Forest Generating Functions of Directed Graphs

by

Ewan Joaquin Kummel

A dissertation submitted in partial fulfilment of the
requirements for the degree of

Doctor of Philosophy
in
Mathematical Sciences

Dissertation Committee:
John Caughman, Chair
J.J.P. Veerman
Panayot Vassilevski
Karen Karavanic

Portland State University
2023

Abstract

A spanning forest polynomial is a multivariate generating function whose variables are indexed over both the vertex and edge sets of a given directed graph. In this thesis, we establish a general framework to study spanning forest polynomials, associating them with a generalized Laplacian matrix and studying its properties. We introduce a novel proof of the famous matrix-tree theorem and show how this extends to a parametric generalization of the all-minors matrix-forest theorem. As an application, we derive explicit formulas for the recently introduced class of directed threshold graphs.

We prove that multivariate forest polynomials are, in general, irreducible and we define a number of specializations that may be compactly expressed in terms of various factors. A specialization in this context is an identification of some of the variables of the polynomial, for example evaluating f(x,y,z) as f(x,x,z). This allows us to derive results that generalize and extend many known properties of the traditional Laplacian matrix in algebraic graph theory.

We analyze the connection between the matrix algebra generated by the traditional Laplacian matrix and certain matrices of forest polynomials. Using this analysis, we derive explicit formulas for these matrices in the cases of Cartesian products of complete graphs and de Bruijn graphs. More generally, we derive an explicit formula relating spanning forest polynomials of a graph to the numbers of D-lazy walks in the graph. These are walks that may choose to remain at a given vertex if that vertex is not of maximum degree D.

This leads us to the study of externally equitable partitions (EEPs), which are objects of recent interest in the control theory literature. We prove that for graphs with EEPs satisfying an additional criteria, the specialized forest polynomials may be factored into a product of forest polynomials of related quotient graphs. We apply this theorem to complete multipartite graphs, hypercube graphs, directed line graphs, and others.

Acknowledgements

My deepest gratitude to all of the excellent educators at Reed College and Portland State University who helped me develop my knowledge and passion in one way or another. Special thanks to my advisor John whose patience, support and attention to detail have greatly improved the final product. Thanks to my parents Marc and Julie, my wife Rosamund, and my kids Ottokar and Kelvin for their continued support and love. Thanks to Jen, Dan, Jen, Jamie, Ryan, Alyssa, Annette, Nathan, Lucas, and Suzy for being awesome friends and providing me with much needed distraction. Finally, thanks to my recently departed pets Rufus and Chester who provided me so much companionship during this whole process. I will continue to miss you both.

Contents

List of Figures

Introduction

Spanning trees of graphs and their associated generating functions have played an important role in algebraic graph theory since its inception. Recently, spanning tree polynomials have been of particular interest in the theory of chemical reaction networks as they are key to describing the steady state solutions of such systems [52, 44, 43, 51]. However, there are challenges to extending these methods to directed graphs, which may not have spanning trees. One way to overcome this is to replace spanning trees with rooted spanning forests.

In this thesis, we establish a framework for studying spanning forest polynomials by associating them with a generalized Laplacian matrix and studying its properties. We also analyze the connection between the matrix algebra generated by the Laplacian matrix and certain matrices of forest polynomials. Using this analysis, we derive explicit formulas for these matrices in the cases of Cartesian products of complete graphs and de Bruijn graphs.

We prove that multivariate forest polynomials are, in general, irreducible. However, when we set some of the variables in an irreducible spanning forest polynomial equal to each other, we open up the possibility that the resulting polynomial is reducible.

One example of this phenomena occurs when variables are identified along an externally equitable partition (EEP). These are partitions of the vertex set of a graph that have been of recent interest in the control theory literature, in particular, as a away of characterizing the controllability of certain networks [23, 7]. We prove that for graphs with EEPs satisfying an additional constraint, the specialized forest polynomials may be factored into a product of forest polynomials of related quotient graphs. This allows us to derive some compact expressions for multivariate forest polynomials of complete multipartite graphs and directed line graphs, and others.

In the first chapter, we introduce basic notation. Our interest is with both edge and vertex labeled directed graphs. This necessitates an approach that is not necessarily standard, but greatly simplifies our exposition. In particular, we make fundamental use of what we call *source* and *target* incidence functions to define more common notions like vertex degree, neighborhood, and so on.

In Chapter 2, we introduce notation for different sets of converging spanning forests of a graph and define multivariate root parameterized forest polynomials as well as various specializations. We prove some basic facts about these polynomials. Most important among these is the fact that multivariate forest polynomials of strongly connected graphs are always irreducible.

In Chapter 3, we define the generalized Laplacian matrix and prove the Matrix Forest Theorem as well as the all-minors generalization. After deriving some important corollaries, we look at an important specialization that defines what we call the univariate forest matrix of a graph and show its connection to traditional objects of study in spectral graph theory. We conclude this chapter with an explicit derivation of the multivariate and univariate forest polynomials of the family of directed threshold graphs.

In Chapter 4, we study the unweighted univariate forest matrix. We prove a reciprocity result and use this to derive explicit formulas for complete multipartite graphs. This leads to a discussion of matrix algebras. After an in-depth analysis of the algebras generated in the case of de Bruijn graphs and Cartesian products of complete graphs, as well as derivations of their unweighted univariate forest matrices, we turn to a more general discussion that culminates in a theorem relating the forest polynomials of a directed graph to the coefficients of its Laplacian matrix as well as the numbers of lazy random walks in the graph.

Finally, in Chapter 5, we attempt to ameliorate the irreducibility of multivariate forest polynomials by looking at specializations of their variables that allow for more compact expressions. After finding expressions for acyclic graphs and graphs with few directed cycles, we find a specialization of the weights of the complete graph that permits a compact expression of its forest polynomial. After noting that any further specialization leads to a reducible polynomial, this leads us to a general theorem regarding factorization in the presence of so-called externally equitable partitions. After proving this theorem, we note a number of results in the literature that may be seen as applications.

1 Notation

In this chapter, we collect a number of common definitions relevant to our work, and we fix notation that will be useful throughout.

1.1 Directed Graphs

An *undirected graph*, G, is a set V of *vertices* together with a set E of *edges* and a function $i : E \to \binom{V}{2} \cup \binom{V}{1}$. Edges that are mapped by i to 1-element sets are called *loops*, while those mapped by i to 2-element sets are called *non-loop edges*. G is *simple* if i is injective and all edges are mapped to 2-element sets (so there are no loops or multiple edges). Otherwise, we say G is a *multigraph*. G becomes *directed* if we supply functions $s : E \to V$ and $t : E \to V$ such that, for each e, we have

$i(e) = \{s(e), t(e)\}$. In this case, e is said to point from $s(e)$ to $t(e)$ and we call $t(e)$ the *target* of the edge and $s(e)$ the *source*. We will consider e to be synonymous with the expression "$i \to j$" when $i = s(e)$ and $j = t(e)$. Finally, G becomes *(edge) weighted* if we supply a weight function $w : E \to W$ for some set W. We will generally assume that W is a set of indeterminates given by $W = \{\omega_e\}_{e \in E}$, and we evaluate these indeterminates if specific weights are needed. By indexing the set W by E, we can usually suppress the weight function w. Note that we do not assume at any point in our exposition that these indeterminates are positive, real or even complex. We do however require that they commute with each other.

Hereafter, we shall use the term *graph* to refer to weighted directed multi-graphs in general, preferring to specialize this term with descriptors like "simple" and "undirected" when necessary. Thus, when we say "let G be a graph", we mean that $G = (V, E, s, t, w)$.[1] When it becomes necessary to distinguish these functions for different graphs, we will use $V(G)$, $E(G), s_G$, and so forth. All graphs considered in this paper are finite and we hereafter assume, unless otherwise stated, that $V = \{1, 2, \ldots, n\}$ and $E = \{1, 2, \ldots, m\}$ for some positive integers n and m. We adopt the convention that $[n]$ denotes the set $\{1, \ldots, n\}$. We define the *order* of G to be $|V| = n$ and the *size* of G to be $|E| = m$. If the edge set of G is only specified by the values of $s(e)$ and $t(e)$ for each e, then we may assume that it is ordered lexicographically so that $(i_1 \to j_1) < (i_2 \to j_2)$ just in case $i_1 < i_2$ or both $i_1 = i_2$ and $j_1 < j_2$.

A *graph isomorphism* from G to H is a pair of bijective maps $\Phi : V_G \to V_H$ and $\Psi : E_G \to E_H$ satisfying $s_H(\Psi(e)) = \Phi(s_G(e))$ and $t_H(\Psi(e)) = \Phi(t_G(e))$. If G and H are simple graphs, then the map Ψ is induced by Φ via $\Psi(i \to j) = \Phi(i) \to \Phi(j)$.

[1] We omit the function i since it is uniquely determined by s and t.

In this case, the definition reduces to the more familiar $i \to j$ in G if and only if $\phi(i) \to \phi(j)$ in H.

In this work, we find it convenient to replace undirected graphs with a related directed analog. Given a *loopless* undirected graph U, we define the associated directed graph to have vertex set $V(U)$ and edge set $E(U) \times \{0,1\}$ with $s(e,0) = \min i_U(e)$, $t(e,0) = \max i_U(e)$, $s(e,1) = \max i_U(e)$, and $t(e,1) = \min i_U(e)$. In other words, each undirected edge is replaced by a pair of oppositely directed edges. We assign each of these opposite directed edges its own weight variable. In what follows, the names of all well known families of undirected graphs will refer to their directed counterparts.

For $v \in V(G)$, we define the *source degree* and *target degree* of v to be the sums $d^s(v) = \sum_{e \in s^{-1}(v)} \omega_e$ and $d^t(v) = \sum_{e \in t^{-1}(v)} \omega_e$ respectively. Similarly, the *target neighborhood* of v is the set of vertices $N^t(v) = t(s^{-1}(v))$ and the *source neighborhood* is the set $N^s(v) = s(t^{-1}(v))$. In words, the target neighborhood of v in G is the set of all vertices u in G so that $u \to v$ is an edge of G and similarly for the source neighborhood. We will also apply the functions s and t and their inverses to sets of edges M and vertices U using the convention that, for example, $s(M) = \cup_{e \in M} s(e)$ and $s^{-1}(U) = \cup_{v \in U} s^{-1}(v)$. Note that these are understood to be sets of vertices and edges and not multisets or sets of sets.

We say that a graph H is a *subgraph* of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$ and s_H, t_H, w_H are the restrictions of s_G, t_G, w_G to $E(H)$. If, for every $e \in E(G)$ with $s(e), t(e) \in V(H)$, we have $e \in E(H)$, then H is an *induced* subgraph of G. Given a subset $S \subseteq V(G)$, there is a unique induced subgraph of G with vertex set S. We call this the *subgraph of G induced by S* .

If the edges of H can be labeled $E(H) = \{e_0, \ldots, e_k\}$ such that $t_H(e_j) = s_H(e_{j+1})$

for $j = 0, \ldots, k-1$ and $t_H(e_k) \neq s_H(e_0)$, then we call H a *directed path* in G. If the first condition holds, but $t_H(e_k) = s_H(e_0)$, then H is a *directed cycle.* If no subgraph of G is a directed cycle, then G is *acyclic.*

If a subgraph H of G satisfies $V(H) = V(G)$, then H is a *spanning* subgraph of G. We will be particularly interested in a few special kinds of spanning subgraphs. A spanning subgraph H of G is a *converging forest* of G if it is acyclic and there is a non-empty set $R \subseteq V(H)$ such that $d^s_H(r) = 0$ for $r \in R$ and $d^s_H(v) = 1$ for $v \in V(H) \backslash R$.

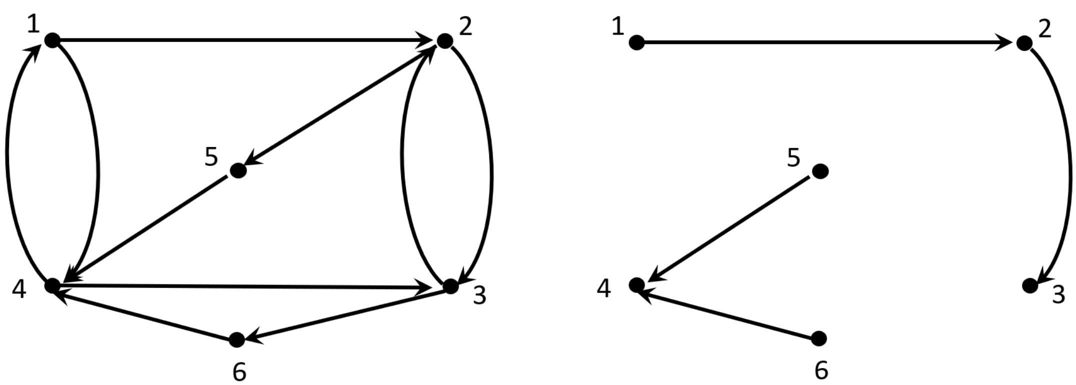


Figure 1.1: A Graph G and a converging forest of G with root set $\{3, 4\}$.

On the other hand, H is a *diverging forest* of G if there is a non-empty set $R \subseteq V(H)$ such that $d^t_H(r) = 0$ for $r \in R$ and $d^t_H(v) = 1$ for $v \in V(H) \backslash R$. In either case, the elements of R are called the *roots* of H. Given such an H, we let $R(H)$ denote the set of roots of H. If H is a converging (respectively diverging) forest with $|R| = 1$, then we call H a *converging (respectively diverging) tree.*

In a converging spanning forest, each non-root vertex has source degree 1 and each connected component has exactly one root. If we relax the acyclic condition, allowing a component to have no root, we get the *functional digraphs*, so called because they are in one to one correspondence with the set of functions defined from V to V. In such a graph, any component that does not have a root terminates in a directed cycle. (There is also a corresponding generalization of diverging spanning forests in which each non-root vertex has target degree 1. This class of graphs does not seem to have a name but we suggest that they be called *reverse functional digraphs*.) A member f of any of these classes is called *maximal* if the addition of any edge of G to f implies that f no longer belongs to the class.

In the context of directed graphs, we must distinguish a few different notions of connectivity. G is *strongly connected* if for any (ordered) pair u, v there is a directed path from u to v. If a graph is not strongly connected, then we say that it is *unilaterally connected* if it contains a vertex v so that for any other vertex u there is either a directed path from u to v. An induced subgraph H of G is a *strongly connected component* if it is strongly connected and any induced subgraph H' of G that properly contains H is not strongly connected. An edge e of G that belongs to a strongly connected component of G is called *essential* while any other edge is called *transient*. Note that at least one of the strongly connected components of G, T, must be *terminal* in the sense that for all $v \in T$, $N_G^t(v) \subseteq T$. In other words, a strongly connected component of G is terminal when there are no edges pointing from it to some other strongly connected component.

The notion of a graph complement will play a role in some of our results. This simple idea is complicated by the presence of multiple edges and edge weights. In general, we will define a graph complement of G relative to a "complete" graph K. We require that $E(G) \subseteq E(K)$ and then the complement of G, $\bar{G}$ is defined by $V(\bar{G}) = V(G)$ and $E(\bar{G}) = E(K) - E(G)$. This allows us to unambiguously define the functions s, t. In practice, we will only apply this construction to graphs without multiple edges so that K can always be take to have edge set $[n] \times [n]$.

We will also be interested in a few different methods of building a larger graph out of smaller graphs. If G and H are graphs, we let the *disjoint union* of G and H be the graph $G \dot{+} H$ with vertex set $V(G) \times \{0\} \cup V(H) \times \{1\}$ and edge set $\{(s(e), 0) \to (t(e), 0)\}_{e \in E(G)} \cup \{(s(e), 1) \to (t(e), 1)\}_{e \in E(H)}$.

We further define $G \times H$, the *Cartesian product* of G and H, to be the graph with $V = \{(v, w) \mid v \in V_G \text{ and } w \in V_H\}$ with edge set defined by $(a, b) \to (c, d)$ in $G \times H$ if and only if $a = b$ in G and $c \to d$ in H, or $c = d$ in H and $a \to b$ in G.

Additionally, the *directed line graph* of G, denoted $DL(G)$, is the graph with vertex set $V_{DL(G)} = E_G$ and edge set $E_{DL(G)} = \bigcup_{v \in V_G} s_G^{-1}(v) \times t_G^{-1}(v)$. In words, each edge of G whose target in G is v points in $DL(G)$ to every edge of G whose source is v in G. In each of these cases, we will assume a natural ordering of vertices inherited from the graphs G and H using the lexicographic ordering.

1.2 Linear Algebra

We are concerned with matrices whose rows and columns are indexed by sets of vertices or sets of edges of some graph. It is important to note that such matrices will depend on an explicit ordering of both V and E. In the previous section, we have built in an explicit ordering of vertices and edges by identifying these sets with sets of consecutive integers. In the rare cases below where we need a different description of V or E, we will make sure to describe an ordering as well.

