binom{n-3}{d_1-1, d_2-2, \ldots, d_{n-1}-1} + \ldots + \binom{n-3}{d_1-1, d_2-1, \ldots, d_{n-1}-2}$$

$$= \binom{n-2}{d_1-1, d_2-1, \ldots, d_{n-1}-1}$$

$$= \binom{n-2}{d_1-1, d_2-1, \ldots, d_{n-1}-1, 0}$$

$$= \binom{n-2}{d_1-1, d_2-1, \ldots, d_{n-1}-1, d_n-1}.$$ ■

Proof 3 of Theorem 2.1.

We can now count all trees on [n] by summing over all possible degree sequences:

$$\sum_{\substack{d_1, \ldots, d_n \geq 1 \\ d_1 + \ldots + d_n = 2n-2}} T(d_1, \ldots, d_n) = \sum_{\substack{d_1, \ldots, d_n \geq 1 \\ d_1 + \ldots + d_n = 2n-2}} \binom{n-2}{d_1-1, d_2-1, \ldots, d_n-1}$$

= the sum of all n-nomial coefficients of order $n - 2$

$= n^{n-2}$

(as $d_1 + \ldots + d_n = 2n - 2$, we have $d_1 - 1 + \ldots + d_n - 1 = 2n - 2 - n = n - 2$) [12]. ■

CHAPTER THREE

TREES ON THE COMPLETE BIPARTITE GRAPH

Given the results from chapter two, we are now ready to take the three proofs of Cayley's theorem and extend them to results on the number of trees on the complete bipartite graph.

Theorem 3.1

Let n, m $\in \mathbb{P}$. There are $m^{n-1}n^{m-1}$ spanning trees on the complete bipartite graph, $K_{n,m}$.

Proof 1 of Theorem 3.1.

For our first proof of the number of spanning trees on the complete bipartite graph we appeal to Theorem 2.2. Let A be the adjacency matrix of $K_{n,m}$. Then

$$A = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 & 1 & 1 & \cdots & 1 & 1 \\ 0 & 0 & \cdots & 0 & 0 & 1 & 1 & \cdots & 1 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 1 & 1 & \cdots & 1 & 1 \\ 0 & 0 & \cdots & 0 & 0 & 1 & 1 & \cdots & 1 & 1 \\ 1 & 1 & \cdots & 1 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 1 & \cdots & 1 & 1 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & \cdots & 1 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 1 & \cdots & 1 & 1 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}$$

$$\underbrace{\qquad\qquad\qquad}_{\text{n columns}} \quad \underbrace{\qquad\qquad\qquad}_{\text{m columns}}$$

and therefore

$$M = -A + \begin{bmatrix} m & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & m & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & m & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & m & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & n & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & n & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & n & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & n \end{bmatrix}$$

$$\underbrace{\qquad\qquad\qquad}_{\text{n columns}} \quad \underbrace{\qquad\qquad\qquad}_{\text{m columns}}$$

$$
= \begin{bmatrix}
m & 0 & \cdots & 0 & 0 & -1 & -1 & \cdots & -1 & -1 \\
0 & m & \cdots & 0 & 0 & -1 & -1 & \cdots & -1 & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & m & 0 & -1 & -1 & \cdots & -1 & -1 \\
0 & 0 & \cdots & 0 & m & -1 & -1 & \cdots & -1 & -1 \\
-1 & -1 & \cdots & -1 & -1 & n & 0 & \cdots & 0 & 0 \\
-1 & -1 & \cdots & -1 & -1 & 0 & n & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
-1 & -1 & \cdots & -1 & -1 & 0 & 0 & \cdots & n & 0 \\
-1 & -1 & \cdots & -1 & -1 & 0 & 0 & \cdots & 0 & n
\end{bmatrix}.
$$

$$
\underbrace{\qquad\qquad\qquad}_{\text{n columns}} \quad \underbrace{\qquad\qquad\qquad}_{\text{m columns}}
$$

Then by using the cofactor associated with the first row and column of M we get that

$$
M_{11} = (-1)^{1+1} \det \begin{bmatrix}
m & 0 & \cdots & 0 & 0 & -1 & -1 & \cdots & -1 & -1 \\
0 & m & \cdots & 0 & 0 & -1 & -1 & \cdots & -1 & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & m & 0 & -1 & -1 & \cdots & -1 & -1 \\
0 & 0 & \cdots & 0 & m & -1 & -1 & \cdots & -1 & -1 \\
-1 & -1 & \cdots & -1 & -1 & n & 0 & \cdots & 0 & 0 \\
-1 & -1 & \cdots & -1 & -1 & 0 & n & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
-1 & -1 & \cdots & -1 & -1 & 0 & 0 & \cdots & n & 0 \\
-1 & -1 & \cdots & -1 & -1 & 0 & 0 & \cdots & 0 & n
\end{bmatrix}
$$

$$
\underbrace{\qquad\qquad\qquad}_{\text{n - 1 columns}} \quad \underbrace{\qquad\qquad\qquad}_{\text{m columns}}
$$

$$
\overset{1}{=} \det \begin{bmatrix}
m & 0 & \cdots & 0 & 0 & -1 & -1 & \cdots & -1 & -1 \\
0 & m & \cdots & 0 & 0 & -1 & -1 & \cdots & -1 & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & m & 0 & -1 & -1 & \cdots & -1 & -1 \\
0 & 0 & \cdots & 0 & m & -1 & -1 & \cdots & -1 & -1 \\
0 & 0 & \cdots & 0 & 0 & n & 0 & \cdots & 0 & -n \\
0 & 0 & \cdots & 0 & 0 & 0 & n & \cdots & 0 & -n \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & n & -n \\
-1 & -1 & \cdots & -1 & -1 & 0 & 0 & \cdots & 0 & n
\end{bmatrix}
$$

n - 1 columns   m columns

$$
\overset{2}{=} \det \begin{bmatrix}
m & 0 & \cdots & 0 & 0 & -1 & -1 & \cdots & -1 & -1 \\
0 & m & \cdots & 0 & 0 & -1 & -1 & \cdots & -1 & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & m & 0 & -1 & -1 & \cdots & -1 & -1 \\
0 & 0 & \cdots & 0 & m & -1 & -1 & \cdots & -1 & -1 \\
0 & 0 & \cdots & 0 & 0 & n & 0 & \cdots & 0 & -n \\
0 & 0 & \cdots & 0 & 0 & 0 & n & \cdots & 0 & -n \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & n & -n \\
0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 1
\end{bmatrix} = m^{n-1}\, n^{m-1}.
$$

n - 1 columns   m columns

Thus the number of spanning trees of $K_{n,m}$ is $m^{n-1}\, n^{m-1}$. ■

[1] Subtract the last row from each of the rows n through n + m – 2.

[2] Add (1/m)(row 1 + … + row (n – 1)) + ((n – 1)/(mn))(rown n + … + row (n + m – 2)) to the last row.

Proof 2 of Theorem 3.1.

The next proof of theorem 3.1 is an adaptation of proof 2 of Theorem 2.1 using a bipartite Prüfer code. Although it was extended using only Proof 2 of Theorem 2.1, another statement of the process used in finding the bipartite Prüfer code can be found in Bodendiek and Henn **[4, pp. 341 – 342]**.

It suffices to show there is a one-to-one correspondence between the set of spanning trees of $K_{n,m}$ and the set of ordered pairs of sequences of integers (**a**, **b**), with **a** = $(a_1, a_2, \ldots, a_n)$ where $n+1 \leq a_i \leq n + m$ for all $1 \leq i \leq n - 1$ and $a_n = n + m$, and **b** = $(b_1, b_2, \ldots, b_{m-1})$ where $1 \leq b_j \leq n$ for all $1 \leq j \leq m - 1$, we call (**a**, **b**) the bipartite Prüfer code of the bipartite tree.

First we need to show that associated with each tree is such an ordered pair of sequences of integers. Let $K_{n,m}$ be the complete bipartite graph. Let T be a spanning tree of $K_{n,m}$. Find the bipartite Prüfer code in the following way. Remove the endpoint having the least label. If the removed endpoint is a left vertex, record to **a** the label of the adjacent vertex. If the endpoint is a right vertex, record to **b** the label of the adjacent vertex. Continue this process until a tree with only one vertex remains. Clearly $1 \leq b_j \leq n$ for all $b_j \in$ **b**. Also $n + 1 \leq a_i \leq n + m$ for all $a_i \in$ **a** and $a_n = n + m$ (since, prior to the last step, every tree has at least two endpoints, the smallest of which will never be n + m). These two sequences associated with T are the bipartite Prüfer code of T. Notice that **b** has m-1 terms and **a** has n terms, and the last element of **a** will always be n + m. In a more formal mathematical way, we are doing the following.

Let $T = (U \cup V, E)$ be a spanning tree (note that $U = [n]$ and $V = \{n + 1, \ldots, n + m\}$). Let $\varepsilon(T) = \{\text{endpoints of } T\}$. Define $u_1 = \min \varepsilon(T)$. Then there exists a unique c_1 such that $\{u_1, c_1\} \in E$. Define $U_1 = U \backslash \{u_1\}$, $V_1 = V \backslash \{u_1\}$, $E_1 = E \backslash \{\{u_1, c_1\}\}$, and let

$T_1 = (U_1 \cup V_1, E_1)$. Clearly T_1 is also a tree. Now let $u_2 = \min \varepsilon(T_1)$. Then there exits a unique c_2 such that $\{u_2, c_2\} \in E_1$. Define $U_2 = U\backslash\{u_1, u_2\}$, $V_2 = V\backslash\{u_1, u_2\}$, $E_2 = E\backslash\{\{u_1, c_1\}, \{u_2, c_2\}\} = E_1\backslash\{\{u_2, c_2\}\}$, and let $T_2 = (U_2 \cup V_2, E_2)$. Again, T_2 is clearly a tree. Continue repeating the process. In general, we have $U_{i-1} = U\backslash\{u_1, \ldots, u_{i-1}\}$, $V_{i-1} = V\backslash\{u_1, \ldots, u_{i-1}\}$, and $E_{i-1} = E\backslash\{\{u_j, a_j\} \mid 1 \leq j \leq i-1\}$. Then $T_{i-1} = (U_{i-1} \cup V_{i-1}, E_{i-1})$ and $u_i = \min \varepsilon(T_{i-1})$. At the final step, the tree $T_{n+m-2} = (U_{n+m-2} \cup V_{n+m-2}, E_{n+m-2})$ has two vertices joined by an edge $\{u_{n+m-1}, c_{n+m-1}\}$ where u_{n+m-1} is the smaller of the two vertices in $U_{n+m-2} \cup V_{n+m-2}$ and $c_{n+m-1} = n + m$ is the larger.

We have produced a sequence $(u_1, \ldots, u_{n+m-1})$ which we will call the minimum endpoint sequence of T. We have also produced the sequence $(c_1, \ldots, c_{n+m-1})$, which is the Prüfer code for T. Define the elements of **a** and **b** according to the following.

Let $a_1 = c_1$ if $u_1 \in [n]$, and let $b_1 = c_1$ if $u_1 > n$. Let $L(0) = R(0) = 1$, and let

$L(1) = \begin{cases} 2 & \text{if } u_1 \in [n] \\ 1 & \text{if } u_1 > n \end{cases}$ and $R(1) = \begin{cases} 1 & \text{if } u_1 \in [n] \\ 2 & \text{if } u_1 > n \end{cases}$. Then in general we have

$L(i) = \begin{cases} L(i-1)+1 & \text{if } u_i \in [n] \\ L(i-1) & \text{if } u_i > n \end{cases}$, and $R(i) = \begin{cases} R(i-1) & \text{if } u_i \in [n] \\ R(i-1)+1 & \text{if } u_i > n \end{cases}$ with $a_{L(i)} = c_{i+1}$ if $u_{i+1} \in [n]$ and $b_{R(i)} = c_{i+1}$ if $u_{i+1} > n$.

At each step, the indices only increase by at most one so we are systematically assigning each c_i to either **a** or **b** as appropriate. Each c_i is used only once, thus there are $n + m - 1$ steps involved and $n + m - 1$ elements in **(a, b)**. Thus we have produced the two sequences $\mathbf{a} = (a_1, \ldots, a_n)$ and $\mathbf{b} = (b_1, \ldots, b_{m-1})$ which form the bipartite Prüfer code of T. Note that $T = (U \cup V, \{\{u_i, c_i\} \mid 1 \leq i \leq n + m - 1\}$. By construction, the u_i are all distinct. Further, since every tree has at least two endpoints and each u_i is the smallest endpoint of a tree, no $u_i = n + m$. Thus the minimum endpoint sequence is a permutation of $[n + m - 1]$, and $a_n = n + m$.

Now we need to show that the bipartite Prüfer code is unique – i.e., given two bipartite spanning trees on U and V with the same bipartite Prüfer code, the trees are identical. Using the notation from above, assume that $S = (U \cup V, E)$ with minimum endpoint sequence $(s_1, \ldots, s_{n+m-1})$, that $T = (U \cup V, F)$ with minimum endpoint sequence $(t_1, \ldots, t_{n+m-1})$, and that S and T have the same bipartite Prüfer code $(\mathbf{a}, \mathbf{b})$ with $\mathbf{a} = (a_1, \ldots, a_n)$ and $\mathbf{b} = (b_1, \ldots, b_{m-1})$. Then as in Lemma 2.1, $\varepsilon(S) = (U \cup V)\backslash(\{a_1, \ldots, a_{n-1}\} \cup \{b_1, \ldots, b_{m-1}\}) = (U \cup V)\backslash(\{a_1, \ldots, a_n\} \cup \{b_1, \ldots, b_{m-1}\}) = \varepsilon(T)$. Thus by definition, $s_1 = \min \varepsilon(S) = \min \varepsilon(T) = t_1$. Then $S_1 = ((U \cup V)\backslash\{s_1\}, E\backslash\{\{s_1, c_1\}\})$ and $T_1 = (U \cup V\backslash\{t_1\}, F\backslash\{\{t_1, c_1\}\}$, where

$c_1 = \begin{cases} a_1 & \text{if } s_1 = t_1 > n \\ b_1 & \text{if } s_1 = t_1 \in [n] \end{cases}$. Then letting $L(0) = R(0) = 1$, $L(1) = \begin{cases} 2 & \text{if } s_1 = t_1 \in n \\ 1 & \text{if } s_1 = t_1 > n \end{cases}$ and

$R(1) = \begin{cases} 1 & \text{if } s_1 = t_1 \in [n] \\ 2 & \text{if } s_1 = t_1 > n \end{cases}$ we have $\varepsilon(S_1) = (U \cup V)\backslash(\{s_1\} \cup \{a_{L(1)}, \ldots, a_{n-1}\} \cup$

$\{b_{R(1)}, \ldots, b_{m-1}\})$ and $\varepsilon(T_1) = (U \cup V)\backslash(\{t_1\} \cup \{a_{L(1)}, \ldots, a_{n-1}\} \cup \{b_{R(1)}, \ldots, b_{m-1}\})$, so $\varepsilon(S_1) = \varepsilon(T_1)$, and therefore $s_2 = \min \varepsilon(S_1) = \min \varepsilon(T_1) = t_2$. In general we get

$L(i-1) = \begin{cases} L(i-1)+1 & \text{if } s_{i-1} = t_{i-1} \in n \\ L(i-1) & \text{if } s_{i-1} = t_{i-1} > n \end{cases}$, $R(i-1) = \begin{cases} R(i-1) & \text{if } s_{i-1} = t_{i-1} \in [n] \\ R(i-1)+1 & \text{if } s_{i-1} = t_{i-1} > n \end{cases}$,

with $S_{i-1} = ((U \cup V)\backslash\{s_1, \ldots, s_{i-1}\}, E\backslash\{\{s_j, c_j\} \mid 1 \le j \le i-1\})$ and $T_{i-1} = ((U \cup V)\backslash\{t_1, \ldots, t_{i-1}\}, E\backslash\{\{t_j, c_j\} \mid 1 \le j \le i-1\})$, where

$c_j = \begin{cases} a_{L(i-2)} & \text{if } s_{i-1} = t_{i-1} > n \\ b_{R(i-2)} & \text{if } s_{i-1} = t_{i-1} \in [n] \end{cases}$. Then again as in Lemma 2.1,

$\varepsilon(S_{i-1}) = ((U \cup V)\backslash\{s_1, \ldots, s_{i-1}\} \cup \{a_{L(i-1)}, \ldots, a_{n-1}\} \cup \{b_{R(i-1)}, \ldots, b_{m-1}\}) =$
$= ((U \cup V)\backslash\{t_1, \ldots, t_{i-1}\} \cup \{a_{L(i-1)}, \ldots, a_{n-1}\} \cup \{b_{R(i-1)}, \ldots, b_{m-1}\}) = \varepsilon(T_{i-1})$. Therefore $s_i = \min \varepsilon(S_{i-1}) = \min \varepsilon(T_{i-1}) = t_i$ for all $1 \le i \le n+m-1$. Then $(s_1, \ldots, s_{n+m-1}) = (t_1, \ldots, t_{n+m-1})$ and therefore $S = T$.