The incidence functions s and t determine the $E \times V$ *source* and *target incidence matrices* of G denoted S_G and T_G respectively. When G is obvious from context, we will omit the subscripts. The entries of S are given by $S_{e,i} = 1$ if $s(e) = i$ and 0 otherwise while the entries of T by $T_{e,i} = 1$ if $t(e) = i$ and 0 otherwise. Using these matrices, we can form the *adjacency matrix* A of G, defined by $A_{ij} = \omega_e$ just in case $i \to j$ in G. The *unweighted adjacency matrix* is defined similarly, with each ω_e replaced by 1.

Proposition 1.2.1. *For a graph G of size m with source and target matrices S and T respectively, the adjacency matrix of G is given by S^TWT, where W is an $m \times m$ diagonal matrix with $W_{ee} = \omega_e$. The unweighted adjacency matrix of G is given by S^TT.*

Proof. This is clear by the definition of matrix multiplication. $\square$

Given matrix M, with rows indexed by X and columns by Y, with subsets $A \subseteq X$ and $B \subseteq Y$, we let $M_{A,B}$ denote the submatrix of M containing only those rows indexed by elements of A and columns indexed by elements of B. We assume that A and B are given the ordering induced from the orderings of X and Y respectively. We will also let $M_{[A,B]} = M_{X-A,Y-B}$.

Throughout the text, we let I_k equal the $k \times k$ identity matrix and J_k equal the $k \times k$ all-ones matrix. In each case, we suppress the subscript whenever the dimension of the matrix is clear from context. Given a vector $x = (x_1, \dots, x_k)$, we let $D(x)$ denote the $k \times k$ diagonal matrix whose (i, i) entry is x_i.

Given matrices M and N with sizes $m_1 \times m_2$ and $n_1 \times n_2$ respectively, the Kronecker product, $M \otimes N$ is an $m_1n_1 \times m_2n_2$ matrix whose i, j entry is defined by the product $M_{a,b}N_{c,d}$ where a, b, c, and d are defined by $i = (a-1)n_1 + (b-1)$ and $j = (c-1)n_1 + (d-1)$. It is important to note that this definition allows us to index the rows of $M \otimes N$ by ordered pairs (a, c) where $1 \le a \le m_1$ and $1 \le c \le n_1$. The columns of $M \otimes N$ may be indexed similarly. If M and N are square matrices, the

Kronecker sum $M \oplus N$ is defined as $M \otimes I_{n_1} + I_{m_1} \otimes N$. Note that the rows and columns of $M \oplus N$ may be indexed in the same manner as $M \otimes N$.

Below, we will make use of iterated Kronecker products and sums. These both follow the summation notation so that, for example, $\bigotimes_{i=1}^{k} A_k = A_1 \otimes A_2 \otimes \cdots \otimes A_k$. In light of the comments in the previous paragraph, it should be clear that such a matrix comes with a number of different ways that its rows and columns may be indexed. For example, a row of $\bigotimes_{i=1}^{k} A_k$ might be referenced by an integer or a k-tuple whose ith entry is a row of A_i. However, there are many other possibilities. For example, if $k = 5$, then we might also index a row by, for example, the triple (x, y, z) where x is a row of A_1, y is a row of $A_2 \otimes A_3$, and z is a row of $A_4 \otimes A_5$.

Proposition 1.2.2. *For graphs G and H with unweighted adjacency matrices A_G and A_H, the disjoint union $G \dot{+} H$ has a block diagonal unweighted adjacency matrix with 2 diagonal blocks equal to A_G and A_H respectively. The cartesian product $G \times H$ has unweighted adjacency matrix $A_G \oplus A_H$.*

Proof. The claim about the disjoint union is obvious from the definitions. For the cartesian product, we partition the edges into two sets. If $e \in V(G \times H)$, then say $s(e) = (u, v)$ and $t(e) = (x, y)$ with $u, x \in V(G)$ and $v, y \in V(H)$. Now, from the definition, we have either $u = x$ and $v \to y$ in H or $u \to x$ and $v = y$. If E_1 and E_2 are the spanning subgraphs of G containing all edges satisfying the first and second conditions respectively, then we claim that $A_{E_1} = I \otimes A_G$ and $A_{E_2} = A_H \otimes I$. Using the ordering discussed in the previous paragraph, we consider the $(u, v)(x, y)$ entry of $I \otimes A_G$. This is $I_{ux}(A_G)_{vy}$ which is 1 just in case $u = x$ and $v \to y$ in H. Thus, $A_{E_1} = I \otimes A_G$. A similar argument establishes the claim for A_{E_2}. $\square$

Given a matrix M, we define the adjugate matrix $\operatorname{adj} M$ to be the unique matrix satisfying M adj$M = (\det M)I$. From basic linear algebra, see for example [31], we know that adj$M$ has ij entry equal to $(-1)^{i+j} \det M_{[\{j\},\{i\}]}$. Note that if M is a matrix over ring R, then adj$M$ is also a matrix over R.

1.3 Ordered Partitions

Let M be a set and Π be a partition of M into k nonempty parts. We assume that the sets that make up Π are given an ordering via $\Pi = \{\Pi_1, \ldots, \Pi_k\}$. If $x \in M$, we write $\Pi(x) = i$ when $x \in \Pi_i$. Similarly, if $Y \subseteq M$, then $\Pi(Y) = \{\Pi_j \mid Y \cap \Pi_j \neq \emptyset\}$. Suppose that $M = \{1, 2, \ldots, n\}$. In matrix computations, we will also identify a partition Π with an $n \times k$ matrix whose (i, j) entry is 1 if $i \in \Pi_j$. Note that the source and target matrices defined above are each examples of partition matrices. We will endeavor to make the appropriate interpretation of Π clear from context.

We note some basic properties of partition matrices here. In particular, since each part of Π is nonempty, the columns of the matrix Π are linearly independent and the $k \times k$ diagonal matrix $D_\Pi = D(\Pi \mathbb{1})$ whose ii entry is $|\Pi_i|$ is invertible. It follows that $D_\Pi^{-1} \Pi^T \Pi = I_k$. On the other hand, $\Pi \Pi^T = [P_{ij}]_{1 \leq ij \leq k}$ is an $k \times k$ block matrix whose ij block has dimension $|\Pi_i| \times |\Pi_j|$ and has each of its entries equal to 1.

1.4 Multivariate Polynomials

Let R be an integral domain and let $x = (x_1, \ldots, x_k)$. We consider the multivariate polynomial ring $R[x]$. Recall that this is also an integral domain. Given a subset $s \subseteq [k]$, we set $x^S = \prod_{i \in S} x_i$. By convention, we let $x^\emptyset = 1$. We say that $P \in R[x]$ is *homogeneous* if each monomial of P has the same degree. Additionally, P is *multiaffine* if each monomial of P has the form $\alpha_S x^S$ for some subset $S \subseteq [m]$ and $\alpha_S \in R$. Equivalently, P is multiaffine if it is homogeneous and linear in each of its variables. Finally, P is *irreducible* over R if, for any polynomials $Q, R \in R[x_1, \ldots, x_m]$ we have that $P = QR$ implies that either Q or R has degree 0.

In this work, we are concerned with polynomials whose variables are related to a given graph G with vertex set V and edge set E. For each $v \in V$, we introduce variable τ_v and for each edge $e \in E$, we introduce variable ω_e. We assume that these variables all commute with each other. We will often assemble these variables into vectors τ and ω using the assumed orderings of V and E. At the most general level, we will be concerned with polynomials in $\mathbb{Z}[\tau, \omega]$. The entries of τ are called *type-τ variables* and similarly for entries of ω.

If $u \in \mathbb{Z}^V$, so that u is a sequence of integers indexed by the vertices of G, then we define the monomial τ^u to be the monomial whose exponents are given by u. In other words, we have $\tau^u = \prod_{v \in V} \tau_v^{u_v}$. We note the following special case. If $S \subseteq V$ and $\mathbb{1}_S$ is the vector with all entries indexed by elements of S equal to 1 and the rest equal to 0, then we abbreviate the monomial $\tau^{\mathbb{1}_S}$ by simply τ^S. We follow an identical convention regarding E and ω. Further, we will associate the name of a graph with its edge set. Thus, if H is a subgraph of G, then $\omega^H = \prod_{e \in H} \omega_e$.

Let $x = [x_1, \ldots, y_n]$ and $y = [y_1, \ldots, y_k]$ be vectors of indeterminates. Let P be a polynomial over $R[x]$ and Π be a partition of $[n]$ into k parts. Then, note that $P(\Pi y)$ is a polynomial over $R[y]$. In most cases, P will be an element of $\mathbb{Z}[\tau, \omega]$ as defined in the previous section. We refer to the process of identifying some of the variables of a polynomial as *specialization* of that polynomial.

1.5 Permutations

For any positive integer m, let $[m] = \{1, 2, ..., m\}$. Recall that for any such m, a *permutation* is a bijective map $\pi : [m] \to [m]$. Given an indexed set of size m, say $S = \{s_1, \ldots, s_m\}$, we can define the action of π on S as $\pi s_i = s_{\pi(i)}$.

For any permutation π acting on a set S, we define $\mathrm{cyc}(\pi)$ and $\mathrm{fix}(\pi)$ to be the set of cycles of π and the set of fixed points of π respectively. Note that the former is a set of ordered sets of elements of S while the latter is just a subset of S. The *sign* of π is given by

$$\mathrm{sgn}(\pi) = (-1)^{n-|\mathrm{cyc}(\pi)|-|\mathrm{fix}(\pi)|}.$$

2 Generating Functions for Forests and Functional Digraphs

In this chapter, we define the basic objects of our study. Specifically, we introduce sets of spanning subgraphs of a given graph, as well as particular multivariate generating functions associated with such sets. We will prove some basic properties of these generating functions and discuss some specializations that we will examine later.

2.1 Sets of Spanning Forests and Functional Digraphs

Let G denote a graph. We begin by considering the set of all spanning functional digraphs of G as well as the subset of converging spanning forests.

Definition 2.1.1. *Let $\mathcal{D}_G$ denote the set of spanning functional digraphs of G and $\mathcal{F}_G$ denote the set of converging spanning forests of the graph G. Note that $\mathcal{F}_G \subseteq \mathcal{D}_G$.*

In what follows we will focus exclusively on converging spanning forests of a graph and their relation to spanning functional digraphs. The reader should note, however, that there is a parallel theory of diverging spanning forests that is naturally dual to the one discussed below. The role of functional digraphs in this dual theory is taken by what we have called reverse functional digraphs. In addition, we will now drop the terms "spanning" and "converging", assuming that e.g. "f is a forest of G" implies that f is a converging forest on vertex set $V(G)$ and $E_f \subseteq E_G$. Note that f still need not be connected, indeed the empty graph on the vertex set $V(G)$ is always a forest of G.

Definition 2.1.2. *If $A, B \subseteq V$, then $\mathcal{D}_G^{A \to *B}$ denotes the set of functional digraphs, f, of G such that $B \subseteq R(f)$ and each component of f rooted in some element of B contains exactly one element of A. Note that this set is empty unless $|A| = |B|$. We further define $\mathcal{F}_G^{A \to *B}$ to be the set $\mathcal{D}_G^{A \to *B} \cap \mathcal{F}_G$.*

When $A = B$, we abbreviate, for example, $\mathcal{F}_G^{A \to *A}$ as $\mathcal{F}_G^{\to *A}$. This is the set of all converging forests of G whose root set contains A. When A and B are singletons, say $A = \{i\}$ and $B = \{j\}$, then we omit the brackets and write, for example, $\mathcal{F}_G^{i \to *j}$ for the set of converging forests of G containing a path from vertex i to root vertex j.

Definition 2.1.3. *Given two ordered sets of vertices A, B with $|A| = |B| = k$ and any $f \in \mathcal{D}_G^{A \to *B}$, we define a permutation $\pi_{f,A,B}$ on $[k]$ by the condition that for each $a_i \in A$, we have a path from a_i to b_j in f if and only if $j = \pi_{f,A,B}(i)$.*

Note that we are not requiring that the sets A and B above be disjoint or even compatibly ordered. On the other hand, the definition given does require that any element $v \in A \cap B$ satisfy $v = a_i = b_{i'}$ where $i' = \pi_{f,A,B}(i)$. This need not imply that i is a fixed point of $\pi_{f,A,B}$. Indeed, if $v = a_i = b_j$ but $i \neq j$, then we would have

$\pi_{f,A,B}(i) = j$. However, in this situation, the definition requires that every member of $\mathcal{F}_G^{A\to *B}$ have vertex v as a root.

Example 2.1.4. If G is the complete graph on 6 vertices with $A = \{2,3,4\}$ and $B = \{2,5,3\}$, then $a_1 = 2, a_2 = 3$, and $a_3 = 4$ while $b_1 = 2, b_2 = 5$, and $b_3 = 3$. By our definitions, $\mathcal{F}_G^{A\to *B}$ contains the forest f with roots $\{2,3,5\}$ and edges $1 \to 6, 6 \to 3, 4 \to 5$. In this case, $\pi_{f,A,B}$ maps $1,2,3$ to $1,3,2$ respectively.

Lemma 2.1.5. *If e is a transient edge of G and $f \in \mathcal{F}^{\to *s(e)}$, then $f \cup e$ is a forest as well.*

Proof. Since $s(e)$ is a root in f, it has source degree 1 in $f \cup e$. Therefore, $f \cup e$ is a functional digraph. To see that it is also acyclic, note that since e is transient in G, e cannot belong to any cycle of G. □

Lemma 2.1.6. *Fix $i \in V$. Then the collection of sets $\{\mathcal{F}^{i\to *j}\}_{j\in V}$ is pairwise disjoint and its union is $\mathcal{F}^{\to *}$.*

Proof. In any forest $f \in \mathcal{F}$, there is a path from the vertex i to a unique root $r_i \in R(f)$. Thus, f belongs to $\mathcal{F}^{i\to *r_i}$ and no other member of the collection. □

Note that for a given $j \in V$, the set $\mathcal{F}^{i\to *j}$ may well be empty.

Definition 2.1.7. *If a forest has k roots, then we call it a k-*forest*. Using the same abbreviations as above, we let $\mathcal{F}_k$ denote the set of converging k-forests of G.* [2] *The sets $\mathcal{F}_k^{\to *A}$ and $\mathcal{F}_k^{A\to *B}$ are defined analogously.*

Definition 2.1.8. *The* forest dimension *of G is the minimum number d such that*

[2] If we need to specify both the graph and the number of roots we will use $\mathcal{F}_{G,k}$.

$\mathcal{F}_d$ *is nonempty.*

For the remainder of this section, we suppose that G has forest dimension d.

Proposition 2.1.9. *Each $f \in \mathcal{F}_d$ has exactly one root in each terminal component of G.*

Proof. Suppose that $f \in \mathcal{F}_d$. Obviously f must have at least one root in each terminal component since f restricted to a terminal component T is still a rooted forest. Since there are no edges leaving T, any roots in the restricted forest must have been roots in the original forest.

Now, we claim that no two roots of f have a path between them in G. Suppose, for the sake of contradiction, that f has root vertices v, w and that G contains a path p from v to w. Without loss of generality, suppose further that no other vertex on this path is a root of f. Finally, suppose that p passes through vertices $v, x_1, \ldots, x_k, w$. Let f' be the subforest of f consisting of all edges e such that $s(e) = x_i$ for some i. Since no vertex on p is a root, we have removed k edges from f. Now, we claim that the subgraph $f' \cup p$ is also a spanning forest of G. To see this, note that if we add the edges of p to f' one at a time, then at each stage we connect a root vertex to another root vertex and so cannot close a cycle. But now $f' \cup p$ has one less root than f and therefore, one more edge. This contradicts our assumption that f is a maximal spanning forest of G. $\square$

Corollary 2.1.10. *For any $i \in V$, the following are true.*

(i) $\mathcal{F}_d^{\to *i}$ is nonempty if and only if i belongs to a terminal component of G.

(ii) G is strongly connected if and only if $\mathcal{F}_1^{\to *i}$ is nonempty for all $i \in V$.

(iii) G is unilaterally connected if and only if there exists an $i \in V$ such that $\mathcal{F}_1^{\to *i}$ is nonempty.