Note that $s_i = t_i = \min((U \cup V)\backslash(\{s_1 = t_1, \ldots, s_{i-1} = t_{i-1}\} \cup$
$\{a_{L(i-1)}, \ldots, a_n, b_{R(i-1)}, \ldots, b_{m-1}\}))$. (1)

This constructively defines the tree from (**a**, **b**), so if we know the bipartite Prüfer code, we know T.

We have shown how to find the bipartite Prüfer code of a tree and that trees with the same bipartite Prüfer code are equal. Now we need to show that given an ordered pair of sequences of integers (**a**, **b**), with **a** = $(a_1, a_2, \ldots, a_n)$ where $n + 1 \le a_i \le n + m$ for all $1 \le i \le n - 1$ and $a_n = n + m$, and **b** = $(b_1, b_2, \ldots, b_{m-1})$ where $1 \le b_j \le n$ for all $1 \le j \le m - 1$, there is a bipartite tree (which is unique from above) with (**a**, **b**) as its bipartite Prüfer code. Let (**a**, **b**) be such an ordered pair. Define u_i recursively by the following.

Let $u_1 = \min([n + m]\backslash\{a_1, \ldots, a_n, b_1, \ldots, b_{m-1}\})$. Define $L(0) = R(0) = 1$, and $L(1) = \begin{cases} 2 & \text{if } u_1 \in [n] \\ 1 & \text{if } u_1 > n \end{cases}$. Define $R(1) = \begin{cases} 1 & \text{if } u_1 \in [n] \\ 2 & \text{if } u_1 > n \end{cases}$. Then in general,

$u_i = \min([n + m]\backslash(\{u_1, \ldots, u_{i-1}\} \cup \{a_{L(i-1)}, \ldots, a_n, b_{R(i-1)}, \ldots, b_{m-1}\}))$ for $i = 2, \ldots, n + m - 1$, with $L(i) = |\{u_1, \ldots, u_i\} \cap [n]| + 1$ and $R(i) = |\{u_1, \ldots, u_i\} \cap \{n + 1, \ldots, n + m\}| + 1$. We could equivalently define L(i) and R(i) as follows:

$$L(i) = \begin{cases} L(i-1)+1 & \text{if } u_i \in [n] \\ L(i-1) & \text{if } u_i > n \end{cases} \text{ and } R(i) = \begin{cases} R(i-1) & \text{if } u_i \in [n] \\ R(i-1)+1 & \text{if } u_i > n \end{cases}.$$

Note that $L(i) + R(i) = i + 2$. This is clearly true for $i = 1$. Then assuming $L(i - 1) + R(i - 1) = (i - 1) + 2 = i + 1$, we have that $L(i) + R(i) = L(i - 1) + R(i - 1) + 1 = i + 1 + 1 = i + 2$. Then $[n + m]\backslash(\{u_1, \ldots, u_{i-1}\} \cup \{a_{L(i-1)}, \ldots, a_n, b_{R(i-1)}, \ldots, b_{m-1}\})$ is never empty as there are at most $n + m - 1$ elements in $\{u_1, \ldots, u_{i-1}\} \cup \{a_{L(i-1)}, \ldots, a_n, b_{R(i-1)}, \ldots, b_{m-1}\}$ (by counting indices, we have at most

$(i - 1) + n + (m - 1) - [L(i - 1) - 1] - [R(i - 1) - 1]$

$= (i - 1) + n + m - 1 + 2 - (L(i - 1) + R(i - 1))$

$= (i - 1) + n + m + 1 - ((i - 1) + 2)$

$= i - 1 + n + m - i$

$= n + m - 1$ elements in the union)

and $n + m$ elements in $U \cup V = [n + m]$. Once again, this definition compels $(u_1, \ldots, u_{n+m-1})$ to be a permutation of $[n + m - 1]$. Define a graph $T = (U \cup V, E)$ where $E = \{\{u_i, c_i\} \mid 1 \leq i \leq n + m\}$ with $c_i = \begin{cases} a_{L(i-1)} & \text{if } u_i > [n] \\ b_{R(i-1)} & \text{if } u_i \in n \end{cases}$. Also, for $1 \leq i \leq n + m$, let $U_i = U \setminus \{u_1, \ldots, u_{i-1}\}$, $V_i = V \setminus \{u_1, \ldots, u_{i-1}\}$, $E_i = E \setminus \{\{u_i, c_i\} \mid 1 \leq j < i\}$ and $T_i = (U_i \cup V_i, E_i)$. We claim that the resulting graphs T and T_i are trees. We need only to show that u_i is an endpoint of the graph T_i and that no other vertex with smaller label is an endpoint. Since u_i is adjacent to c_i in T_i and $c_i = u_j$ for some $j > i$ (that is, c_i is an endpoint of a later T_j), we can trace a path from each u_i to $a_n = n + m$, showing connectedness of T and each T_i.

Note that $\{u_i, c_i\} \in E_i$. Therefore u_i has a neighbor in T_i, namely c_i. Also u_i cannot be adjacent to any other vertex of T_i, since if it were, then for some j, $\{c_j, u_j\}$ is also an edge of T_i that includes u_i. Then $j > i$ as u_j is a vertex of T_i. Also, we have that either $u_i = u_j$ or $u_i = c_j$. But $u_i \neq u_j$ since $u_i \notin T_j$, and $u_i \neq c_j$ since c_j is equal to a later $a_{L(j-1)}$ or $b_{R(j-1)}$ and thus in $\{a_{L(i-1)}, \ldots, a_n, b_{R(i-1)}, \ldots, b_{m-1}\}$. Thus u_i is an endpoint of T_i. Hence we have that T and all T_i's are trees. Since T is a tree, it has a bipartite Prüfer code. Note that we have just defined each u_i as it was defined in (1). Therefore u_i is the endpoint of smallest label in T_i.

We have shown that there is a one-to-one correspondence between the set of spanning trees on $K_{n,m}$ and the set of ordered sequences of integers $\mathbf{a} = (a_1, a_2, \ldots, a_n)$ where $n + 1 \leq a_i \leq n + m$ for all $1 \leq i \leq n - 1$ and $a_n = n + m$, and $\mathbf{b} = (b_1, b_2, \ldots, b_{m-1})$ where $1 \leq b_j \leq n$ for all $1 \leq j \leq m - 1$. Hence, as the total number of such sequences is $m^{n-1}n^{m-1}$, we have that the total number of spanning trees of $K_{n,m}$ is $m^{n-1}n^{m-1}$. ■

For the last proof of theorem 3.1, we need the following lemma, which can be found in Pak [11].

Lemma 3.1

Given a sequence of positive integers $(d_1, d_2, \ldots, d_n; d_{n+1}, d_{n+2}, \ldots, d_{n+m})$ with $\sum_{i=1}^{n} d_i = \sum_{i=n+1}^{n+m} d_i = m + n - 1$, the number of spanning trees of $K_{n,m}$ such that vertex i has degree d_i, denoted $T(d_1, d_2, \ldots, d_n; d_{n+1}, d_{n+2}, \ldots, d_{n+m})$, is

$$\binom{m-1}{d_1-1, d_2-1, \ldots, d_n-1}\binom{n-1}{d_{n+1}-1, d_{n+2}-1, \ldots, d_{n+m}-1}.$$

Proof.

This proof can be shown by induction on n + m.

For n + m = 2, we have that n = m = 1. So we need positive integers d_1 and d_2 such that $d_1 + d_2 = 2$. Thus $d_1 = d_2 = 1$ and therefore $\binom{1-1}{d_1-1}\binom{1-1}{d_2-1} = \binom{0}{0}\binom{0}{0} = 1$ if $d_1 = d_2 = 1$. The theorem holds in this case since there is only one spanning tree on $K_{1,1}$.

Let $n + m \geq 3$. Suppose the theorem holds for all sequences of positive integers $(d_1, d_2, \ldots, d_p; d_{p+1}, d_{p+2}, \ldots, d_{p+q})$ with $\sum_{i=1}^{p} d_i = \sum_{i=p+1}^{p+q} d_i = p + q - 1$ for $p \leq n$ and $q \leq m$, with at least one of the inequalities strict. Suppose we have a sequence of positive integers $(d_1, d_2, \ldots, d_n; d_{n+1}, d_{n+2}, \ldots, d_{n+m})$ with $\sum_{i=1}^{n+m} d_i = 2(n + m - 1)$.

Using the same argument that appears in the proof of Lemma 2.1, we may assume that $d_1 \geq \ldots \geq d_n = 1$. Then

$T(d_1, d_2, \ldots, 1; d_{n+1}, d_{n+2}, \ldots, d_{n+m}) =$ (1)

$= T(d_1, d_2, \ldots, 1; d_{n+1} - 1, d_{n+2}, \ldots, d_{n+m}) + T(d_1, d_2, \ldots, 1; d_{n+1}, d_{n+2} - 1, \ldots, d_{n+m})$

$+ \ldots + T(d_1, d_2, \ldots, 1; d_{n+1}, d_{n+2}, \ldots, d_{n+m} - 1),$

by categorizing the trees enumerated by $T(d_1, d_2, \ldots, 1; d_{n+1}, d_{n+2}, \ldots, d_{n+m})$ according to the vertex adjacent to n.

Applying the induction hypothesis to the RHS of (1) and invoking the multinomial recurrence gives us

$T(d_1, d_2, \ldots, 1; d_{n+1}, d_{n+2}, \ldots, d_{n+m}) =$

$$= \binom{m-1}{d_1-1, d_2-1, \ldots, d_{n-1}-1}\binom{n-2}{d_{n+1}-2, d_{n+2}-1, \ldots, d_{n+m}-1} +$$

$$\binom{m-1}{d_1-1, d_2-1, \ldots, d_{n-1}-1}\binom{n-2}{d_{n+1}-1, d_{n+2}-2, \ldots, d_{n+m}-1} + \ldots +$$

$$\binom{m-1}{d_1-1, d_2-1, \ldots, d_{n-1}-1}\binom{n-2}{d_{n+1}-1, d_{n+2}-1, \ldots, d_{n+m}-2}$$

$$= \binom{m-1}{d_1-1, \ldots, d_{n-1}-1}\sum_{k=1}^{m}\binom{n-2}{d_{n+1}-1, \ldots, d_{n+k}-2, \ldots, d_{n+m}-1}$$

$$= \binom{m-1}{d_1-1, \ldots, d_{n-1}-1, 0}\sum_{k=1}^{m}\binom{n-2}{d_{n+1}-1, \ldots, d_{n+k}-2, \ldots, d_{n+m}-1}$$

$$= \binom{m-1}{d_1-1, \ldots, d_n-1}\sum_{k=1}^{m}\binom{n-2}{d_{n+1}-1, \ldots, d_{n+k}-2, \ldots, d_{n+m}-1}$$

$$= \binom{m-1}{d_1-1, d_2-1, \ldots, d_n-1}\binom{n-1}{d_{n+1}-1, d_{n+2}-1, \ldots, d_{n+m}-1}. \qquad \blacksquare$$

Proof 3 of theorem 3.1.

We can now count all spanning trees of $K_{n,m}$, by summing over all possible degree sequences. Note that in the complete bipartite graph, as all edges begin on one side and end on the other, the sum of the degrees of the vertices on the left side equals the sum of the degrees of the vertices on the right side.

$$\sum_{\substack{\text{all possible}\\\text{degree sequences}}} T(d_1,\dots,d_n;d_{n+1},\dots,d_{n+m})$$

$$= \sum_{\substack{\text{all possible}\\\text{degree sequences}}} \binom{m-1}{d_1-1,\dots,d_n-1}\binom{n-1}{d_{n+1}-1,\dots,d_{n+m}-1}$$

$$= \sum_{\substack{d_i \geq 1\\ d_1+\dots+d_n=m+n-1\\ d_{n+1}+\dots+d_{n+m}=m+n-1}} \binom{m-1}{d_1-1,\dots,d_n-1}\binom{n-1}{d_{n+1}-1,\dots,d_{n+m}-1}$$

$$= \sum_{\substack{d_i \geq 1\\ d_1+\dots+d_n=m+n-1}} \binom{m-1}{d_1-1,\dots,d_n-1} \sum_{\substack{d_i \geq 1\\ d_{n+1}+\dots+d_{n+m}=m+n-1}} \binom{n-1}{d_{n+1}-1,\dots,d_{n+m}-1}$$

= (the sum of all n-nomial coefficients of order $m-1$) • (the sum of all m-nomial coefficients of order $n-1$)

$= n^{m-1}m^{n-1}$

(as $d_1 + \dots + d_n = m + n - 1$, we have that $d_1 - 1 + \dots + d_n - 1 = m + n - 1 - n = m - 1$ and as $d_{n+1} + \dots + d_{n+m} = m + n - 1$, we have that $d_{n+1} - 1 + \dots + d_{n+m} - 1 = m + n - 1 - m = n - 1$). ■

CHAPTER FOUR
TREES ON THE COMPLETE TRIPARTITE GRAPH

Now that we have the results for the number of trees on [n] and the number of spanning trees on the complete bipartite graph, we are ready to take the first two proofs and extend them to a result on the number of spanning trees on the complete tripartite graph.

Theorem 4.1

Let p, q, r, $\in \mathbb{P}$, and let $n = p + q + r$. There are $n(n-p)^{p-1}(n-q)^{q-1}(n-r)^{r-1}$ spanning trees on the complete tripartite graph, $K_{p,q,r}$.

Proof 1 of Theorem 4.1.

For our first proof, which was aided by Pak **[11]**, of the number of spanning trees on complete tripartite graph, we once again appeal to Theorem 2.2. Let A be the adjacency matrix of $K_{p,q,r}$. Then

$$
A = \underbrace{\begin{bmatrix}
0 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 \\
0 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 \\
1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 \\
1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 \\
1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\
1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0
\end{bmatrix}}_{\text{p columns} \quad \text{q columns} \quad \text{r columns}}
$$

and therefore

$$
M = -A + \underbrace{\begin{bmatrix}
q+r & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\
0 & q+r & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & q+r & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & p+r & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & 0 & p+r & \cdots & 0 & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & p+r & 0 & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & p+q & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & p+q & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & p+q
\end{bmatrix}}_{\text{p columns} \quad \text{q columns} \quad \text{r columns}}
$$

$$
= \begin{bmatrix}
q+r & 0 & \cdots & 0 & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
0 & q+r & \cdots & 0 & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & q+r & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
-1 & -1 & \cdots & -1 & p+r & 0 & \cdots & 0 & -1 & -1 & \cdots & -1 \\
-1 & -1 & \cdots & -1 & 0 & p+r & \cdots & 0 & -1 & -1 & \cdots & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
-1 & -1 & \cdots & -1 & 0 & 0 & \cdots & p+r & -1 & -1 & \cdots & -1 \\
-1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 & p+q & 0 & \cdots & 0 \\
-1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 & 0 & p+q & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
-1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 & 0 & 0 & \cdots & p+q
\end{bmatrix}
$$

$\underbrace{\qquad\qquad}_{\text{p columns}}$ $\underbrace{\qquad\qquad}_{\text{q columns}}$ $\underbrace{\qquad\qquad}_{\text{r columns}}$