Lemma 2.1.11. *For any terminal component T of G, the sets $\{\mathcal{F}_d^{\to *i}\}_{i \in T}$ partition $\mathcal{F}_d$.*

Proof. By 2.1.9 each member of $\mathcal{F}_d$ has a unique root in T and so belongs to a unique set in $\{\mathcal{F}_d^{\to *i}\}_{i \in T}$. By Corollary 2.1.10, each of these sets in nonempty. $\square$

2.2 Multivariate Polynomials and Spanning Subgraphs

Let G denote a graph. In this section, we define generating functions associated with the sets defined in the previous section. We will use polynomials defined as weighted sums over sets of rooted spanning subgraphs of G. In general, given a spanning subgraph H of G with root set R, we associate the polynomial weight $\omega^{E(H)}\tau^R$. We will usually abbreviate the edge set $E(H)$ simply by the name of the graph H. Finally, we describe certain sets of spanning functional digraphs and use these to define a polynomial in the variables τ and ω.

Definition 2.2.1. *Recall $\mathcal{D}_G$ denotes the set of spanning functional digraphs of G. Let P be any subset of $\mathcal{D}_G$. Then the (τ, ω)* generating function *of P is given by*

$$M_P(\tau, \omega) = \sum_{f \in P} \mu(f),$$

where $\mu(f) = \omega^f \tau^{R(f)}$.

Note that a single edge e is an example of a functional digraph. If we take P to be the edge set $N^s(v)$ for some vertex v together with the empty graph, then the corresponding (τ, ω) generating function is clearly $\tau_v + \sum_{e \in N^s(v)} \omega^e$.

Lemma 2.2.2. *The polynomial M_P is homogeneous and multiaffine.*

Proof. Every monomial of M_P is $\mu(f)$ for some function digraph f. Since every vertex of G that is not a root of f contributes exactly one edge to f, the monomial $\mu(f)$ must contain n variables in all. Further, since no root edge of f is the source of an edge of f, each variable appearing in a given monomial must be distinct. $\square$

Lemma 2.2.3. *If $H(x_1, \ldots, x_n)$ is a multiaffine polynomial and factors into $H = QR$, then no variable x_i occurs in both Q and R.*

Proof. Suppose, by way of contradiction, that x_i appears in both Q and R. Then QR will have some monomial term with x_i^2 in it. One way to easily see this is to write $Q = \tilde{Q} + x_i Q_0$. Then, since Q is multiaffine, neither $\tilde{Q}$ nor Q_0 contains x_i and Q_0 is not 0. Do the same for R and multiply out QR giving $\tilde{Q}\tilde{R} + x_i(\tilde{Q}R_0 + \tilde{R}Q_0) + x_i^2(Q_0 R_0)$. But $Q_0 R_0$ is not zero, contradicting our assumption that H is multiaffine. $\square$

Definition 2.2.4. *Given any vertex v, let $N_v(\tau, \omega) = \tau_v + \sum_{e \in N^s(v)} \omega_e$. We call this the* source generating function *of v.*

Lemma 2.2.5. *Suppose P and Q are subsets of $\mathcal{D}_G$ satisfying $s(E(f)) \cap s(E(h)) = \emptyset$ for all $f \in P$ and $h \in Q$. Then for any such f, h, the graph $f * h$, whose edges are the edges of both f and h, is also a spanning functional digraph. If $P * Q$ denotes the set $\{f * h\}_{f \in P, h \in Q}$, then*

$$M_{P*Q} = M_P M_Q.$$

Proof. For the first claim, it is sufficient to note that the degree of each vertex of

$f * h$ is 0 or 1. The second claim is a consequence of the fact that each graph in $P * Q$ is the combination of a unique f, h pair. $\square$

This lemma allows us to easily characterize the (τ, ω) generating function for certain collections of spanning functional digraphs. Namely, we characterize those collections of all functional digraphs that share some common subgraph.

Proposition 2.2.6. *The (τ, ω) generating function for $\mathcal{D}$ is given by*

$$M_{\mathcal{D}} = \prod_{v \in V} N_v.$$

More generally, let f be any spanning functional digraph, and let us define $\mathcal{D}(f) = \{h \in \mathcal{D} \mid E(f) \subseteq E(h)\}$. Then

$$M_{\mathcal{D}(f)} = \omega^f \prod_{v \in V - s^{-1}(E(f))} N_v.$$

Proof. This follows by application of Lemma 2.2.5 with $P = \{f\}$ and Q the set of all spanning functional digraphs whose root set contains $s^{-1}(E(f))$. $\square$

The preceding proposition shows that (τ, ω) generating functions for sets of spanning functional digraphs of G may be simple to express and with many non-trivial factors. In what follows, we restrict ourselves to the collections of spanning converging forests discussed in Section 2.1. We will see that the corresponding generating functions are not so well-behaved or easy to characterize in general. This is of course a two sided observation. On the one hand, this means that these *forest polynomials* are capable of capturing nuanced information about the graph G. On the other hand, it also means that they are generally difficult to work with. We emphasize here that our interest in these multivariate forest polynomials is mostly theoretical and our goal is to study the different ways that we might simplify them through the identification of variables. With this in mind, we define the main object of our investigation.

Definition 2.2.7. *The* multivariate forest polynomial *of G is $M_S(\tau, \omega)$ where $S = \mathcal{F}_G$. We will denote this polynomial by $F_G(\tau, \omega)$. In other words,*

$$F_G(\tau, \omega) = \sum_{f \in \mathcal{F}} \omega^f \tau^{R(f)}.$$

We will drop the G subscript when no confusion is likely.

Example 2.2.8. Let G denote the graph with $V = \{1, 2, 3\}$ and edges $e_1 = 1 \to 2$, $e_2 = 2 \to 3$, and $e_3 = 3 \to 2$. Then

$$F_G(\tau, \omega) = \tau_1\tau_2\tau_3 + \omega^{e_3}\tau_1\tau_2 + \omega^{e_2}\tau_1\tau_3 + \omega^{e_1}\tau_2\tau_3 + \omega^{e_1}\omega^{e_3}\tau_2 + \omega^{e_1}\omega^{e_2}\tau_3.$$

Lemma 2.2.9. *Each monomial in the polynomial $F = F_G(\tau, \omega)$ defined above contains at least d type-τ variables, where d is the forest dimension of G.*

Proof. This is immediate from the definition. Each forest $f \in \mathcal{F}$ has at least d roots so that $\mu(f)$ contains at least d τ-type variables. $\square$

Lemma 2.2.10. *For any edge e of the graph G, the variables ω_e and $\tau_{s(e)}$ never occur together in any monomial term of F.*

Proof. Let $f \in \mathcal{F}$. Then e is an edge of f only if $s(e)$ is not a root of f. Therefore $\mu(f)$ cannot contain both $\tau_{s(e)}$ and ω_e. $\square$

Proposition 2.2.11. *If G is strongly connected, then F is irreducible.*

Proof. Assume G is strongly connected and suppose, by way of contradiction, that we have a factorization into $F = P(\tau,\omega)R(\tau,\omega)$. In light of Lemma 2.2.3, we can partition V into two sets A, B and use this to write $P = \tilde{P}(\tau,\omega) + \sum_{v\in A} \tau_v P_v(\omega) + P_\emptyset(\omega)$ and $R = \tilde{R}(\tau,\omega) + \sum_{v\in B} \tau_v R_v(\omega) + R_\emptyset(\omega)$, where each monomial in $\tilde{P}$ and $\tilde{R}$ has at least two type-τ variables in it, while the $P_v, R_v, P_\emptyset$, and $R_\emptyset$ are all free of type-τ variables. Then,

$$F = \tilde{U}(\tau,\omega) + \sum_{v\in A} \tau_v P_v(\omega) R_\emptyset(\omega) + \sum_{v\in B} \tau_v R_v(\omega) P_\emptyset(\omega) + P_\emptyset(\omega) R_\emptyset(\omega)$$

where each monomial of $\tilde{U}$ again contains at least two type-τ variables.

It follows from Lemma 2.2.9 that one of $P_\emptyset(\omega)$ or $R_\emptyset(\omega)$ are equal to zero since $d = 1$ for a strongly connected graph. Without loss of generality, say that $R_\emptyset(\omega) = 0$. In light of Lemma 2.1.10, it now follows that $B = V$. To see this, note that by our construction, the only monomials of F with only one τ variable occur in $\sum_{v\in B} \tau_v R_v(\omega) P_\emptyset(\omega)$.

Now, consider an edge e of G. If ω_e occurs in a monomial of P, then, since we have shown that $\tau_{s(e)}$ occurs in R, we see that the product $\tau_{s(e)}\omega_e$ occurs in some monomial of F. This is impossible by Lemma 2.2.10. Therefore, P is constant and our factorization is trivial. □

Proposition 2.2.11 demonstrates the impracticality of working with the full multivariate polynomial F, except perhaps in the case where each strong component of G is relatively small.

We will also consider some other (τ,ω) generating functions related to G, this time defined over the sets $\mathcal{F}^{A\to *B}$ described in Definition 2.1.2. It would be natural to simply apply the definition of M_S to these sets. However, our goal in looking at these polynomials is to relate them to certain linear algebraic facts about matrices that we will define in the next chapter and this task is made simpler by this adjusted definition.

Definition 2.2.12. *Let $A, B \subseteq V$ with $|A| = |B| = k$. Then, the* multivariate $A \to *B$ forest polynomial *of G is defined to be*

$$F^{A\to *B}(\tau, \omega) = \sum_{f \in \mathcal{F}^{A\to *B}} \omega^f \tau^{R(f)-B}.$$

Obviously, if $S = \mathcal{F}^{A\to *B}$, then $F^{A\to *B} = {}^{M_S}/_{\tau^B}$ so that these polynomials are not substantially different. However, removing the common τ variables from M_S will make some of our later analysis cleaner. Of course we continue the convention from the previous section, letting $F^{\to *A} = F^{A\to *A}$.

The $A \to *B$ forest polynomials possess some of the nice properties of the polynomial F above. In particular, they are still multiaffine and homogeneous. On the other hand, they do not possess the same irreducibility condition as F. To see this, let G be any strongly connected graph with a vertex v with $d^t(v) = 1$. Then, every v rooted forest contains the unique edge e in $t^{-1}(v)$ and therefore, ω^e is a factor of every monomial in $F_G^{\to *v}$.

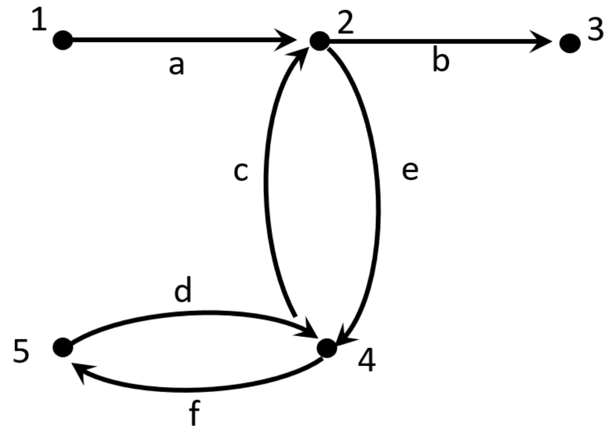


Figure 2.2: The graph used in Example 2.2.13

Example 2.2.13. For the graph shown in Figure 2.2, we have

$$\begin{aligned}
F(\tau,\omega)^{4\to *3} &= \omega_c\omega_b(\tau_1+\omega_a)(\tau_5+\omega_f) \\
F(\tau,\omega)^{5\to *4} &= (\tau_1+\omega_a)(\tau_2+\omega_b+\omega_e) \\
F(\tau,\omega)^{2\to *1} &= 0.
\end{aligned}$$

The full forest polynomial is given by

$$\begin{aligned}
F(\tau,\omega) &= \tau_3(\tau_1+\omega_a)(\tau_2\tau_4\tau_5+\omega_d\tau_2\tau_4+(\omega_c+\omega_f)\tau_2\tau_5+(\omega_b+\omega_e)\tau_4\tau_5 \\
&\quad +\omega_c\omega_d\tau_2+(\omega_d\omega_e+\omega_b\omega_d)\tau_4+(\omega_b\omega_c+\omega_b\omega_f+\omega_e\omega_f)\tau_5 \\
&\quad +\omega_b\omega_c\omega_d)
\end{aligned}$$

In the next chapter, we will see that the $A \to *B$ polynomials are, in general, not easily obtainable by linear algebraic means. However, when A and B are singleton sets $\{i\}$ and $\{j\}$, the polynomials $F^{i\to *j}(\tau,\omega)$ can be written as the determinant of a matrix. Indeed, these $i \to *j$ forest polynomials will become a main object of our attention due to their linear algebraic connections. To bring this out, note that they can naturally be arranged into a matrix.

Definition 2.2.14. *The* converging forest matrix *of G is the matrix $Q(\tau, \omega)$ whose ij entry is $F^{i \to *j}(\tau, \omega)$.*

We close this section by noting a property of the matrix Q.

Proposition 2.2.15. *Viewing τ as a column vector, we have*

$$Q(\tau, \omega)\tau = F(\tau, \omega)\mathbb{1}.$$

Proof. This follows from Lemma 2.1.6. We compute F as below.

$$\begin{aligned} F(\tau, \omega) &= \sum_{f \in \mathcal{F}} \omega^f \tau^{R(f)} \\ &= \sum_{j=1}^{n} \tau_j \sum_{f \in \mathcal{F}^{i \to *j}} \omega^f \tau^{R(f) - \{j\}} \\ &= \sum_{j=1}^{n} \tau_j F^{i \to *j}(\tau, \omega). \end{aligned}$$

Of course, the last line is the ith entry of $Q(\tau, \omega)\tau$. $\square$

2.3 Specializations of Multivariate Forest Polynomials

One natural response to the irreducibility of $F_G(\tau, \omega)$ for strongly connected graphs is to look for specializations of the τ and ω variables that might yield a reducible polynomial. In fact, the graph theory literature has a large body of work examining properties of the particular specialized forest polynomials and their associated matrices when $\tau = t\mathbb{1}$. This approach to specializing variables further connects the study of forest polynomials to the study of graph partitions, especially the theory of *equitable partitions* [26] We will have more to say about this later. For the moment, we will define some important specialized forest polynomials.

To begin, let us organize our forest polynomial expressions by grouping the monomials sharing a particular set of τ variables. Thus

$$F(\tau, \omega) = \sum_{S \subseteq V} R_S(\omega)\tau^S \tag{2.1}$$

and

$$F^{A \to *B}(\tau, \omega) = \sum_{S \subseteq V - B} R^*_{S \cup B}(\omega)\tau^S, \tag{2.2}$$

where $R_S(\omega)$ and $R^*_S(\omega)$ are sums of ω^f for all f in $\mathcal{F}$ and $\mathcal{F}^{A \to *B}$ respectively such that $R(f) = S$. Note that the ω polynomial $\sum_{v \in V} R_v$ is the well-known Kirchoff polynomial which has been of some recent interest, for example [42].

For a graph G, we let Π be a partition of V into k parts and Ψ be a partition of E into l parts. Then, we define variables τ and ω, this time indexed over the cells of Π and Ψ respectively. This allows us to evaluate F_G at $\tau^* = \Pi\tau$ and $\omega^* = \Psi\omega$. This has the effect of identifying the variable τ_v for each $v \in \Pi_i$ with the variable τ_i and similarly for the ω variables. We will see that by choosing the partitions Π and Ψ carefully, we are able to collapse the often unwieldy F_G into something more manageable.

Definition 2.3.1. *Suppose that* Π *and* Ψ *partition* V *and* E *respectively. Let* τ *and* ω *be indexed over* Π *and* Ψ *as described above. Then we define the* Π, Ψ partitioned forest polynomial *to be* $F_G(\Pi\tau, \Psi\omega)$.

Note that these partitioned forest generating functions still count something. Given $\mu \in \mathbb{Z}^k$, with $0 \leq \mu_i \leq |\Pi_i|$, and $\eta \in \mathbb{Z}^l$, with $0 \leq \eta_i \leq |\Psi_i|$, the integer coefficient of $\omega^\eta \tau^\mu$ counts the number of forests of G with exactly μ_i roots in Π_i and η_i edges in Ψ_i.

There are a few broad classes of partitioned forest polynomials that appear in the literature.

Definition 2.3.2. *The* univariate forest polynomials *of G result from setting $\tau = t\mathbb{1}$. or equivalently, taking Π to be the singleton partition and Ψ to be the trivial one. Here we collect the different R_S terms with a fixed size of S recovering ω polynomials for the various k forest sets defined in Definition 2.1.7. Thus, 2.1 becomes*

$$F(t\mathbb{1}, \omega) = \sum_{i=d}^{n} F_k(\omega) t^i,$$

where $F_k(\omega) = \sum_{f \in \mathcal{F}_k} \omega^f$ and d is the forest dimension of G. We abbreviate $F(t\mathbb{1}, \omega)$ by $F(t, \omega)$ or $F(t)$ when this is clear from context. Equation 2.2 becomes

$$F^{A \to *B}(t\mathbb{1}, \omega) = \sum_{i=d-1}^{n-1} F_{i+1}^{A \to *B}(\omega) t^i$$

where we define $F_i^{A \to *B}$ similarly and make the same abbreviations.