Then by using the cofactor associated with the last row and last column of M we get that

$M_{n,n} = (-1)^{2n} \bullet \det$

$$
\begin{bmatrix}
q+r & 0 & \cdots & 0 & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
0 & q+r & \cdots & 0 & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & q+r & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
-1 & -1 & \cdots & -1 & p+r & 0 & \cdots & 0 & -1 & -1 & \cdots & -1 \\
-1 & -1 & \cdots & -1 & 0 & p+r & \cdots & 0 & -1 & -1 & \cdots & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
-1 & -1 & \cdots & -1 & 0 & 0 & \cdots & p+r & -1 & -1 & \cdots & -1 \\
-1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 & p+q & 0 & \cdots & 0 \\
-1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 & 0 & p+q & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
-1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 & 0 & 0 & \cdots & p+q
\end{bmatrix}
$$

$\underbrace{\qquad\qquad}_{\text{p columns}}$ $\underbrace{\qquad\qquad}_{\text{q columns}}$ $\underbrace{\qquad\qquad}_{\text{r - 1 columns}}$

$$
=^{1} \det \begin{bmatrix}
1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\
0 & q+r & \cdots & 0 & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & q+r & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
-1 & -1 & \cdots & -1 & p+r & 0 & \cdots & 0 & -1 & -1 & \cdots & -1 \\
-1 & -1 & \cdots & -1 & 0 & p+r & \cdots & 0 & -1 & -1 & \cdots & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
-1 & -1 & \cdots & -1 & 0 & 0 & \cdots & p+r & -1 & -1 & \cdots & -1 \\
-1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 & p+q & 0 & \cdots & 0 \\
-1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 & 0 & p+q & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
-1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 & 0 & 0 & \cdots & p+q
\end{bmatrix}
$$

p columns q columns r - 1 columns

$$
=^{2} \det \begin{bmatrix}
1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\
0 & q+r & \cdots & 0 & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & q+r & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
0 & 0 & \cdots & 0 & p+r+1 & 1 & \cdots & 1 & -1 & -1 & \cdots & -1 \\
0 & 0 & \cdots & 0 & 1 & p+r+1 & \cdots & 1 & -1 & -1 & \cdots & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & 1 & 1 & \cdots & p+r+1 & -1 & -1 & \cdots & -1 \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & p+q & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & p+q & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & p+q
\end{bmatrix}
$$

p columns q columns r - 1 columns

[1] Replace row 1 in the previous matrix by itself plus rows 2 through $p + q + r - 1$.

[2] Replace each of the rows $p + 1$ through $p + q + r - 1$ in the previous matrix by itself plus row 1.

$$
=^{1} \det\begin{bmatrix}
1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\
0 & q+r & \cdots & 0 & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & q+r & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
0 & 0 & \cdots & 0 & p+r+q & p+r+q & \cdots & p+r+q & -q & -q & \cdots & -q \\
0 & 0 & \cdots & 0 & 1 & p+r+1 & \cdots & 1 & -1 & -1 & \cdots & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & 1 & 1 & \cdots & p+r+1 & -1 & -1 & \cdots & -1 \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & p+q & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & p+q & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & p+q
\end{bmatrix}
$$

p columns   q columns   r - 1 columns

$$
=^{2} \det\begin{bmatrix}
1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\
0 & q+r & \cdots & 0 & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & q+r & -1 & -1 & \cdots & -1 & -1 & -1 & \cdots & -1 \\
0 & 0 & \cdots & 0 & p+r+q & p+r+q & \cdots & p+r+q & -q & -q & \cdots & -q \\
0 & 0 & \cdots & 0 & 0 & p+r & \cdots & 0 & c & c & \cdots & c \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & p+r & c & c & \cdots & c \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & p+q & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & p+q & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & p+q
\end{bmatrix}
$$

p columns   q columns   r - 1 columns

$= (p + q + r)(q + r)^{p-1}(p + r)^{q-1}(p + q)^{r-1}$, where $c = (-p - r)/(p + q + r)$.

$= n(n - p)^{p-1}(n - q)^{q-1}(n - r)^{r-1}$

Thus the number of spanning trees on $K_{p,q,r}$ is $n(n - p)^{p-1}(n - q)^{q-1}(n - r)^{r-1}$. ∎

---

[1] Replace row p + 1 in the previous matrix by itself plus each of the rows p + 2 through p + q.

[2] Replace each of rows p + 2 through p + q in the previous matrix by itself plus (-1)/(p + r + q) times row p + 1.

Proof 2 of Theorem 4.1.

The next proof of Theorem 4.1 is an adaptation of proof 2 of Theorem 2.1 using a tripartite Prüfer code.

It suffices to show there is a one-to-one correspondence between the set of trees on the complete tripartite graph $\mathbf{K_{p,q,r}}$, and the set of quadruples of sequences of integers ($\mathbf{a_1}$, $\mathbf{a_2}$, $\mathbf{a_3}$, $\mathbf{a}$), with $\mathbf{a_1} = (a_{11}, a_{12}, \ldots, a_{1(p-1)})$ where $p + 1 \leq a_{1i} \leq n$ for all $1 \leq i \leq p - 1$, $\mathbf{a_2} = (a_{21}, a_{22}, \ldots, a_{2(q-1)})$ where $1 \leq a_{2j} \leq p$ or $(p + q + 1) \leq b_j \leq n$ for all $1 \leq j \leq q - 1$, $\mathbf{a_3} = (a_{31}, a_{32}, \ldots, a_{3(r-1)})$ where $1 \leq a_{3k} \leq p + q$ for all $1 \leq k \leq r - 1$, and $\mathbf{a} = (a_1, a_2)$ where $1 \leq a_1 \leq n$ and $a_2 = n$. We call ($\mathbf{a_1}$, $\mathbf{a_2}$, $\mathbf{a_3}$, $\mathbf{a}$) the tripartite Prüfer code.

First we need to show that associated with each spanning tree there is such a quadruple of ordered sequences of integers. Let $\mathbf{K_{p,q,r}}$ be the complete tripartite graph. Let T be a spanning tree of $\mathbf{K_{p,q,r}}$. Find the tripartite Prüfer code in the following way. Remove the endpoint having the least label. If the endpoint is the sole remaining vertex in its vertex set, record to **a** the label of the adjacent vertex. Otherwise, if the removed endpoint is a left vertex, record to $\mathbf{a_1}$ the label of the adjacent vertex. If the endpoint is a middle vertex, record to $\mathbf{a_2}$ the label of the adjacent vertex. If the endpoint is a right vertex, record to $\mathbf{a_3}$ the label of the adjacent vertex. Continue this process until a tree with only one vertex remains. Clearly $p + 1 \leq a_{1i} \leq n$ for all $a_i \in \mathbf{a_1}$, $1 \leq a_{2j} \leq p$ or $(p + q + 1) \leq a_{2j} \leq n$ for all $a_{2j} \in \mathbf{a_2}$, $1 \leq a_{3k} \leq p + q$ for all $a_{3k} \in \mathbf{a_3}$, $1 \leq a_1 \leq n$ and $a_2 = n$ as by the procedure n will be the last vertex remaining, since after each removal we still have a tree which has at least two endpoints, the smaller of which will never be n. These four sequences associated with T define the tripartite Prüfer code. More formally, we are doing the following.

Let $T = (V_1 \cup V_2 \cup V_3, E)$ be a spanning tree, where $V_1 = [p]$, $V_2 = \{p+1, \ldots, p+q\}$, and $V_3 = \{p+q+1, \ldots, n\}$). Let $\varepsilon(T) = \{\text{endpoints of } T\}$. Define $u_1 = \min \varepsilon(T)$. Then there exists a unique k_1 such that $\{u_1, k_1\} \in E$. Define $V_{1,1} = V_1 \backslash \{u_1\}$, $V_{2,1} = V_2 \backslash \{u_1\}$, $V_{3,1} = V_3 \backslash \{u_1\}$, $E_1 = E \backslash \{\{u_1, k_1\}\}$, and let $T_1 = (V_{1,1} \cup V_{2,1} \cup V_{3,1}, E_1)$. Clearly T_1 is also a tree. Now let $u_2 = \min \varepsilon(T_1)$. Then there exits a unique k_2 such that $\{u_2, k_2\} \in E_1$. Define $V_{1,2} = V_1 \backslash \{u_1, u_2\}$, $V_{2,2} = V_2 \backslash \{u_1, u_2\}$, $V_{3,2} = V_3 \backslash \{u_1, u_2\}$, $E_2 = E \backslash \{\{u_1, k_1\}, \{u_2, k_2\}\} = E_1 \backslash \{\{u_2, k_2\}\}$, and let $T_2 = (V_{1,2} \cup V_{2,2} \cup V_{3,2}, E_2)$. Again, T_2 is clearly a tree. Continue repeating the process. In general, $V_{j,i-1} = V_1 \backslash \{u_1, \ldots, u_{i-1}\}$ for $j \in [3]$, and $E_{i-1} = E \backslash \{\{u_j, k_j\} \mid 1 \le j \le i-1\}$. Then $T_{i-1} = (V_{1,i-1} \cup V_{2,i-1} \cup V_{3,i-1}, E_{i-1})$, and $u_i = \min \varepsilon(T_{i-1})$. At the final step, the tree $T_{n-2} = (V_{1,n-2} \cup V_{2,n-2} \cup V_{3,n-2}, E_{n-2})$ has two vertices joined by an edge $\{u_{n-2}, k_{n-2}\}$ where u_{n-2} is the smaller of the two vertices in $V_{1,n-2} \cup V_{2,n-2} \cup V_{3,n-2}$, and $k_{n-2} = n$ is the larger.

We have produced a sequence $(u_1, \ldots, u_{n-1})$ which we will call the minimum endpoint sequence of T. We have also produced the sequence $(k_1, \ldots, k_{n-1})$, which is the Prüfer code. Define the elements of $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{a}_3$, and $\mathbf{a}$ according to the following.

First, $u_1 \in V_h$ for some h, then $a_1 = k_1$ if $V_h \backslash \{u_1\} = \varnothing$ and $a_{h1} = k_1$ otherwise. Then for $j \in [3]$, let $f_j(1) = \begin{cases} 2 & \text{if } j = h \\ 1 & \text{otherwise} \end{cases}$ and $f_4(1) = \begin{cases} 1 & \text{if } V_h \backslash \{u_1\} \ne \varnothing \\ 2 & \text{otherwise} \end{cases}$. Next, $u_2 \in V_h$ for some h, then $a_{f_4(1)} = k_2$ if $V_h \backslash \{u_1, u_2\} = \varnothing$ and $a_{hf_h(1)} = k_2$ otherwise. Then for $j \in [3]$, let $f_j(2) = \begin{cases} f_j(1) + 1 & \text{if } j = h \\ f_j(1) & \text{otherwise} \end{cases}$, and $f_4(2) = \begin{cases} 1 & \text{if } V_h \backslash \{u_1, u_2\} \ne \varnothing \\ 2 & \text{otherwise} \end{cases}$.

Then in general for $u_i \in V_h$, let $f_j(i-1) = \begin{cases} f_j(i-2)+1 & \text{if } j = h \\ f_j(i-2) & \text{otherwise} \end{cases}$ for $j \in [3]$, and

$f_4(i-1) = \begin{cases} 1 & \text{if } V_h \backslash \{u_1, ..., u_{i-1}\} \neq \varnothing \\ 2 & \text{otherwise} \end{cases}$. Define $a_{f_4(i-1)} = k_i$ if $V_h \backslash \{u_1, \ldots, u_i\} = \varnothing$,

and $a_{hf_h(i-1)} = k_i$ otherwise.

Note that at each step, the indices only increase by at most one so we are assigning each k_i to $\mathbf{a_1}$, $\mathbf{a_2}$, $\mathbf{a_3}$, or $\mathbf{a}$ as required. Therefore as each k_i is used only once, there are $n-1$ steps involved and $n-1$ elements in $(\mathbf{a_1}, \mathbf{a_2}, \mathbf{a_3}, \mathbf{a})$. Thus we have produced the four sequences $\mathbf{a_1} = (a_{11}, a_{12}, \ldots, a_{1(p-1)})$, $\mathbf{a_2} = (a_{21}, a_{22}, \ldots, a_{2(q-1)})$, $\mathbf{a_3} = (a_{31}, a_{32}, \ldots, a_{3(r-1)})$, and $\mathbf{a} = (a_1, a_2)$ which form tripartite Prüfer code of T. Note that $T = (V_1 \cup V_2 \cup V_3, \{\{u_i, k_i\} \mid 1 \leq i \leq n-1\}$. By construction, the u_i are all distinct. Further, since every tree has at least two endpoints and each u_i is the smallest of a tree, no $u_i = n$. Thus the minimum endpoint sequence is a permutation of $[n-1]$, and $a_2 = n$. Notice that $V_3 \backslash \{u_1, \ldots, u_i\}$ is never empty since $u_i \neq n$ for all i. Therefore $\mathbf{a}$ needs only two terms.

Now we need to show that the bipartite Prüfer code is unique – i.e., given two trees on V_1, V_2, and V_3 with the same bipartite Prüfer code, the trees are identical. Using the notation from above, assume that $S = (V_1 \cup V_2 \cup V_3, E)$ with minimum endpoint sequence $(s_1, \ldots, s_{n-1})$, that $T = (V_1 \cup V_2 \cup V_3, F)$ with minimum endpoint sequence $(t_1, \ldots, t_{n-1})$, and that S and T have the same tripartite Prüfer code

$\mathbf{a_1} = (a_{11}, a_{12}, \ldots, a_{1(p-1)})$, $\mathbf{a_2} = (a_{21}, a_{22}, \ldots, a_{2(q-1)})$, $\mathbf{a_3} = (a_{31}, a_{32}, \ldots, a_{3(r-1)})$, and $\mathbf{a} = (a_1, a_2)$. Then by using the same reasoning as in Lemma 2.1,
$\varepsilon(S) = (V_1 \cup V_2 \cup V_3) \backslash (\{a_{11}, a_{12}, \ldots, a_{1(p-1)}, a_{21}, a_{22}, \ldots, a_{2(q-1)}, a_{31}, a_{32}, \ldots, a_{3(r-1)}, a_1\}) = \varepsilon(T)$. Thus by definition, $s_1 = \min \varepsilon(S) = \min \varepsilon(T) = t_1$. Then

$S_1 = ((V_1 \cup V_2 \cup V_3)\backslash\{s_1\}, E\backslash\{\{s_1, k_1\}\}$ and $T_1 = (V_1 \cup V_2 \cup V_3\backslash\{t_1\}, F\backslash\{\{t_1, k_1\}\}$, for

$s_1 = t_1 \in V_h$, where $k_1 = \begin{cases} a_1 & \text{if } V_h\backslash\{u_1\} = \varnothing \\ a_{h1} & \text{otherwise} \end{cases}$. Then for $j \in [3]$, letting

$f_j(1) = \begin{cases} 2 & \text{if } j = h \\ 1 & \text{otherwise} \end{cases}$, and $f_4(1) = \begin{cases} 1 & \text{if } V_h\backslash\{u_1\} \neq \varnothing \\ 2 & \text{otherwise} \end{cases}$ we have

$\varepsilon(S_1) = (V_1 \cup V_2 \cup V_3)\backslash(\{s_1\} \cup \{a_{1f_1(1)}, a_{12}, \ldots, a_{1(p-1)}, a_{2f_2(1)}, a_{22}, \ldots, a_{2(q-1)},$

$a_{3f_3(1)}, a_{32}, \ldots, a_{3(r-1)}, a_1\}) = \varepsilon(T_1)$. Thus $\varepsilon(S_1) = \varepsilon(T_1)$, and therefore $t_2 = s_2$.