Finally, the same specialization applied to the matrix $Q(\tau, \omega)$ defined in 2.2.14 gives the univariate forest matrix *which we abbreviate by $Q(t, \omega)$ or $Q(t)$. Expanding* *this matrix around its t variable, we have*

$$Q(t) = \sum_{i=d-1}^{n-1} Q_{i+1}(\omega)t^i$$

where $Q_k(\omega)$ is the k forest matrix *of G whose ij entry is $F_k^{i\to *j}$.*

Finally, any of our (multivariate or univariate) forest polynomials and matrices become unweighted *if we set $\omega = \mathbb{1}$. In this case, the ω polynomials $R_S, R_S^*, F_k, F_k^{A\to *B}$ all become integers counting the forests in the sets that they are defined over.*

The unweighted univariate forest polynomials and matrix were first defined in the context of directed graphs by Chebotarev and Agaev [10], although we see special cases such as the Kirchoff polynomials in the work of Tutte and others, especially regarding undirected graphs.

3 The Matrix Forest Theorem and its Generalizations

In this chapter, our goal is to derive a generalization of the Matrix Tree Theorem that captures the determinental nature of the forest polynomials, including $F_G(\tau,\omega)$ and $F_G^{i\to *j}(\tau,\omega)$ described above. We will then illustrate how our generalization provides combinatorial interpretations of the matrix polynomials associated with other matrices used in the current literature to describe graphs.

3.1 The Generalized Laplacian and the Matrix Forest Theorems

In this section, we prove the Matrix-Forest Theorem and the All-Minors version that is appropriate in this setting. For the remainder of this section, we let G be a fixed graph.

Definition 3.1.1. *Let G have source and target incidence matrices S and T, respectively. The matrix*

$$L(\tau, \omega) = D(\tau) + S^T D(\omega)(S - T)$$

is the root and edge parameterized Laplacian matrix *of G.*

We will hereafter refer to the above matrix simply as the Laplacian matrix of G. Note that there are numerous matrices in the literature that claim this title.

Remark 3.1.2. Observe that if $i \neq j$, then $L(\tau, \omega)_{ij} = -\omega_e$ if there is an edge of G with $s(e) = i$ and $t(e) = j$. Of course, $L(\tau, \omega)_{ij} = 0$ otherwise. If $i = j$, then $L(\tau, \omega)_{ii} = \tau_i + \sum_{e \in s^{-1}(i)} \omega_e$.

Definition 3.1.3. *Let G be a graph and π be a permutation on V. Then the* cyclic part of π in G *is the set $C(G, \pi)$ containing all edges of the form $i \to \pi(i)$ for some $i \in V$ such that $\pi(i) \neq i$. Similarly, the* fixed part of π in G *is the complementary set $X(G, \pi)$ of vertices of G that are fixed by π.*

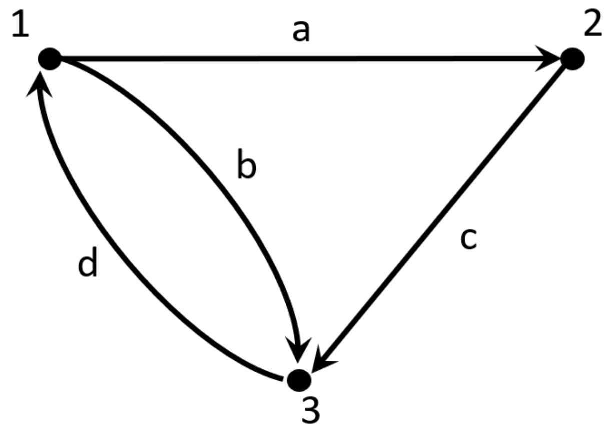


Figure 3.3: The graph used in Example 3.1.4

Example 3.1.4. Let G be the graph depicted in figure 3.3. Then,

$$S = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} T = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} L(\tau, \omega) = \begin{bmatrix} \tau_1 + \omega_a + \omega_b & -\omega_a & -\omega_b \\ 0 & \tau_2 + \omega_c & -\omega_c \\ -\omega_d & 0 & \tau_3 + \omega_d \end{bmatrix}.$$

We are now ready to state and prove the Matrix Forest Theorem.

Theorem 3.1.5. *[Matrix Forest Theorem] For any graph G,*

$$\det(L(\tau, \omega)) = F_G(\tau, \omega).$$

Proof. Let $L = L(\tau, \omega)$. From the definition of the determinant,

$$\det(L) = \sum_{\pi \in \mathcal{S}_n} \operatorname{sgn}(\pi) \prod_{i=1}^{n} L_{i\pi(i)}.$$

Setting $f_\pi(\tau,\omega) = \prod_{i=1}^n L_{i\pi(i)}$, we note that f_π is non-zero if and only if every vertex not fixed by π is the source of an edge in $C(G,\pi)$. This is equivalent to the condition that $|C(G,\pi)| = n - |\text{fix}(\pi)|$. Applying Definition 3.1.1 and Lemma 2.2.6 we find that

$$\begin{aligned} f_\pi &= (-1)^{|C(G,\pi)|}\omega^{C(G,\pi)} \prod_{i\in\text{fix}(\pi)} \left(\tau_i + \sum_{e\in N^s(i)} \omega_e\right) \\ &= (-1)^{|C(G,\pi)|} \sum_{f\in D(\pi)} \omega^f \tau^{R(f)} \end{aligned}$$

where, by Lemma 2.2.5, $D(\pi)$ is the set of all spanning functional digraphs f of G containing $C(G,\pi)$ as a subgraph.

Now since $|C(G,\pi)| = n - |\text{fix}(\pi)|$, we can write $\text{sgn}(\pi) = (-1)^{|C(G,\pi)|+|\text{cyc}(\pi)|}$. Applying this to our determinant sum, we have

$$\begin{aligned} \det(L) &= \sum_{\pi\in\mathcal{S}_n} \text{sgn}(\pi) f_\pi \\ &= \sum_{\pi\in\mathcal{S}_n} (-1)^{|C(G,\pi)|+|\text{cyc}(\pi)|}(-1)^{|C(G,\pi)|} \sum_{f\in D(\pi)} \omega^f \tau^{R(f)} \\ &= \sum_{\pi\in\mathcal{S}_n} \sum_{f\in D(\pi)} (-1)^{|\text{cyc}(\pi)|} \omega^f \tau^{R(f)}. \end{aligned}$$

Now, we would like to exchange the summation signs in the last equation. For a given $f \in D(G)$, we see that $f \in D(\pi)$ just in case each cycle of π corresponds to a cycle in f. Thus, each choice of subset S of the cycles in $C(f)$ determines exactly one π with $f \in D(\pi)$. Of course, for any such π, we have that $|\text{cyc}(\pi)| = |S|$. Recalling that, for any non-empty set T, the sum $\sum_{S\subseteq T}(-1)^{|S|}=0$, we see that

$$\begin{aligned}\sum_{\pi\in\mathcal{S}_n}\sum_{f\in F_\pi}(-1)^{|\mathrm{cyc}(\pi)|}\omega^f\tau^{R(f)} &= \sum_{f\in D(G)}\left(\sum_{S\subseteq C(f)}(-1)^{|S|}\right)\omega^f\tau^{R(f)}\\ &= \sum_{f\in\mathcal{F}_{\vec{G}}^*}\omega^f\tau^{R(f)}.\end{aligned}$$

The last equation follows because the only elements f of $D(G)$ with $C(f)=\emptyset$ are the converging spanning forests of G. □

While there are many other proofs of this theorem such as [17, 37], this particular proof has the advantage that it can easily be adjusted to prove the more general "all-minors" version stated below. Although we will not use the full generality of the all-minors theorem in what follows, we include this proof in part because it greatly simplifies existing proofs of the all-minors theorem and simultaneously generalizes the theorem to include a root parameterization. We encourage the reader to compare [9] to see the efficiency afforded by our approach.

In what follows, we let $A, B\subseteq V(G)$ with $|A|=|B|=k$.

Definition 3.1.6. *Let A and B be any two ordered k-sets of vertices. The sign of A and B is given by*

$$\mathrm{sgn}(A,B)=(-1)^{\sum_{i=1}^k a_i+b_i}.$$

Definition 3.1.7. *Given graph G with Laplacian matrix $L(\tau,\omega)$ and ordered k-sets $A, B\subseteq V$, we define an $n\times n$ matrix $L^{A,B}(\tau,\omega)$ to have i,j entry equal to $L(\tau,\omega)_{ij}$ if $i\notin B$ and $j\notin A$, 1 if $i=b_l$ and $j=a_l$ for some l, and 0 otherwise.*

Lemma 3.1.8. *Given graph G with Laplacian matrix $L(\tau,\omega)$ and ordered k-sets*

$A, B \subseteq V$, the matrix $L^{A,B}$ satisfies

$$\det L(\tau, \omega)_{[B,A]} = \operatorname{sgn}(A, B) \det L^{A,B}(\tau, \omega).$$

Proof. Since a_l, b_l is the only nonzero entry in the a_l row of $L^{A,B}$, this follows from applying Laplace expansion to each row a_l for $l = 1, \dots, k$. $\square$

Definition 3.1.9. *Let $\kappa = \{i \in [k] \mid a_i \neq b_i\}$ denote the set of indices at which the sets A and B differ with $|\kappa| = k^*$. Then, let $\bar{\kappa} = [k] - \kappa$.*

Theorem 3.1.10. *[All-Minors Matrix Forest Theorem] Let A and B be subsets of V each of size k. Then*

$$\det L(\tau, \omega)_{[B,A]} = \operatorname{sgn}(A, B) \sum_{f \in \mathcal{F}_G^{A \to * B}} \operatorname{sgn}(\pi_f) \omega^f \tau^{R(f) - B},$$

where π_f is as in Definition 2.1.2.

Proof. Using Lemma 3.1.8, we can evaluate $\det L^{A,B}(\tau, \omega)$, which is naturally a sum over permutations of $V(G)$ as in the previous theorem. As before, we calculate the contribution of each permutation $\pi \in \mathcal{S}_n$ as $f_\pi(\tau, \omega) = \prod_{v \in V(G)} L^{A,B}_{v, \pi(v)}$. The structure of $L^{A,B}$ now implies that $f_\pi = 0$ unless $\pi(b_i) = a_i$ for each $i \in [k]$ and every vertex in $V - B$ that is not fixed by π is the source of an edge in $C(G, \pi)$.

As a result, we can conclude that f_π is non-zero just in case $|C(G, \pi) - s^{-1}(B)| = n - |\operatorname{fix}(\pi)| - k^*$. In other words, each vertex that is neither a member of B nor fixed by π is the source of an edge in $C(G, \pi)$.

In the proof of the previous theorem, each $(v, \pi(v))$ pair from each cycle of π

formed an edge in G and contributed a factor of -1 times an edge variable to f_π. In our current case, all pairs of the form $(b_i, \pi(b_i))$ contribute only a factor of 1, and, in fact, may or may not even correspond to an edge in G. Similarly, if $v \in X(G,\pi) \cap B$, then it must be the case that $v = b_i = a_i$ for some i, and this vertex again contributes a factor of 1 to f_π.

For $v \in \text{cyc}(\pi) - B$, the contribution of the pair $(v, \pi(v))$ is simply $-\omega_{v \to \pi(v)}$. On the other hand, if $v \in X(G,\pi) - B$, we have the same contribution as in the previous theorem. It follows that

$$f_\pi = (-1)^{(n - |\text{fix}(\pi)| - k^*)} \omega^{(C(G,\pi) - s^{-1}(B))} \left(\prod_{v \in X(G,\pi) - B} \tau_v + \sum_{e \in N_G^s(v)} \omega_e \right).$$

As before, f_π is a scalar multiple of a subgraph generating function over G

$$f_\pi = (-1)^{(n - |\text{fix}(\pi)| - k^*)} \sum_{f \in \mathcal{D}_\pi^{A \to *B}} \omega^f \tau^{R(f) - B}$$

where the set $\mathcal{D}_\pi^{A \to *B}$ contains all spanning functional digraphs of G with $C(G,\pi) - s^{-1}(B)$ as a subgraph. The notation refers to the fact that any cycle of π containing, say l pairs b_i, a_i corresponds to a collection of l paths in f with each path rooted in an element of B. For example, if cycle C consists of vertices $b_1, a_1, x, \ldots, y, b_2, a_2, z, \ldots, w$, then each f in $\mathcal{D}_\pi^{A \to *B}$ contains the paths $a_1, x, \ldots, y, b_2$ and $a_2, z, \ldots, w, b_1$ with b_1 and b_2 as roots. Since each b_i, a_i pair appears in exactly one cycle, π determines a rooted $A \to *B$ path system. $\mathcal{D}_\pi^{A \to *B}$ is the set of all spanning functional digraphs of G containing the $A \to *B$ path system determined by π.

Now we compute as follows:

$$\begin{aligned}
\det\left(L^{A,B}\right) &= \sum_{\pi\in\mathcal{S}_n} \operatorname{sgn}(\pi) f_\pi \\
&= \sum_{\pi\in\mathcal{S}_n} (-1)^{(n-|\mathrm{cyc}(\pi)|-|\mathrm{fix}(\pi)|)}(-1)^{(n-|\mathrm{fix}(\pi)|-k^*)} \sum_{f\in\mathcal{D}_\pi^{A\to *B}} \omega^f \tau^{R(f)-B} \\
&= \sum_{\pi\in\mathcal{S}_n} \sum_{f\in\mathcal{D}_\pi^{A\to *B}} (-1)^{|\mathrm{cyc}(\pi)|+k^*} \omega^f \tau^{R(f)-B}.
\end{aligned}$$

As in the previous theorem, we would like to exchange the summation signs. As we run through all permutations π, it is easy to see that we run through every possible $A \to *B$ path system in G. If we fix an $f \in \mathcal{D}^{A\to *B}$, then f similarly prescribes an $A \to *B$ path system.

The monomial $\omega^f \tau^{R(f)-B}$, associated with f, appears as a monomial in f_π whenever π contains the cycles prescribed, as in the previous paragraph, by the $A \to *B$ path system of f. Every other cycle of π must also correspond to some cycle of f. Now, just like in the previous proof, the ω variable for an edge with source i in the forest part of f must come from the factor $(\tau_i + \sum_{e\in N_G^s(i)} \omega_e)$ in f_π. This means that i must be a fixed point of π. On the other hand, any cycle c in f could arise in π either as a cycle or as a term in

$$\prod_{i\in V(c)} \left(\tau_i + \sum_{e\in N_G^s(i)} \omega_e\right)$$

corresponding to a fixed point. We can therefore determine a π whose f_π makes a monomial contribution to f by choosing any subset S of $C(f)$ and combining these with the AB-cycles determined by f. We call this permutation π_S and note that, if c_f is the number of AB cycles determined by f, then π_S has a total of $|S| + c_f$ cycles.

It follows that the coefficient of $\omega^f \tau^{R(f)-B}$ in f_{π_S} is $(-1)^{|S|+c_f+k*}$.

Thus, by similar reasoning to the proof of the previous theorem, the sum over functional digraphs collapses into a sum over spanning forests.

$$\begin{aligned} \sum_{\pi \in \mathcal{S}_n} \sum_{f \in \mathcal{D}_\pi^{A \to * B}} (-1)^{|\mathrm{cyc}(\pi)|+k^*} \omega^f \tau^{R(f)-B} &= \\ \sum_{f \in \mathcal{D}^{A \to * B}(G)} \left(\sum_{S \subseteq C(f)} (-1)^{|S|} \right) (-1)^{c_f+k^*} \omega^f \tau^{R(f)-B} &= \sum_{f \in \mathcal{F}_G^{A \to * B}} (-1)^{c_f+k^*} \omega^f \tau^{R(f)-B} \end{aligned}$$

Finally, we have that $(-1)^{c_f+k^*} = \operatorname{sgn}(\pi_f)$. To see this, note that there are c_f cycles and $k - k^*$ fixed points in π_f. $\square$

If $A = B$, then we are guaranteed that $\operatorname{sgn}(\pi_f)$ and $\operatorname{sgn}(A, B)$ are positive, giving us a straightforward way to relate principal minors of $L(\tau, \omega)$ to forest polynomials.

Corollary 3.1.11. *Let A be any subset of V. Then*

$$\det L(\tau, \omega)_{[A,A]} = F^{\to * A}(\tau, \omega)$$

When $A \neq B$, Theorem 3.1.10 does not yield a straightforward way to recover the generating function for $F^{A \to * B}$ in general, due to the $\operatorname{sgn}(\pi_{f,A,B})$ term. However, if $|A \Delta B| = 1$, then $\pi_{f,A,B}$ has the same sign for all f.