Continuing we get $f_j(i-1) = \begin{cases} f(i-2) + 1 & \text{if } j = h \\ f(i-2) & \text{otherwise} \end{cases}$ for $j \in [3]$, and

$f_4(i-1) = \begin{cases} 1 & \text{if } V_h\backslash\{u_1, \ldots, u_{i-1}\} \neq \varnothing \\ 2 & \text{otherwise} \end{cases}$. Then

$S_{i-1} = ((V_1 \cup V_2 \cup V_3), E\backslash\{\{s_j, k_j\} \mid 1 \le j \le i-1\})$ and

$T_{i-1} = ((V_1 \cup V_2 \cup V_3), E\backslash\{\{t_j, k_j\} \mid 1 \le j \le i-1\})$. Therefore $\varepsilon(S_{i-1}) = (V_1 \cup V_2 \cup V_3)\backslash(\{s_1, \ldots, s_{i-1}\} \cup \{a_{1f_1(i-1)}, a_{12}, \ldots, a_{1(p-1)}, a_{2f_2(i-1)}, a_{22}, \ldots, a_{2(q-1)},$

$a_{3f_3(i-1)}, a_{32}, \ldots, a_{3(r-1)}, a_1\}) = \varepsilon(T_{i-1})$ and hence $s_i = \min \varepsilon(S_{i-1}) = \min \varepsilon(T_{i-1}) = t_i$ for all $1 \le i \le n-1$. Then $(s_1, \ldots, s_{n-1}) = (t_1, \ldots, t_{n-1})$ and therefore $S = T$. Note that

$$s_i = t_i = \min((V_1 \cup V_2 \cup V_3)\backslash(\{s_1 = t_1, \ldots, s_{i-1} = t_{i-1}\} \cup \{a_{1f_1(i-1)}, a_{12}, \ldots, a_{1(p-1)}, a_{2f_2(i-1)}, a_{22}, \ldots, a_{2(q-1)}, a_{3f_3(Ii-1)}, a_{32}, \ldots, a_{3(r-1)}, a_1, a_2\})). \quad (1)$$

This constructively defines the tree from $(\mathbf{a_1}, \mathbf{a_2}, \mathbf{a_3}, \mathbf{a})$, so if we know the tripartite Prüfer code, we know T.

We have shown how to find the tripartite Prüfer code of a tree and that trees with the same tripartite Prüfer code are equal. Now we to show that given a quadruple of sequences of integers $(\mathbf{a_1}, \mathbf{a_2}, \mathbf{a_3}, \mathbf{a})$, with with $\mathbf{a_1} = (a_{11}, a_{12}, \ldots, a_{1(p-1)})$ where $p + 1 \le a_{1i} \le n$ for all $1 \le i \le p-1$, $\mathbf{a_2} = (a_{21}, a_{22}, \ldots, a_{2(q-1)})$ where $1 \le a_{2j} \le p$ or

$(p + q + 1) \le b_j \le n$ for all $1 \le j \le q - 1$, $\mathbf{a_3} = (a_{31}, a_{32}, \ldots, a_{3(r-1)})$ where $1 \le a_{3k} \le p + q$ for all $1 \le k \le r - 1$, and $\mathbf{a} = (a_1, a_2)$ where $1 \le a_1 \le n$ and $a_2 = n$, there is a tree (which is unique from above) with $(\mathbf{a_1}, \mathbf{a_2}, \mathbf{a_3}, \mathbf{a})$ as its tripartite Prüfer code. Let $(\mathbf{a_1}, \mathbf{a_2}, \mathbf{a_3}, \mathbf{a})$, be such a quadruple.

Define u_i recursively by the following.

Let $u_1 = \min ([n]\backslash\{a_{11}, a_{12}, \ldots, a_{1(p-1)}, a_{21}, a_{22}, \ldots, a_{2(q-1)}, a_{31}, a_{32}, \ldots, a_{3(r-1)}, a_1, a_2\})$.

Then $u_1 \in V_h$ for some h. For $j \in [3]$, define $f_j(0) = 1$, $f_j(1) = \begin{cases} 2 & \text{if } j = h \\ 1 & \text{otherwise} \end{cases}$, and

$f_4(1) = \begin{cases} 1 & \text{if } V_h\backslash\{u_1\} \neq \varnothing \\ 2 & \text{otherwise} \end{cases}$. Then in general, define

$u_i = \min ([n]\backslash(\{u_1, \ldots, u_{i-1}\} \cup \{a_{1f_1(i-1)}, a_{12}, \ldots, a_{1(p-1)}, a_{2f_2(i-1)}, a_{22}, \ldots, a_{2(q-1)},$

$a_{3f_3(i-1)}, a_{32}, \ldots, a_{3(r-1)}, a_{f_4(i-1)}, a_2\}))$, with $f_j(i-1) = \begin{cases} f_j(i-2) + 1 & \text{if } j = h \\ f_j(i-2) & \text{otherwise} \end{cases}$ for

$j \in [3]$, and $f_4(i-1) = \begin{cases} 1 & \text{if } V_h\backslash\{u_1, \ldots, u_{i-1}\} \neq \varnothing \\ 2 & \text{otherwise} \end{cases}$.

Note that $f_1(i) + f_2(i) + f_3(i) + f_4(i) = i + 4$. From above we see this is true for $i = 1$. Then assuming $f_1(i-1) + f_2(i-1) + f_3(i-1) + f_4(i-1) = (i-1) + 4 = i + 3$, we have that $f_1(i) + f_2(i) + f_3(i) + f_4(i) = f_1(i-1) + f_2(i-1) + f_3(i-1) + f_4(i-1) + 1 = i + 3 + 1 = i + 4$. Then $[n]\backslash(\{u_1, \ldots, u_{i-1}\} \cup \{a_{1f_1(i-1)}, a_{12}, \ldots, a_{1(p-1)}, a_{2f_2(i-1)}, a_{22}, \ldots, a_{2(q-1)},$ $a_{3f_3(i-1)}, a_{32}, \ldots, a_{3(r-1)}, a_1, a_2\})$ is never empty as there are at most $n - 1$ elements in

$\{u_1, \ldots, u_{i-1}\} \cup \{a_{1f_1(i-1)}, a_{12}, \ldots, a_{1(p-1)}, a_{2f_2(i-1)}, a_{22}, \ldots, a_{2(q-1)}, a_{3f_3(i-1)},$

$a_{32}, \ldots, a_{3(r-1)}, a_1, a_2\}$ (by counting indices, we have at most

$(i-1) + (p-1) + (q-1) + (r-1) - [f_1(i-1) - 1] - [f_2(i-1) - 1] - [f_2(i-1) - 1] -$
$[f_3(i-1) - 1] - [f_4(i-1) - 1] + 2$

$= (i-1) + (p-1) + (q-1) + (r-1) + 4 - [f_1(i-1) + f_2(i-1) + f_3(i-1) + f_4(i-1)] - 2$

$= i - 1 + p + q + r - 3 + 4 - (i + 4) - 2$
$= n - 6$
$\leq n - 1$ elements in the union)

and n elements in $V_1 \cup V_2 \cup V_3 = [n]$. Once again, this definition compels $(u_1, \ldots, u_{n-1})$ to be a permutation of $[n - 1]$. Define $T = (V_1 \cup V_2 \cup V_3, E)$ where

$E = \{\{u_i, k_i\} \mid 1 \leq i \leq n + m\}$ where $k_i = \begin{cases} a_{f_4(i-1)} & \text{if } V_h \setminus \{u_1, \ldots, u_i\} = \varnothing \\ a_{hf_h(i-1)} & \text{otherwise} \end{cases}$, where

$u_i \in V_h$. Also, for $1 \leq i \leq n$, let $V_{1,i} = V_1 \setminus \{u_1, \ldots, u_{i-1}\}$, $V_{2,i} = V_2 \setminus \{u_1, \ldots, u_{i-1}\}$, $V_{3,i} = V_3 \setminus \{u_1, \ldots, u_{i-1}\}$, and $E_i = E \setminus \{\{u_i, k_i\} \mid 1 \leq j < i\}$ and $T_i = (V_{1,i} \cup V_{2,i} \cup V_{3,i}, E_i)$. We claim that the resulting graph T is a tree with tripartite Prüfer code

$\mathbf{a_1} = (a_{11}, a_{12}, \ldots, a_{1(p-1)})$, $\mathbf{a_2} = (a_{21}, a_{22}, \ldots, a_{2(q-1)})$, $\mathbf{a_3} = (a_{31}, a_{32}, \ldots, a_{3(r-1)})$, and $\mathbf{a} = (a_1, a_2)$. We need only to show that u_i is an endpoint if the graph T_i and that no other vertex with smaller label is an endpoint. Since u_i is adjacent to k_i in T_i and $k_i = u_j$ for some $j > i$ (that is, k_i is an endpoint of a later T_j), we can trace a path each u_i to a_n, showing connectedness of T and each T_i.