Corollary 3.1.12. *Let A and B be subsets of V each of size k. If $|A \Delta B| = 1$, then*

$$\det L(\tau, \omega)_{[A,B]} = \pm F^{A \to * B}(\tau, \omega).$$

Finally, if A and B are singletons, then we get our important special case that we will put to much use in the remainder of this thesis.

Corollary 3.1.13. *If G is a graph with Laplacian matrix $L(\tau, \omega)$, then*

$$Q(\tau, \omega) = \operatorname{adj}(L(\tau, \omega)).$$

Proof. The ij entry of $\operatorname{adj}(L(\tau, \omega))$ is $(-1)^{i+j} \det L(\tau, \omega)_{[B,A]}$, where $A = \{i\}$ and $B = \{j\}$. Applying Theorem 3.1.10, we note that if A and B only contain one element, then $\operatorname{sgn}(\pi_f) = 1$ for any $f \in \mathcal{F}_G^{A \to * B}$. Therefore, the ij entry of $\operatorname{adj}(L(\tau, \omega))$ is

$$(-1)^{2(i+j)} \sum_{f \in \mathcal{F}_G^{A \to * B}} \omega^f \tau^{R(f)-B} = F_G^{i \to * j}(\tau, \omega).$$

$\square$

Corollary 3.1.14. *If G is a graph with Laplacian matrix $L(\tau, \omega)$, then*

$$L(\tau, \omega) Q(\tau, \omega) = F(\tau, \omega) I.$$

We might call the matrix $L(\tau, \omega)$ the generalized source Laplacian matrix for G. This matrix is something of a wellspring for algebraic graph theory in that it captures a wide variety of graph matrices under a single parametric family. Our approach to the matrix tree theorem is essentially due to Aigner [1]. He defines a signed involutional map on the maximal spanning functional digraphs of an undirected graph by using a graph transformation and the lemma of Gessel, Viennot, and Lindstrom

[25]. Our contribution here is to extend this approach, now in the setting of directed graphs, to the root parameterized generating functions defined above. In the course of generalizing this argument, we found that neither the graph transformation nor the aforementioned lemma was actually needed, because the proof relies solely on the cycle argument that we gave above. One can view this argument as essentially employing an involution that maps an appropriate functional digraph f to its weight with a sign determined by the number of cycles that f possesses.

Since Aigner's original argument applies to the undirected matrix tree theorem, it is worth pointing out how this theorem relates to our directed version. In particular, the original theorem counts unweighted spanning trees of an undirected graph, while our directed version counts rooted converging trees of the associated directed graph.

To see how these relate, we can define a simple many-to-one map that associates each rooted tree with an undirected spanning tree. For this, we simply ignore the root and the direction of each edge. In an undirected graph, there are exactly n directed converging trees mapping to each undirected tree and therefore there is an n to one mapping from directed to undirected trees. This is not the case for forests and undirected spanning forests are particularly hard to get at by linear algebraic methods.

Other root-parameterized approaches to the Matrix Tree Theorem have been considered, for example, in [10, 38]. However, none of these have, to our knowledge, been applied to the all-minors version and particularly to our setting using τ instead of $t\mathbb{1}$. The all-minors version of the matrix tree theorem is due to [9]. We view our proof as a substantial simplification of these results that also unifies them with a number of others appearing in the graph theory literature. One advantage of our approach is that it is general enough to apply to a broad range of matrices. We will close this section by listing a few of these matrices and showing how they may be seen as evaluations of $L_G(\tau, \omega)$. These expressions often appear without a root parametrization or with a univariate root parametrization, however they can be naturally generalized to include this.

To begin, we will let d be defined by $d_v = d_G^s(v)$. Although the observation is somewhat trivial, we can write the adjacency matrix as an evaluation of L given by $A = L(-d, -\omega)$. Expanding and simplifying the associated forest generating function $F(t-d, -\omega)$, we recover the well known signed weighted generating function of the *linear digraphs* [6], consisting of all functional digraphs whose forest part is empty. One consequence of this is that the so-called Ihara Zeta Function, defined for directed graphs by $Z_G(t) = \det(1 - tA)^{-1}$ can be viewed as the inverse of a forest polynomial [50, 35].

Another matrix of interest is the *signless Laplacian matrix*, defined as $L_G^+ = S^T D(\omega)(S+T)$. In our notation, this matrix is given by $L_G^+ = L_G(2d, -\omega)$. Thus, the analysis in [53, 15] can be understood in terms of forest polynomials as well. Even less directly, the *voltage Laplacian matrix*, defined in [13] is given by

$$L_G(d^*, \omega^*)$$

where $(d^*)_i = \sum_{e \in N^s(i)} (1 - v(e))\omega(e)$ and $(\omega^*)_e = v(e)\omega(e)$. Note that in this case, the entries $v(e)$ actually lie in the group algebra $\mathcal{G}[V]$ for some commutative group G however, since this is still a commutative ring, our result applies.

Finally, there has been some recent interest [16, 46, 29, 48] in parametric families of graphs that include some of $L(0, \mathbb{1})$, A, or L^+. One example from [41] studies the characteristic polynomial of the so called A_α matrix of a graph given by $A_\alpha = \alpha D + (1-\alpha)A$. In our terminology, this amount to evaluating the forest polynomial $F(t\mathbb{1} - (2\alpha - 1)d, (\alpha - 1)\omega)$.

3.2 Examples of Unweighted Univariate Forest Polynomials

In this section, we compute the forest generating functions for several well-known graph families that we will analyze in more detail later. In each case, the identity follows from well-known results. We give citations except in the case of K_n which follows immediately from the form of the Laplacian matrix form $L_{K_n}(0, \mathbb{1}) = nI - J$.

Proposition 3.2.1. *The forest generating function for the complete graph is given by*

$$F_{K_n}(t) = t(t+n)^{n-1}.$$

Proposition 3.2.2. *The forest generating function for the complete multipartite graph is given by*

$$F_{K_{n_1,\ldots,n_k}}(t) = t(t+n)^{k-1} \prod_{i=1}^{k} (t + n - n_i)^{n_i - 1}.$$

This result can be found, for example, in [34]

Proposition 3.2.3. *The forest generating function for the de Bruijn graph is given by*

$$F_{B(n,k)} = t(t+k)^{k^n - 1}.$$

This result can be found, for example, in [18].

Proposition 3.2.4. *If Q_n the hypercube graph, then he forest generating function for $Q_n$is given by*

$$F_{Q_n}(t) = \prod_{i=0}^{n} (t + 2i)^{\binom{n}{i}}.$$

More generally, we have the following.

Proposition 3.2.5. *Let $k, n_1, \ldots, n_k$ be fixed and let $G = K_{n_1} \times \cdots \times K_{n_k}$. Then, for $S \subseteq [k]$, we set $n_S = \sum i \in Sn_i$. With this,*

$$F_G(t) = \prod_{S \subseteq [k]} (t + n_S)^{\mu(S)}$$

where $\mu(S) = \prod_{i \in S}(i - 1)$.

The argument for Q_n given in [4] can also be extended to the more general case. This uses the simple fact that the eigenvalues of matrix $A \oplus B$ are exactly the sums $\lambda + \mu$ where λ, μ are eigenvalues of matrices A and B respectively. Using this fact and 3.2.1 the proposition above follows.

3.3 The Weighted Univariate Forest Matrix and Laplacian Eigenvectors

An interesting consequence of the All-Minors Matrix Forest Theorem, noted first by Chebatorev [10], is a combinatorial description of the matrix algebra generated by the (un-parameterized, unweighted) Laplacian matrix $L(0, \mathbb{1})$. Here we will study a generalization of this setting. We take $\tau = t\mathbb{1}$ but leave ω unspecialized. In this setting, each k-forest matrix Q_k belongs to the matrix algebra generated by $L = L(0, \omega)$. We will analyze some linear algebraic properties of this matrix algebra as well as the matrix L itself. In the interest of brevity, we will abbreviate $Q(t\mathbb{1}, \omega) = Q(t)$ and $F(t\mathbb{1}, \omega) = F(t)$ so that L, F, and each Q_i all contain ω variables.

Definition 3.3.1. *If X is a square matrix, then we denote the matrix algebra generated by X and I as $\mathfrak{M}(X)$.*

Definition 3.3.2. *If G is a graph with adjacency matrix A and Laplacian matrix L, we define the* adjacency algebra *of G to be $\mathfrak{M}(A)$ and the Laplacian algebra to be $\mathfrak{M}(L)$.*

Proposition 3.3.3. *For any graph G, we have $\mathfrak{M}(A) = \mathfrak{M}(L)$ if and only if G is a d-source regular graph for some d.*

Proof. If G is a d-regular graph, then $L = dI - A$ and so obviously $\mathfrak{M}(A) = \mathfrak{M}(L)$. On the other hand, if $L \in \mathfrak{M}(A)$, then since $L\mathbb{1} = 0$ and since every element of $\mathfrak{M}(A)$ is simultaneously diagonalizable, we conclude that $\mathbb{1}$ must be an eigenvector of A. This implies that A is d- source regular. [27] □

Lemma 3.3.4. *If G is a graph with forest dimension d, then $\mathfrak{M}(L)$ is generated by the set of forest matrices*

$$\{Q_{G,k} \mid k = 1, \ldots, n-d\}.$$

In general, this set spans the matrix algebra but is not a basis for this algebra. To see this, one need only consider a strongly connected graph with many repeated Laplacian eigenvalues. To illustrate this, let us return to the complete graph its unweighted Laplacian matrix.

Example 3.3.5. Let $G = K_n$ the bi-directed complete graph on n vertices. Then, $L(0, \mathbb{1}) = nI - J$ and

$$\begin{aligned} Q(t) &= (t+n)^{n-2}(tI + J) \\ &= (t+n)^{n-2}(t+1)I + (t+n)^{n-2}(J - I). \end{aligned}$$

Therefore, we have $Q_k = n^{n-1-k}\left(n\binom{n-2}{k-2}I + \binom{n-2}{k-1}J\right)$. [3]

Here, $Q(t) = \operatorname{adj} L(t\mathbb{1}, \mathbb{1})$. One way to derive this result is to simply verify that $L(t\mathbb{1}, \mathbb{1})Q(t) = F_{K_n}(t)I$ which holds because $L(t\mathbb{1}, \mathbb{1}) = (t+n)I - J$ and $((t+n)I - J)(tI + J) = t(t+n)I$. It is easy to see from the form of L_{k_n} that $\mathfrak{M}(K_n)$ is generated by the two matrices I and J.

Forest polynomials are intimately connected with the eigenvectors of the matrix L. Indeed, forest polynomials give us a way to approach Laplacian eigenvector techniques in algebraic graph theory combinatorially. This is in contrast to the usual Markov-chain inspired approaches although we will see later that these methods have some overlap. One immediate challenge in applying these matrices in practice is that the coefficients of their entries grow extremely fast as the size of the graph grows. However, $Q(t)$ does have some special structure in general.

Proposition 3.3.6. *The forest matrix satisfies*

$$Q(t)\mathbb{1} = \frac{F(t)}{t}\mathbb{1}.$$

[3]Note that this formula remains valid when $n = 1$. In this case $tI + J = t + 1$ is a scalar so that $Q(t) = 1$ corresponding to the unique empty forest that spans K_1.

Proof. Take Proposition 2.2.15 and set $\tau = t\mathbb{1}$. $\square$

In the current setting, Corollary 3.1.14 specializes to the following.

Proposition 3.3.7. *The forest matrix $Q(t)$ and $L(t,\omega)$ are related by*

$$L(t,\omega)Q(t) = F(t)I.$$

Using Definition 2.3.2 and comparing coefficients in 3.3.7, we can describe the action of L on Q_k.

Corollary 3.3.8. *The Laplacian L is annihilated by the matrix of maximal spanning forests. That is,*

$$LQ_d = 0.$$

Proof. Setting $t = 0$ is 3.3.7 $\square$

Corollary 3.3.9. *For $0 < k < n$, we have*

$$LQ_{k+1} = f_k I - Q_k.$$

Proof. First, we use the definition to write $Q(t) = \sum_{k=0}^{n-1} Q_{k+1}t^k$. Inserting this into Proposition 3.3.7 and rearranging yields

$$LQ_1 + \left(\sum_{k=1}^{n-1}(Q_k + LQ_{k+1} - f_k I)t^k\right) + (Q_n - f_n I)t^n = 0.$$

Recalling that $LQ_1 = 0$, $Q_n = I$ and $f_n = 1$, the corollary follows. □

Note that this corollary is equivalent to the LeVerrier-Faddeev Algorithm [28].

Proposition 3.3.7 also allows us to draw a simple connection between the eigenvectors of the Laplacian matrix L and the forest matrix $Q(t)$.

Corollary 3.3.10. *If λ is an eigenvalue of L, then each column of $Q(-\lambda)$ is an eigenvector of L.*

Proof. Rearranging Proposition 3.3.7 we see that $LQ(t) = F(t)I - tQ(t)$. Note that $F(-\lambda) = 0$ from the definition of $F(t)$. Thus, if we take $t = -\lambda$, then $LQ(-\lambda) = \lambda Q(-\lambda)$. □ This corollary may be of interest as a way to give combinatorial insight into various spectral partitioning algorithms.

Note that G has spanning trees if and only if $Q_1 \neq 0$. Of course, by Lemma 2.1.10, this occurs exactly when G is unilaterally connected. The generating functions for these maximal spanning forests are called the Kirchoff Polynomials [55]. As these objects are of considerable interest, especially in chemical reaction network theory, we note that in our notation, the Kirchoff polynomial of a unilaterally connected graph G is equal to $\sum_{v \in V} F_1^{\rightarrow *v}(0, \omega)$.

One nice application of the theory that we have developed so far is a combinatorial description of the 0 Laplacian eigenvectors of G. This has been proved in a few different places. For example, the authors of [8] employ the theory of Markov chains to derive a similar result, but stated in terms of the steady state solution of a Markov process defined on G. For a discussion of the connection between spanning forests and Markov processes, we recommend [47]. In addition, the authors in [44] present an argument using the the all-minors matrix tree theorem similar to that given below and relate this result to certain dynamical systems defined on V.

Theorem 3.3.11. *Let G be a graph with forest dimension d and terminal components $T_1, \ldots, T_d$. Then*

$$Q_d = \sum_{k=1}^{d} \bar{\gamma}_k \gamma_k^T,$$

where γ_k and $\bar{\gamma}_k$ are both $n \times 1$ column vectors with

$$(\gamma_k)_v = \delta_{v \in T_k} F_d^{\to * v},$$

and

$$(\bar{\gamma}_k)_v = \frac{\sum_{w \in T_k} F_d^{v \to * w}}{F_d^{\to *}}.$$

Proof. Looking at the ij entry of $\sum_{k=1}^{d} \bar{\gamma}_k \gamma_k^T$, we have

$$\sum_{k=1}^{d} (\bar{\gamma}_k)_i (\gamma_k)_j = \frac{\sum_{k=1}^{d} \delta_{j \in T_k} F^{\to * j} \sum_{w \in T_k} F_d^{i \to * w}}{F_d^{\to *}},$$

so that the theorem follows if

$$\sum_{k=1}^{d} \delta_{j \in T_k} F_d^{\to * j} \sum_{w \in T_k} F_d^{i \to * w} = F_d^{i \to * j} F_d^{\to *}.$$

To see this, first note that both sides are 0 if j does not belong to a terminal component of G. Let us then suppose that $j \in T_k$ so that the above equation reduces to

$$F_d^{\to * j} \sum_{w \in T_k} F_d^{i \to * w} = F_d^{i \to * j} F_d. \tag{3.3}$$

Now, both sides of this equation are polynomials in ω that can each be written as sums over certain pairs of forests with a forest pair (f_1, f_2) contributing $\omega^{f_1} \omega^{f_2}$. To express these sums explicitly, let us define $\bar{\mathcal{F}}$ to be the subset of $\mathcal{F}_d$ consisting of all forests containing a path from i to a root not contained in T_k as well as $\bar{\mathcal{F}}^{(j)}$ to be $F_d^{\to *j} \cap \bar{\mathcal{F}}$. Then, we have the disjoint sum of sets

$$\mathcal{F}_d^{\to *j} = \mathcal{F}_d^{i \to *j} + \bar{\mathcal{F}}^{(j)}$$

as well as

$$\bigcup_{w \in T_k} \mathcal{F}_d^{i \to *w} = \mathcal{F}_d - \bar{\mathcal{F}}.$$

Now, we can define the polynomial $\bar{F} = \sum_{f \in \bar{\mathcal{F}}} \omega^f$ and similarly for $\bar{F}^{(j)}$. With this, the left hand side of Equation 3.3 becomes

$$\left(F^{i \to *j} + \bar{F}^{(j)}\right)\left(F^{\to *} - \bar{F}\right) = F^{i \to *j} F^{\to *} + F^{\to *} \bar{F}^{(j)} - F^{\to *j} \bar{F}.$$

Thus, we can prove our theorem by showing that $F^{\to *} \bar{F}^{(j)} = F^{\to *j} \bar{F}$ or, equivalently, by defining a weight preserving bijection, Υ, from the set $\mathcal{F}^{\to *} \times \bar{\mathcal{F}}^{(j)}$ to the set $\mathcal{F}^{\to *j} \times \bar{\mathcal{F}}$.

To this end, we let (f_1, f_2) belong to the former set and $\Upsilon(f_1, f_2) = (g_1, g_2)$ where $g_1 = f_1 - f_1 \mid_{T_k} + f_2 \mid_{T_k}$ and $g_2 = f_2 - f_2 \mid_{T_k} + f_1 \mid_{T_k}$. In words, Υ swaps all edges contained in T_k between f_1 and f_2. The resulting pair of subgraphs are both maximal spanning forests of G. Since f_2 has j as a root, so too does g_1. Similarly, though g_2 is no longer necessarily rooted in j, it does still contain the entire path, found in f_2, from i to its root vertex outside of T_k. Therefore, $g_1 \in \mathcal{F}^{\to *j}$ and $g_2 \in \bar{\mathcal{F}}$. To see that Υ is indeed a bijection, note that its inverse is very easy to define, simply applying the same edge swap used to define Υ to the set $\mathcal{F}^{\to *j} \times \bar{\mathcal{F}}$. It is also immediate that the map is weight preserving since the pair (g_1, g_2) has the same multi-set of edges as

(f_1, f_2) and we therefore have $\omega^{f_1}\omega^{f_2} = \omega^{g_1}\omega^{g_2}$. $\square$

Theorem 3.3.12. *The vector sets $\{\gamma_k\}_{k=1}^d$ and $\{\bar{\gamma}_k\}_{k=1}^d$ are bases for the left and right null space of L respectively.*

Proof. Note that each set is clearly linearly independent since, for each $v \in T_k$, γ_k and $\bar{\gamma}_k$ are the only members of their respective sets with a non-zero vth entry. Since the dimension of the nullspace of L is equal to d by Corollary 2.1.10, we need only show that each γ_k and $\bar{\gamma}_k$ is annihilated by L.

Evaluating $L\bar{\gamma}_k$ directly, we find that it has ith entry proportional to

$$\left(\sum_{e\in S^{-1}(i)} \omega^e\right)\left(\sum_{v\in T_k} F_d^{i\to *v}\right) - \sum_{e\in S^{-1}(i)} \omega^e \sum_{v\in T_k} F_d^{t(e)\to *v}$$

which we rewrite as

$$\sum_{v\in T_k}\left(\sum_{e\in S^{-1}(i)}\sum_{f\in\mathcal{F}_d^{i\to v}} \omega^e\omega^f - \sum_{e\in S^{-1}(i)}\sum_{f\in\mathcal{F}_d^{t(e)\to *v}} \omega^e\omega^f\right). \tag{3.4}$$

We will show that, for each $v \in T_k$, the parenthesized expression in (3.4) is equal to 0. First, we deal with some special cases. If $i \in T_k$, then for any edge $e \in s^{-1}(i)$, in $t(e) \in T_k$ as well. By Lemma 2.2.9, we then have $\mathcal{F}_d^{\to i} = \mathcal{F}_d^{t(e)\to i}$ implying that each double sum above is equal. Alternately if $i \in T_l$ for some $l \neq k$, then for any $v \in T_k$ and $e \in s^{-1}(i)$, both $\mathcal{F}_d^{i\to v}$ and $\mathcal{F}_d^{t(e)\to v}$ are empty.

Now, we may assume that i is not in any terminal component of G and therefore is not a root of any forest in $\mathcal{F}_d$. The left hand double sum in 3.4 is a sum over the set $A = s^{-1}(i)\times\mathcal{F}_d^{i\to *v}$ while the right hand one is a sum over $B = \bigcup_{e\in s^{-1}(i)}\{e\}\times\mathcal{F}_d^{t(e)\to v}$.

In both cases, (e, f) contributes ω^{f+e} to the sum. Thus the two sums are equal if we can define a weight preserving bijection from the former set to the latter. To this end, we define the map ϕ that takes a pair $(e, f) \in A$ and returns a pair $(\tilde{e}, \tilde{f})$ where $\tilde{e}$ is the unique edge in f so that $s(\tilde{e}) = i$ unless $f \in \mathcal{F}_d^{t(e)\to v}$ in which case $\tilde{e} = e$. In either case, $\tilde{f} = f - \tilde{e} + e$. So, if $f \in \mathcal{F}_d^{t(e)\to v}$ then ϕ is the identity. Otherwise, it acts to swap the edges $\tilde{e}$ and e in f. In either case, the resulting pair belongs to $\{\tilde{e}\} \times \mathcal{F}_d^{t(\tilde{e})\to v}$. The first definition case gives us edge forest pairs in B so that i has a path to v in f while the second case gives us edge forest pairs in B so that i does not have a path to v in f. In fact, with this observation, the map ϕ is clearly invertible. Since Φ is also clearly weight preserving, we see that L indeed annihilates $\bar{\gamma}_k$.

Similarly, evaluating $\gamma_k^T L$ directly, we find that it has ith entry equal to

$$\left(\sum_{e\in S^{-1}(i)} \omega^e\right)\left(\delta_{i\in T_k} F_d^{\to *i}\right) - \sum_{e\in T^{-1}(i)} \omega^e \delta_{s(e)\in T_k} F_d^{\to *s(e)}.$$

If $i \notin T_k$, then neither is any vertex in $N^s(i)$. In other words, both sides of the above expression are equal to 0.

On the other hand, if $i \in T_k$, then note $N^s(i) \cap T_k$ must contain at least one vertex so that neither right nor left sum is equal to 0. In fact, if we let $\bar{\mathcal{D}}$ denote the set of all spanning functional digraphs of G with exactly $d+1$ edges and with a single cycle contained in T_k and containing i, then we have

$$\left(\sum_{e\in S^{-1}(i)} \omega^e\right)\left(\delta_{i\in T_k} F_d^{\to *i}\right) \;=\; \sum_{e\in T^{-1}(i)} \omega^e \delta_{s(e)\in T_k} F_d^{\to *s(e)} = \sum_{f\in\bar{\mathcal{D}}} \omega^f. \tag{3.5}$$

To see this, let $f \in \bar{\mathcal{D}}$. Then there are unique edges e_1 and e_2 in f so that

$s(e_1) = t(e_2) = i$. Now, $f - e_1$ is clearly belongs to $\mathcal{F}_d^{\to *i}$ while $e_2 \in t^{-1}(i)$ and $f - e_2$ belongs to $\mathcal{F}_d^{\to * s(e_2)}$ with $s(e_2) \in T_k$. Similarly, an edge from $s^{-1}(i)$ and a forest from $\mathcal{F}_d^{\to *i}$ combine to make a unique element of $\bar{\mathcal{D}}$ as do an edge from $t^{-1}(i) \cap T_k$ and a forest from $\mathcal{F}_d^{\to * s(e)}$. $\square$

Corollary 3.3.13. *If G is strongly connected, so that $d = 1$, then $\gamma_1 = diag(Q_1)$ and $\bar{\gamma}_1 = \mathbb{1}$. In general, γ_k is equal to $diag(Q_1(\tau, \omega))$ with $\tau_i = \delta_{i \in T_k}$ while the ith entry of $\bar{\gamma}_k$ is equal to 1 if and only if there is a path from i to T_k and no path from i to any other terminal component of G.*

3.4 Directed Threshold Graphs

To illustrate the computational challenges that arise when working with multivariate forest polynomials, we consider here a class of graphs that possess rich and yet simplified adjacency structure. The results of this section were inspired by and generalize [22].

Definition 3.4.1. *A graph G, on vertex set $[n]$, is a* directed threshold graph *if there are vectors $a, b \in \{0,1\}^{n-1}$ such that $i < j$ implies that $i \to j$ if and only if $a_{j-1} = 1$ and $j \to i$ if and only if $b_{i-1} = 1$.*

Example 3.4.2. Consider the vectors $a = [1, 0, 0, 1]$ and $b = [0, 1, 1, 0]$. Then the corresponding directed threshold graph is given by Figure 3.5.

To study these graphs we will consider a pair of sequences $a_1, a_2, \ldots$ and $b_1, b_2, \ldots$. The graph $T(n, a, b)$ will then be the threshold graph defined by both $[a_1, \ldots, a_{n-1}]$ and $[b_1, \ldots, b_{n-1}]$. To keep things general we will treat the a_i and b_i as variables in the analysis below. Of course, there is an obvious generalization to edge weighted

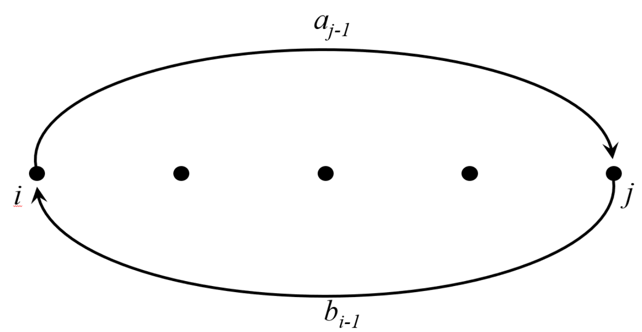


Figure 3.4: Edge dependency in $T(n, a, b)$ for $i \to j$ and $j \to i$ when $i < j$.

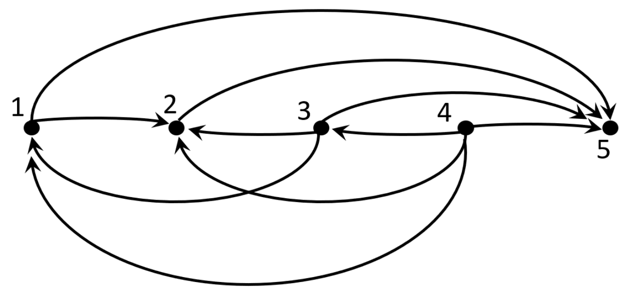


Figure 3.5: The graph described in Example 3.4.2

graphs in which we allow these variables to range over values other than just $\{0, 1\}$.

Directed threshold graphs are of interest in part because they represent a rich class of graphs in which certain algorithmic tasks are guaranteed to be easier than the general worst case. For a broad discussion of these issues see [3]. Given this fact, it is natural to wonder if their forest generating functions might possess simple closed formulas. The first reason to think that this might be so is that the Laplacian matrix of a directed threshold graph is highly structured.

In what follows, we denote the forest polynomial and forest matrix of $T(n, a, b)$ by $F_n(\tau)$ and $Q_n(\tau)$. Note that we suppress the edge weight function ω, however it will always be given by $\omega(i \to j) = a_j$ when $i < j$ and b_i when $j < i$. We can also think of $F_n(\tau)$ as the forest polynomial of the complete graph K_n with its ω variable specialized by a suitable partition.

Lemma 3.4.3. *The directed threshold graph T_{n+1} has a Laplacian matrix $L_{n+1}(\tau)$ that can be described recursively as*

$$L_{n+1}(\tau) = \begin{bmatrix} L_n(\tau + a_n) & -a_n\mathbb{1} \\ -b_n\mathbb{1}^T & \tau_{n+1} + nb_n \end{bmatrix}.$$

From Lemma 3.4.3, we can derive a recursive expression for the multivariate forest polynomial of T_{n+1}.

Proposition 3.4.4. *The forest polynomial of T_{n+1} satisfies*

$$F_{n+1}(\tau) = (\tau_{n+1} + nb_n)F_n(\tau + a_n) - a_nb_n\mathbb{1}^TQ_n(\tau + a_n)\mathbb{1}.$$

Proof. Using standard row operations, the matrix in 3.4.3 is equivalent to

$$\begin{bmatrix} L_n(\tau + a_n) - \frac{a_nb_n}{\tau_{n+1}+nb_n}J & 0 \\ -b_n\mathbb{1}^T & \tau_{n+1} + nb_n \end{bmatrix}$$

The result then follows from the well known matrix-determinant lemma which we state in chapter 4 below as lemma 4.1.4. □

This recurrence will be useful if we can do something with the expression $\mathbb{1}^TQ_n(\tau + a_n)\mathbb{1}$. Unfortunately, it is not obvious that this is possible. Nonetheless, we might begin by attempting to exploit the adjugate equation and the recursive structure of L_{n+1}.

To this end, let

$$Q_{n+1}(\tau) = \begin{bmatrix} B(\tau) & y(\tau) \\ x(\tau)^T & u(\tau) \end{bmatrix} \tag{3.6}$$

where B is $n \times n$. From the equations $L_{n+1}(\tau)Q_{n+1}(\tau) = Q_{n+1}(\tau)L_{n+1}(\tau) = F_{n+1}(\tau)I$, we can infer 8 matrix equations, given in the following lemma.

Lemma 3.4.5.

(i) $L_n(\tau + a_n)B(\tau) - a_n\mathbb{1}x(\tau)^T = F_{n+1}(\tau)I$

(ii) $L_n(\tau + a_n)y(\tau) - a_n u_n(\tau)\mathbb{1} = 0$

(iii) $-b_n\mathbb{1}^T y(\tau) + (\tau_{n+1} + nb_n)u(\tau) = F_{n+1}(\tau)$

(iv) $-b_n\mathbb{1}^T B(\tau) + (\tau_{n+1} + nb_n)x(\tau)^T = 0$

(v) $B(\tau)L_n(\tau + a_n) - b_n y(\tau)\mathbb{1}^T = F_{n+1}(\tau)I$

(vi) $x(\tau)^T L_n(\tau + a_n) - b_n u_n(\tau)\mathbb{1}^T = 0$

(vii) $-a_n x(\tau)^T\mathbb{1} + (\tau_{n+1} + nb_n)u_n(\tau) = F_{n+1}(\tau)$

(viii) $-a_n B(\tau)\mathbb{1} + (\tau_{n+1} + nb_n)y(\tau) = 0.$

Using Proposition 3.4.4, these equations relate B, x, y, and u to L_n, F_n, a_n, and b_n.

Lemma 3.4.6. *The submatrices* x*,*y*,and* u *are independent of* τ_{n+1}*.*

Proof. By definition, u and each entry of y has the form $F_{n+1}^{v\to *n+1}(\tau)$ for some $v \in V$ and this is independent of τ_{n+1}. Similarly, each entry of x has the form $F_{n+1}^{n+1\to *v}(\tau)$ for some $v \in V - \{n+1\}$. $\square$

Proposition 3.4.7. *The polynomial u satisfies*

$$u(\tau) = F_n(\tau + a_n).$$

Proof. Using the recurrence from Proposition 3.4.4 in item (iii) of Lemma 3.4.5, we need only differentiate with respect to τ_{n+1}. The result then follows directly from Lemma 3.4.6. $\square$

Proposition 3.4.8. *The vectors x and y satisfy*

$$x(\tau)^T = b_n \mathbb{1}^T Q_n(\tau + a_n)$$

and

$$y(\tau) = a_n Q_n(\tau + a_n)\mathbb{1}.$$

Proof. These identities follow from multiplying equations (ii) and (vi) from Lemma 3.4.5, on the left and right respectively, by $Q_n(\tau + a_n)$ and then applying Lemma 3.4.7. $\square$

Proposition 3.4.9. *The matrix B satisfies*

$$B(\tau) = (\tau_{n+1} + nb_n)\, Q_n(\tau + a_n) - \frac{a_n b_n Q_n(\tau + a_n)}{F_n(\tau + a_n)} \left(\mathbb{1}^T Q_n(\tau + a_n)\mathbb{1} I - J Q_n(\tau + a_n)\right).$$

Proof. Multiply equation (i) from 3.4.5 on the left by $Q_n(\tau + a_n)$ and apply Lemma 3.4.5 and Proposition 3.4.4. $\square$

Corollary 3.4.10. *The sum of entries of the matrix B is given by*

$$\mathbb{1}^T B(\tau)\mathbb{1} = (\tau_{n+1} + nb_n)\, \mathbb{1}^T Q_n(\tau + a_n)\mathbb{1}.$$

Proof. This follows from Proposition 3.4.9 by multiplying left and right by $\mathbb{1}^T$ and $\mathbb{1}$ respectively. To see this, note that $\mathbb{1}^T Q_n(\tau + a_n) J Q_n(\tau + a_n)\mathbb{1} = \left(\mathbb{1}^T Q_n(\tau + a_n)\mathbb{1}\right)^2$. $\square$

Proposition 3.4.11. *Letting $F_0(\tau) = 1$ and $a_0 = 0$, we have*

$$\mathbb{1}^T Q_{n+1}(\tau)\mathbb{1} = \sum_{k=0}^{n} \left(\prod_{i=k+1}^{n} (\tau_{i+1} + (i+1)b_i + a_i + \cdots + a_n \right) F_k(\tau + a_k + \ldots + a_n).$$

Proof. Applying Equation 3.6 to the left hand side, we have $\mathbb{1}^T Q_{n+1}(\tau)\mathbb{1} = \mathbb{1}^T B\mathbb{1} + x^T\mathbb{1} + \mathbb{1}^T y + u$. By Propositions 3.4.7,3.4.8, and Corollary 3.4.10, this means that

$$\mathbb{1}^T Q_{n+1}(\tau)\mathbb{1} = F_n(\tau + a_n) + (\tau_{n+1} + (n+1)b_n + a_n)\mathbb{1}^T Q_n(\tau + a_n)\mathbb{1}$$

which, by repeated application, becomes the given proposition. $\square$

To simplify our expressions below, we introduce the following notation.