Note that $\{u_i, k_i\} \in E_i$. Therefore u_i has a neighbor in T_i, namely k_i. Also u_i cannot be adjacent to any other vertex of T_i, since if it were, then for some j, $\{k_j, u_j\}$ is also an edge of T_i that includes u_i. Then $j > i$ as u_j is a vertex of T_i. Also, we have that either $u_i = u_j$ or $u_i = k_j$. But $u_i \neq u_j$ as $u_i \notin T_j$, and $u_i \neq k_j$ since k_j will equal some later a_{xy}. Thus u_i is an endpoint of T_i. Hence we have that T and all T_i's are trees. Thus as T is a tree, it has a tripartite Prüfer code. Note that we have just defined each u_i as it was defined in (1). Therefore u_i is the vertex of smallest label in T_i.

We have shown that there is a one-to-one correspondence between the set of spanning trees on $K_{p,q,r}$, and the set of quadruples of ordered sequences of integers $(\mathbf{a_1}, \mathbf{a_2}, \mathbf{a_3}, \mathbf{a})$, with $\mathbf{a_1} = (a_{11}, a_{12}, \ldots, a_{1(p-1)})$ where $p + 1 \leq a_{1i} \leq n$ for all $1 \leq i \leq p - 1$,
$\mathbf{a_2} = (a_{21}, a_{22}, \ldots, a_{2(q-1)})$ where $1 \leq a_{2j} \leq p$ or $(p + q + 1) \leq b_j \leq n$ for all $1 \leq j \leq q - 1$,

$a_3 = (a_{31}, a_{32}, \ldots, a_{3(r-1)})$ where $1 \le a_{3k} \le p + q$ for all $1 \le k \le r - 1$, and $a = (a_1, a_2)$ where $1 \le a_1 \le n$ and $a_2 = n$. Hence, as the total number of such sequences is $n(n-p)^{p-1}(n-q)^{q-1}(n-r)^{r-1}$, we have that the total number of spanning trees of $K_{p,q,r}$ is $n(n-p)^{p-1}(n-q)^{q-1}(n-r)^{r-1}$. ■

CONCLUSION

We have now proved Cayley's theorem using three different proofs. We first proved it using a little bit of algebra and Kirchhoff's matrix tree theorem. We then showed there is a one-to-one correspondence between trees on [n] and Prüfer codes. Finally we counted the number of trees by using degree sequences and properties of multinomial coefficients.

Once we showed all of those results we turned to the number of spanning trees on the complete bipartite graph, and extended each of the three proofs. Finally we extended the first two results to count the number of spanning trees on the complete tripartite graph. The reader will note that we did not extend the degree sequence argument for the tripartite case. The problem that arises is knowing what acceptable degree sequences look like to ensure that we obtain a tripartite tree.

One might hope for an extension to the number of spanning trees on the complete k – partite graph on $n = n_1 + n_2 + \ldots + n_k$ vertices, $K_{n_1,n_2,\ldots,n_k}$, and it turns out that such an extension exists **[4, p. 338]**. It is obvious how one might prove this results algebraically using Kirchhoff's Theorem, but the mechanics involved appear formidable. A combinatorial proof using a k – partite Prüfer code looks practicable. The k – partite Prüfer code will take the form of a $k + 1$ tuple of sequences of integers

$(\mathbf{a_1}, \mathbf{a_2}, \ldots, \mathbf{a_k}, \mathbf{a})$ in which $\mathbf{a_1} = (a_{11}, a_{12}, \ldots, a_{1(n_1-1)})$ with $a_{1j} \in [n]\backslash V_1$, $\mathbf{a_2} = (a_{21}, a_{22}, \ldots, a_{2(n_2-1)})$ with $a_{2j} \in [n]\backslash V_2$, ..., $\mathbf{a_k} = (a_{k1}, a_{k2}, \ldots, a_{k(n_k-1)}))$ with $a_{kj} \in [n]\backslash V_k$, and $\mathbf{a} = (a_1, a_2, \ldots, a_{k-1})$ with $a_i \in [n]$ for $1 \leq i \leq k-2$ and $a_{k-1} = n$.

Given a k – partite tree, one can form the code in the usual way: Find the endpoint of smallest label and call it u_1. If $u_1 \in V_j$, record to $\mathbf{a_j}$ the label of its adjacent vertex. Then remove u_1 and its incident edge from the tree. Then repeat the process. Whenever u_i is the sole remaining vertex of V_j, for $u_i \in V_j$, record to $\mathbf{a}$ the label of its adjacent vertex. This process will result in the k – partite Prüfer code
$(\mathbf{a_1}, \mathbf{a_2}, \ldots, \mathbf{a_k}, \mathbf{a})$.

Using similar tactics to the ones already used, one can rigorously prove that this is indeed a one-to-one correspondence. Therefore as there are $n - n_1$ possible elements that a_{1j} can take on, there are $(n - n_1)^{n_1 - 1}$ ways to complete $\mathbf{a_1}$. Similarly, there are $(n - n_2)^{n_2 - 1}$ possibilities for $\mathbf{a_2}$. In general, for a_{ij} there are $n - n_i$ possible values, and therefore $(n - n_i)^{n_i - 1}$ ways to complete $\mathbf{a_i}$. Completing $\mathbf{a}$ can be done in n^{k-2} ways as there are $k - 2$ elements which can take on any value in $[n]$, with the last element predetermined. Hence, putting it all together, we have that there are
$n^{k-2}(n - n_1)^{n_1 - 1}(n - n_2)^{n_2 - 1} \ldots (n - n_k)^{n_k - 1}$ such $k + 1$ tuples. Therefore, there are $n^{k-2}(n - n_1)^{n_1 - 1}(n - n_2)^{n_2 - 1} \ldots (n - n_k)^{n_k - 1}$ spanning trees on $K_{n_1, n_2, \ldots, n_k}$.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Martin Aigner and Günter M. Ziegler, *Proofs from THE BOOK*, Springer-Verlag, Berlin Heidelberg New York 1998

[2] Lowell W. Beineke and Robin J. Wilson, *Selected Topics In Graph Theory*, Academic Press 1978

[3] Claude Berge, *Graphs, Second Edition*, North – Holland Publishing Company 1985

[4] Rainer Bodendiek and Rudolf Henn, *Topics in Combinatorics and Graph Theory*, Physica – Verlag Heidelberg 1990

[5] G. Chartrand and L. Lesniak, *Graphs and Digraphs, Third Edition*, Chapman and Hall 1996

[6] Frank Harary, *A Seminar on Graph Theory*, Holt, Rinehart and Winston, Inc. 1967

[7] Richard Johnsonbaugh, *Discrete Mathematics, Fourth Edition*, Prentice Hall 1997

[8] V. Krishnamurthy, *Combinatorics: Theory and Applications*, Ellis Horwood Limited 1986

[9] László Lovász, *Combinatorial Problems and Exercises, Second Edition*, North – Holland Publishing Company 1993

[10] Wataru Mayeda, *Graph Theory*, Wiley – Interscience 1972

[11] Igor Pak, *Igor Pak Home Page*, http://www-math.mit.edu/~pak accessed February 2003

[12] Carl Wagner, *Enumerative Combinatorics Lecture Notes* 2002-2003

[13] Herbert S. Wilf, *generatingfunctionology*, Academic Press, Inc. 1990 and 1994

[14] Robin J. Wilson, *Introduction to Graph Theory*, Academic Press, Inc. 1972

VITA

Michelle Renee Brown was born in Honolulu, Hawaii on February 25, 1976. As a child she moved around until her family settled in Columbus, Georgia where she attended Daniel Junior High School and Jordan Vocational High School. She began her college career at Columbus State University but completed her Bachelor of Science with a major in mathematics in 2000 at the University of Alaska, Fairbanks.

Michelle entered graduate school at the University of Tennessee, Knoxville in August 2001 and received her Master of Science with a major in mathematics in August 2003.