Definition 3.4.12. *Given a sequence s and indices i, j, we set*

$$s[i, j] = s_i + \cdots + s_j$$

with the convention that $s[i, j] = 0$ when $j > i$.

With this notation, we can finally get a pure recursive expression for $F_{n+1}(\tau)$ which follows immediately from Propositions 3.4.4 and 3.4.11.

Theorem 3.4.13. *The multivariate forest polynomial F_{n+1} satisfies the recurrence*

$$F_{n+1}(\tau) = (\tau_{n+1} + nb_n)F_n(\tau + a_n)$$
$$-a_n b_n \sum_{k=0}^{n-1} \left(\prod_{i=k+1}^{n-1} (\tau_{i+1} + (i+1)b_i + a[i,n]) \right) F_k(\tau + a[k,n]).$$

It is unclear if there is a simpler recursive formula for F_n as we have defined it. Interestingly, there is a closely related digraph family whose forest polynomial does exhibit a much simpler recursive formula.

Definition 3.4.14. *A graph G, on vertex set $[n]$, is a directed co-threshold graph if there are vectors $a, b \in \{0,1\}^{n-1}$ such that $i < j$ implies $i \to j$ if and only if $a_i = 1$ and $j \to i$ if and only if $b_j = 1$.*

Note that we have only very slightly altered the Definition 3.4.1. To clarify the distinction between the two definitions, we can think of each construction as alternative weightings of the complete graph. In the case of directed threshold graphs, for each edge e of K_n, we assign e the weight $a_{s(e)}$ if $s(e) < t(e)$ and $b_{s(e)}$ otherwise. For the co-threshold graphs, we assign e the weight $a_{t(e)}$ if $s(e) < t(e)$ and $b_{t(e)}$ otherwise. Let us denote the family of directed co-threshold graphs by $\tilde{T}(n, a, b)$.

Lemma 3.4.15. *Let $a, b \in \{0,1\}^n$ and let $a(i) = (a_1, a_2, \ldots, a_i)^T$ and similarly for $b(i)$. Then the graph $\tilde{T}(n+1, a, b)$ has a Laplacian matrix $\tilde{L}_{n+1}(\tau)$ that can be* *described recursively as*

$$\tilde{L}_{n+1}(\tau) = \begin{bmatrix} \tilde{L}_n(\tau + a(n)) & -a(n) \\ -b(n)^T & \tau_{n+1} + a[1,n] \end{bmatrix}$$

Carrying the recursion one more step forward, we have

$$\tilde{L}_{n+1}(\tau) = \begin{bmatrix} \tilde{L}_{n-1}(\tau + 2a(n)) & -a(n-1) & -a(n-1) \\ -b(n-1)^T & \tau_n + b[1,n-1] + a_n & -a_n \\ -b(n-1)^T & -b_n & \tau_{n+1} + b[1,n] \end{bmatrix}$$

which yields the following recurrence, where $\tilde{F}_n$ denotes the forest polynomial of $\tilde{T}(n,a,b)$.

Proposition 3.4.16. *The multivariate forest polynomial $\tilde{F}_{n+1}$ satisfies the recurrence*

$$\tilde{F}_{n+1}(\tau) = (\tau_{n+1} + 2b[1,n-1] + a_n)\tilde{F}_n(\tau + a(n)) -$$

$$((2b[1,n-1] + 2a_n + b_n)\tau_n + (b[1,n] + a_n)(b[1,n-1] + 2a_n))\tilde{F}_{n-1}(\tau + 2a(n-1)).$$

Proof. Subtract the second to last row of $\tilde{L}_{n+1}(\tau)$ from the last row. Then, subtract the second to last column from the last column. This produces an equivalent matrix

$$\begin{bmatrix} \tilde{L}_{n-1}(\tau + 2a(n-1)) & -a(n-1) & 0 \\ -b(n-1)^T & \tau_n + b[1,n-1] + a_n & -(\tau_n + b[1,n-1] + 2a_n) \\ 0 & -(\tau_n + b[1,n] + a_n) & \tau_{n+1} + \tau_n + 2b[1,n] + 2a_n \end{bmatrix}$$

and the proposition follows by Laplace expansion along the bottom row. Using the fact that $\tilde{F}_n$ is multilinear, we can be sure that any terms involving τ_n^2 cancel out. This leaves behind the given formula. □

This two term recurrence allows for a significantly more efficient recursive computation of $\tilde{F}_n(\tau)$. Interestingly, these two graphs are obviously isomorphic.

Proposition 3.4.17. *Let $a' = (a_{n-1}, \ldots, a_1)$ and similarly for b'. Then, $\tilde{T} = \tilde{T}(n, a, b)$ is isomorphic to $T = T(n, b'), a')$.*

Proof. Note that these are not multigraphs so that we need only specify a map on V. We claim that the map ϕ, taking $i \in [n]$ to $n - i$ is a graph isomorphism. To see this, consider $i, j \in [n]$. If $i < j$, then $i \to j$ is an edge of T just in case $a_{j-1} = 1$. In this case $n - j < n - i$ so that $\phi(i) \to \phi(i)$ is an edge of $\tilde{T}$ just in case $a'_{n-j} = 1$. Of course, by our definition, $a'_{n-j} = a_{j-1}$. An identical argument for the case $j < i$ shows that the two graphs are isomorphic. □

This proposition means that we could compute $\tilde{T}$ via the recurrence in Proposition 3.4.16 and then transform the resulting polynomial by substituting $\tau_{\phi(i)}$ for τ, $b_{\phi(i)}$ for a_i and $a_{\phi(i)}$ for b_i.

On the other hand, we will see below that the form of the Laplacian matrix for $T(n, a, b)$ seems much more suited to deriving a formula for $Q(t)$.

Specializing the vector τ to $t\mathbb{1}$ simplifies the above formulas considerably, allowing us to recover a closed formula.

Proposition 3.4.18. *The univariate forest polynomial of $T(n, a, b)$ is equal to*

$$F_n(t) = t \prod_{j=1}^{n-1} (t + jb_j + a[j, n-1]) \,.$$

Proof. By Proposition 3.3.6, we have $\mathbb{1}^T Q_n(t)\mathbb{1} = \frac{n}{t}F_n(t)$. Therefore, by Proposition 3.4.4,

$$\begin{aligned} F_n(t) &= (t+(n-1)b_{n-1})F_{n-1}(t+a_{n-1}) - \frac{a_{n-1}b_{n-1}(n-1)}{t+a_{n-1}}F_{n-1}(t+a_{n-1}) \\ &= \frac{t(t+a_{n-1}+(n-1)b_{n-1})}{t+a_{n-1}}F_{n-1}(t+a_{n-1}). \end{aligned}$$

Letting $\hat{F}_n(t) = \frac{F_n(t)}{t}$, we see that $\hat{F}_n(t) = (t+a_{n-1}+(n-1)b_{n-1})\hat{F}_{n-1}(t+a_{n-1})$ with the recurrence terminating with $\hat{F}_1(t) = 1$. It follows that

$$\hat{F}_n(t) = \prod_{j=1}^{n-1} (t + jb_j + a[j, n-1])\,.$$

$\square$

This formula also gives us the Laplacian eigenvalues of $T_n(a, b)$ [32].

Proposition 3.4.19. *The vertex j comprises a root strong component of T_n if and only if b_{j-1} and $a_{j-1}, \ldots, a_{n-1}$ are all* 0.

Proof. Looking at the definition of T_n, this is simply the condition for vertex j to have no outgoing edges. $\square$

With a reasonably compact expression for $F_n(t)$, we are also able to derive expressions for the univariate forest matrix.

Proposition 3.4.20. *Specializing u and y with $\tau = t\mathbb{1}$ we have*

$$u(t) = (t + a_n) \prod_{j=1}^{n-1} (t + jb_j + a[j, n])$$

and

$$y(t) = a_n \prod_{j=1}^{n-1} (t + jb_j + a[j,n]) \mathbb{1}.$$

Proof. The expression for u follows directly from 3.4.18. Similarly, we have $y(t) = a_n Q_n(t + a_n)\mathbb{1} = a_n \hat{F}_n(t + a_n)\mathbb{1}$. □

Unfortunately, we cannot similarly use Proposition 3.3.6 to get at $x(t)$ directly. Instead, we appeal a bit more to the recursive structure of Q_{n+1}. The following proposition follows from specializing Proposition 3.4.9.

Proposition 3.4.21. *The matrix B specialized by $\tau = t\mathbb{1}$ is equal to*

$$B(t) = \frac{1}{(t + a_n)} \left(t(t + nb_n + a_n)I + a_n b_n J\right) Q_n(t + a_n).$$

This recurrence allows us to build a direct formula for both $x(t)$ and ultimately $B(t)$ through a related simple recurrence.

Corollary 3.4.22. *The column sums of $B(t)$ satisfy*

$$\mathbb{1}^T B(t) = (t + nb_n)\mathbb{1}^T Q_n(t + a_n)$$

Proposition 3.4.23. *Specializing x with $\tau = t\mathbb{1}$, we have*

$$x(t)_i = b_n \left(t + ia_{i-1} + a[i,n]\right) \left(\prod_{\substack{j=1 \\ j \neq i}}^{n-1} (t + jb_j + a[j,n]) \right).$$

Proof. First, let

$$h_i(t) = (t + ia_{i-1}) \prod_{j=1}^{i-2} (t + jb_j + a[j, i-1]) .$$

By Equation 3.6, we have $\mathbb{1}^T Q_{n+1}(t) = \left[\mathbb{1}^T B(t) + x(t)^T \mid \mathbb{1}^T y(t) + u(t)\right]$. Applying Proposition 3.4.20, we see that $\mathbb{1}^T y(t)+u(t) = h_{n+1}(t)$. Similarly, by Proposition 3.4.8 and Corollary 3.4.22, $\mathbb{1}^T B(t) + x(t)^T = (t + (n+1)b_n)\mathbb{1}^T Q_n(t + a_n)$. Re-indexing, we have $\mathbb{1}^T Q_n(t) = \left[(t + nb_{n-1})\mathbb{1}^T Q_{n-1}(t + a_{n-1}) \mid h_n(t)\right]$ Unpacking this expression to the ith entry, yields

$$\left(\prod_{j=i}^{n-1} t + (j+1)b_j + a[j+1, n-1]\right) h_i\left(t + a[i, n-1]\right).$$

Now we apply Proposition 3.4.8. $\square$

Having found a closed expression for $x(t)$, we can now derive a closed expression for $Q(t)$. By applying Proposition 3.4.8 to Proposition 3.4.21, $B(t)$ satisfies the following.

$$B(t) = \frac{t(t + nb_n + a_n)}{t + a_n} Q_n(t + a_n) + \frac{a_n}{t + a_n} 1x(t)^T.$$

Since we have found a closed formula for $x(t)$, we can use this give a tractable recurrence for $Q(t)$ directly. For this, we define the $n+1 \times n+1$ matrix

$$\bar{Q}_i(t) = \begin{bmatrix} Q_i(t) & 0 \\ 0 & 0 \end{bmatrix}$$

where the 0 entries are zero matrices of the required sizes. Note that $\bar{Q}_{n+1}(t) = Q_{n+1}(t)$. We will also have to parameterize u, x and y by n. That is, we take u_n, x_n, and y_n, to be the defined as in Propositions 3.4.20 and 3.4.23. Thus, for example, u_i is a scalar polynomial of degree i and $x_i(t)$ is an $i \times 1$ column vector. With these definitions we can express a simple recurrence for $\bar{Q}_n(t)$.

Proposition 3.4.24. *The matrix $\tilde{Q}_{n+1}$ satisfies the recurrence relation*

$$\bar{Q}_{n+1}(t) = \alpha_n(t)\bar{Q}_n(t + a_n) + X_n(t),$$

where

$$\alpha_i(t) = \frac{t(t + ib_i + a_i)}{t + a_i},$$

and

$$X_i(t) = \begin{bmatrix} \frac{a_i}{t+a_i}\mathbb{1}x_i(t)^T & y_i(t) & 0 \\ x_i(t)^T & u_i(t) & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

where again zero entries in X_i are matrices of the appropriate size. To be explicit, X_i is an $n+1$ by $n+1$ matrix with an $i \times i$ nonzero block in the upper left hand corner.

Note that $Q_1(t)$ is just the 1×1 matrix with entry 1. With this in mind, we can unpack the above recurrence directly.

Theorem 3.4.25. *The univariate forest matrix of $T(n+1, a, b)$ is equal to*

$$Q_{n+1}(t) = \sum_{k=1}^{n} \left(\prod_{i=k+1}^{n} \alpha_i(t + a[i+1, n]) \right) X_k(t + a[k+1, n]).$$

We will refrain from recovering the full closed expression for $F^{i\to *j}(t)$, however note that each entry of X_k is known by a closed formula. We can see that there are $\max(i,j)$ of the matrices X_k with a nonzero ij entry. Thus, as this numbers gets smaller, the polynomial $F^{i\to *j}(t)$ gets increasingly complex. In light of the simplicity of $F(t)$ itself, this case study shows that the univariate $i \to *j$ polynomials, and by extension, their multivariate counterparts, need not resemble $F(t)$ much at all. On the other hand, we will see cases in Sections 4.2 and 4.3 where these polynomials are nearly identical. In Section 4.5 we will explain this phenomena and quantify exactly how similar F and $F^{i\to *j}(t)$ will be to each other for arbitrary graphs.

4 The Unweighted Univariate Forest Matrix

In this chapter, we study the relation between forest polynomials and the matrix algebra generated by $L(0,\omega)$. We will call this matrix algebra the *Laplacian algebra* of a graph G. In light of Proposition 2.2.11, we begin by specializing the τ variables by taking $\tau = t\mathbb{1}$ and letting $\omega = 1$. We will however consider how these techniques might apply to alternative specializations of ω, returning to this subject in the final chapter. In what follows, we let $L = L(0,\mathbb{1})$ and the $Q(t) = Q(t\mathbb{1},\mathbb{1})$. We will also refer to L as *the* Laplacian matrix of G.

4.1 Reciprocity and Perturbation Formulas

In this section, we develop a reciprocity formula for $Q_{\bar{G}}(t)$ where $\bar{G}$ is the complement of G. Note that our definition of complement in 1.1 is independent of any choice of edge weighting. Instead, it refers only to the adjacency relations

The next lemma is more or less obvious from the functional equation $M\ \mathrm{adj}M =$

$(\det M)I$.

Lemma 4.1.1. *If M is a block diagonal matrix with diagonal blocks $M_1, \ldots, M_k$, then $\operatorname{adj} M$ is block diagonal with diagonal blocks $d_1(x)\ \operatorname{adj} M_1, \ldots, d_k(x)\ \operatorname{adj} M_k$ where $d_i(x) = \prod_{j \neq i} \det M_j$.*

Now, from the definition given in 1.1, it is obvious that the Laplacian matrix of a disjoint union of graphs is block diagonal with ith block equal to the Laplacian matrix of the ith graph, we can now derive the following.

Corollary 4.1.2. *If $G = G_1 \dot{+} \ldots \dot{+} G_k$, then $Q_G(t)$ is block diagonal with diagonal blocks $\hat{f}_1(t) Q_{G_1}(t), \ldots, \hat{f}_k(t) Q_{G_k}(t)$ where $\hat{f}_i(t) = \prod_{j \neq i} f_{G_j}(t)$.*

We can apply this observation to Example 3.3.5.

Example 4.1.3. For $G = K_{n_1} \oplus \cdots \oplus K_{n_k}$,

$$Q(G) = t^{k-1} \prod_{i=1}^{k} (t + n_i)^{n_i - 2} \left[\delta_{i,j} \left(\prod_{l \neq i} (t + n_l) \right) (t I_{n_i} + J_{n_i}) \right]_{1 \leq i,j \leq k}.$$

To develop a reciprocity formula for the unweighted forest matrix, we will use some standard results from matrix perturbation theory [31].

Lemma 4.1.4. *[Matrix Determinant Lemma] For any square matrix A, we have*

$$\det \left(A + uv^T \right) = \det A + v^T \left(\operatorname{adj} A \right) u.$$

Lemma 4.1.5. *[Sherman-Morrison Formula] Suppose that both A and $A + uv^T$ are invertible. Then, $1 + v^T A^{-1} u \neq 0$ and*

$$\left(A + uv^T\right)^{-1} = A^{-1} - \frac{1}{1 + v^T A^{-1} u} A^{-1} uv^T A^{-1}.$$

Taken together, these lemmas give us an update formula for the adjugate of an invertible rank one update of an invertible matrix.

Corollary 4.1.6. *Given the same assumptions as in lemma 4.1.5, we have*

$$\operatorname{adj}\left(A + uv^T\right) = \left((1 + v^T A^{-1} u)I + A^{-1} uv^T\right) \operatorname{adj}(A).$$

Lemma 4.1.7. *For any real matrix A, the matrix $tI + A$ is invertible over the field of fractions of $\mathbb{R}[t]$.*

Proof. If $(tI + A)v = 0$, then on the one hand, by setting $t = 0$, we have $Av = 0$. But also, $Av = (-t)v$ and it follows that $(-t)v = 0$ for all values of t. This is only possible if $v = 0$. □

Finally, to relate $Q_{\bar{G}}$ to Q_G, we need to relate the Laplacian matrices of these two graphs.

Lemma 4.1.8. *The Laplacian matrix of a graph and its complement are related by*

$$L_{\bar{G}} = nI - J - L_G.$$

Proof. By definition, $A_{\bar{G}} = J - I - A_G$. In addition, we have $d^s_{\bar{G}}(v) = (n-1) - d^s_G(v)$. Now, we can apply Definition 3.1.1. $\square$

Now, the matrix forest generating function of $\bar{G}$ can be neatly written in terms of Q_G.

Theorem 4.1.9.

$$Q_{\bar{G}}(t) = \frac{(-1)^{n-1}}{t+n}\,(tI + J)\,Q_G(-t-n).$$

Proof. From the definition, $Q_{\bar{G}}(t) = \operatorname{adj}(tI + L_{\bar{G}})$. Applying Lemma 4.1.8 and rearranging terms, we have $tI + L_{\bar{G}} = (-1)\left((-t-n)I + L_G + J\right)$. If we let $s = -t-n$ and note that $J = \mathbb{1}\mathbb{1}^T$, then it becomes clear that $Q_{\bar{G}}(t)$ is a rank one update of $(-1)^{n-1}Q_G(s)$. Applying Corollary 4.1.6 and simplifying gives the result. $\square$

The above argument does not work for the full multivariate forest matrix as we made crucial use of the fact that $nI - L_{\bar{G}}$ is a rank 1 correction of L_G. As we will discuss farther in Chapter 5.1, using the specialization $\omega = Tx$ we obtain a complete graph has a rank 1 adjacency matrix $|x|I + \mathbb{1}x^T$. In this case, we have that

$$L_{\bar{G}}(t\mathbb{1}, Tx) = |x|I - \mathbb{1}x^T - L_G(t\mathbb{1}, Tx)$$

so that analogous results may be derived. This approach provides a generalization of some of the results in [36]. More generally, we can see that similar methods will apply to graphs H, G whenever $L_H(0,\omega)$ is a rank one perturbation of $L_G(0,\omega)$. Of course, this will typically require some specialization of ω.

As an application of 4.1.9, we determine $Q_G(t)$ for the complete multipartite graph $G = K_{n_1,\dots,n_k}$.

Corollary 4.1.10. *If $n = n_1 + \cdots + n_k$, then*

$$Q_{K_{n_1,\dots,n_k}} = (t+n)^{k-2} \prod_{i=1}^{k} (t+n-n_i)^{n_i-2} \cdot$$
$$\left(\left[\delta_{ij} \left(\prod_{l \neq i} (t+n-n_l) \right) ((t+n)I_{n_i} + J_{n_i}) \right]_{1 \leq i,j \leq k} + \left[(t+n) \prod_{i=1}^{k} (t+n-n_i) \right] J_n \right) \tag{4.7}$$

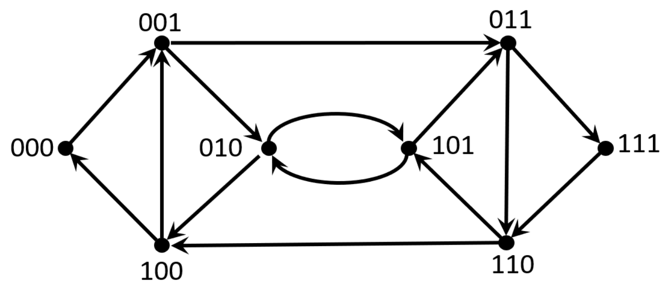
Proof. Since the complement of $K_{n_1,\dots,n_k}$ is $K_{n_1} \oplus \cdots \oplus K_{n_k}$, we apply Theorem 4.1.9 to Example 4.1.3 . □

4.2 de Bruijn Graphs

Another excellent case study for the univariate forest matrix comes from the de Bruijn graphs and their many relatives. These graphs have been extensively studied and applied to a range of contexts from coding theory [30] to network design [11] to genomic assembly [20] to synchronizing automata [5]. Here we will characterize the Laplacian algebra of such graphs and use this to derive a closed expression for $Q(t)$.

Definition 4.2.1. *For $n, k > 0$, the $(n, k)-$de Bruijn Graph, $B(n, k)$, is defined to be the graph with vertex set $V = [n]^k$, the set of k-tuples of the numbers $1, \dots, n$. The arcs of $B(n, k)$ are defined by the condition that $a \to b$ just in case $a = (x_1, \dots, x_k)$, $b = (y_1, \dots, y_k)$ and $x_i = y_{i-1}$ for $i = 2, \dots, k$.*

Example 4.2.2. The figure below depicts $B(2, 3)$. Note that we have suppressed the loops at 000 and 111.



We will continue referring to the above definition of $B(n,k)$. However, it is worth noting that because a loop can never appear in a spanning forest, almost everything that we say will also hold for the graph obtained by deleting the loops from $B(n,k)$. In fact, the only difference in our exposition is that the adjacency matrix that we refer to becomes the Laplacian cospectral adjacency matrix introduced in Section 4.5.

Proposition 4.2.3. *With loops included, the graph $B(n,k)$ is balanced and n regular.*

Proof. : A vertex $a = (x_1, \ldots, x_k)$ points to each vertex of the form $(x_2, \ldots, x_{k-1}, y)$ for $y = 1, \ldots, n$. Similarly, any vertex of the form $(y, x_1 \ldots x_{k-1}$ for $y = 1, \ldots, n$, points to a. □

Of course, by ignoring the loops, the graph ceases to be balanced or regular. On the other hand, the Laplacian matrix of both graphs is the same and, as we shall see, satisfies $L_{B(n,k)}(0, \mathbb{1}) = nI - A_{B(n,k)}$ where $A_{B(n,k)}$ is the adjacency matrix, *with the loops included*.

We begin by establishing some facts about the S and T matrices of balanced and regular graphs.

Proposition 4.2.4. *If G is r source regular with n vertices, then for each ordering* *of $V(G)$, there exists an ordering of $E(G)$ so that $S_G = I_n \otimes \mathbb{1}_d$. If G is balanced, then there is a permutation on E with matrix P so that $T_G = PS_G$.*

Proof. For the first claim, suppose that V has been ordered $v_1, \ldots, v_n$. Then, we can define a partial order on E by $e \lesssim e'$ whenever $s(e) \leq s(e')$. Any linear extension of this order will satisfy the condition. For the second claim, if G is balanced, then for each $v \in V$, we have a bijection $\phi_v : t^{-1}(v) \to s^{-1}(v)$. Thus, we define a permutation on E by $\sigma(e) = \phi_{t(e)}(e)$. Note that in this case, the e, v entry of PS is 1 just in case v is the source of $\sigma^{-1}(e)$. In this case, we see that $v = t(e)$. $\square$

Proposition 4.2.4 allows us to characterize the adjacency algebra of any regular and balanced graph using the permutation P. This follows from the simple observation that, for any m, $(S^*PS)^m = S^*P^mS$.

For the graphs $B(n, k)$, we take the lexicographic ordering of V. In other words, we say that $(x_1, \ldots, x_k) \leq (y_1, \ldots, y_k)$ whenever $x_l \leq y_l$ where l is the largest index so that $x_i = y_i$ for all $i < l$. We will describe the permutation P after fixing a compatible ordering of E. To describe this ordering, we turn to another famous property of $B(n, k)$. Recall that $\mathcal{L}(G)$ denotes the directed line graph of G whose vertex set is $E(G)$ with $e \to e'$ if and only if $t(e) = s(e')$.

Proposition 4.2.5. *The de Bruijn graph satisfies*

$$B(n, k+1) \cong \mathcal{L}(B(n, k)).$$

Proof. We consider a map $\phi : E_{B(n,k)} \to V_{B(n,k+1)}$ given by $\phi(e) = (x_1, \ldots, x_{n+1})$ where $s(e) = (x_1, \ldots, x_n)$ and $t(e) = (x_2, \ldots, x_{n+1})$. This map is clearly a set bijection as it has an easily defined inverse. In addition, we see that $e \to e'$ in $\mathcal{L}(B(n,k))$ if and only if $t(e) = s(e')$. Call this vertex v and suppose that $v = (x_1, \dots, x_n)$. In this case, we have $y, z \in [n]$ so that $s(e) = (y, x_1, \dots, x_{n-1})$ and $t(e) = (x_2, \dots, x_n, z)$. Now, it is clear that $\phi(e) = (y, x_1, \dots, x_n)$ and $\phi(e') = (x_1, \dots, x_n, z)$, or equivalently, that $\phi(e) \to \phi(e')$ in $B(n, k+1)$. It follows that ϕ is a graph isomorphism. $\square$

With this proposition, we can now apply lexicographic ordering to the edge set of $B(n,k)$ as well as its vertex set.

Lemma 4.2.6. *For the de Bruijn graph $B(n,k)$, lexicographic ordering on E is compatible, in the sense of Proposition 4.2.4, with lexicographic ordering on V.*

Proof. This follows from the definition of the lexicographic ordering. Since $s(e)$ is identified with the initial n entries of e, we are guaranteed that if $e < e^{'}$ under lexicographic ordering, then $s(e) < s(e^{'})$ as well. Thus the ordering on E is a linear extension of the ordering induced by the lexicographic ordering on V. $\square$

From this lemma, we can assume that $S = I_n \otimes \mathbb{1}_{n^d}$. It is worth noting that lexicographic ordering on $[n]^k$ actually coincides with the natural ordering obtained by mapping the tuple $(x_1, \dots, x_k)$ to the number $\sum_{i=1}^{k} (x_i - 1) n^{k-i}$. In this sense, we have $V = \{0, 1, 2, \dots, n^k - 1\}$ and $E = \{0, 1, 2, \dots, n^{k+1} - 1\}$. Furthermore, under this ordering, the T matrix takes on a special form.

Proposition 4.2.7. *The target incidence matrix of $B(n,k)$ is*

$$T_{B(n,k)} = \mathbb{1}_n \otimes I_{n^k}.$$

Proof. The (e, v) entry of T is 1 just in case $e = (x_1, \ldots, x_{k+1})$ and $v = (x_2, \ldots, x_{k+1})$. Translating this into the numerical ordering in the previous paragraph, this means that if $e = (x_1 - 1)n^k + b$ where $0 \leq b < n_k$, then $v = b$. For example, if $x_1 = 1$, then e belongs to the first block of n^k (edge indexed) rows in T. This block together with all of the columns of T induces an identity matrix. Indeed, as x_1 increases, we move down one block of n^k edges at a time, finding an identity matrix at each stage. It follows that T is a stack of n identity matrices of size $n^k \times n^k$. □

With these preliminaries in place, we can now describe the adjacency algebra of $B(n, k)$ directly.

Proposition 4.2.8. *With the vertex ordering described in Lemma 4.2.6, we have*

$$A_{B(n,k)} = \mathbb{1}_n \otimes I_{n^{k-1}} \otimes \mathbb{1}_n^*,$$

and, in general, if $0 \leq l \leq k$,

$$A^l_{B(n,k)} = \mathbb{1}_{n^l} \otimes I_{n^{k-l}} \otimes \mathbb{1}^*_{n^l}.$$

Proof. Although S and T are both given in terms of tensor products, it is not immediately clear how to multiply them. However, using the fact that $I_{ab} = I_a \otimes I_b$ and the associativity of the tensor product, we can write $S = I_n \otimes I_{n^{k-1}} \otimes \mathbb{1}_n$. From proposition 4.2.7, $T = \mathbb{1}_n \otimes I_{n^{k-1}} \otimes I_n$. Since these factorizations are compatible, we can see that

$$\begin{aligned} S^*T &= (I_n \mathbb{1}_n) \otimes (I_{n^{k-1}} I_{n^{k-1}}) \otimes (\mathbb{1}^* I_n) \\ &= \mathbb{1}_n \otimes I_{n^{k-1}} \otimes \mathbb{1}_n^*. \end{aligned}$$

The expression for $A^l_{B(n,k)}$ can then be established by induction on l. Let $A = A_{B(n,k)}$. Assuming that $A^l = \mathbb{1}_{n^l} \otimes I_{n^{k-l}} \otimes \mathbb{1}^*_{n^l}$, we multiply

$$\begin{aligned}
A \cdot A^l &= (\mathbb{1}_n \otimes I_{n^{k-1}} \otimes \mathbb{1}^*_n) \cdot (\mathbb{1}_{n^l} \otimes I_{n^{k-l}} \otimes \mathbb{1}^*_{n^l}) \\
&= ((\mathbb{1}_n \otimes I_{n^l}) \otimes (I_{n^{k-l-1}} \otimes \mathbb{1}^*_n)) \cdot ((\mathbb{1}_{n^l}) \otimes (I_{n^{k-l-1}} \otimes (I_n \otimes \mathbb{1}^*_{n^l}))) \\
&= ((\mathbb{1}_n \otimes I_{n^l}) \cdot (\mathbb{1}_{n^l})) \otimes ((I_{n^{k-l-1}} \otimes \mathbb{1}^*_n) \cdot (I_{n^{k-l-1}} \otimes (I_n \otimes \mathbb{1}^*_{n^l})) \\
&= \mathbb{1}_{n^{l+1}} \otimes (I_{n^{k-l-1}} \otimes \mathbb{1}_{n^{l+1}}).
\end{aligned}$$

□

Corollary 4.2.9. *If $l \geq k$, then*

$$A^l = n^{l-k} J_n.$$

Lemma 4.2.10. *For $u, v \in [n]^k$, there is a walk of length l in $B(n,k)$ if and only if either $l \geq k$ or there is an $x \in [n]^{k-l}$ so that $u = zx$ and $v = xy$ for some $z, y \in [n]^l$. If $l \leq k$, then this walk is unique.*

Corollary 4.2.11. *If dist$(u,v) = l$, then $u = wy$, $v = yz$ with $y \in [n]^{k-l}$ as long as possible and $w, z \in [n]^l$. Further, if $y = xx \ldots x$ where $x \in [n]^d$ is repeated m times (so that $k = l + md$), then there is a unique walk from u to v of length $p > 0$ if and only if $p = l + jd$ for some $j = 0, 1, \ldots, m$.*

This characterization gives us a particularly simple structure for the adjacency algebra and, from this description, we can derive a tractable description of the forest generating function for $B(n,k)$.

Proposition 4.2.12. *The adjacency algebra of $B(n,k)$ has basis*

$$\left\{I, A, \ldots, A^{k-1}, J\right\}.$$

Using the rules, $A^k = J$ and $AJ = nJ$ together with the fact that the Laplacian matrix of $B(n,k)$ lies in the algebra generated by A, we can assume that

$$Q(t) = \sum_{j=0}^{k} \alpha_j(t) A^j = \alpha_0(t) I + \alpha_1(t) A + \cdots + \alpha_{k-1}(t) A^{k-1} + \alpha_k(t) J$$

for some polynomials $\alpha_i(t)$. Recall further that $(tI + L)Q(t) = F(t)I$ with $F(t) = t(t+n)^{n^k-1}$. It follows that

$$\begin{aligned} ((t+n)I - A)\left(\sum_{j=0}^{k} \alpha_j(t) A^j\right) &= t(t+n)^{n^k-1} I \\ (t+n)\alpha_0 I + \sum_{j=1}^{k-1} \left((t+n)\alpha_j - \alpha_{j-1}\right) A^j + (t\alpha_k - \alpha_{k-1}) J &= t(t+n)^{n^k-1} I \end{aligned}$$

This yields a system of linear equations in the $\alpha_i(t)$

$$\begin{aligned} (t+n)\alpha_0 &= t(t+n)^{n^k-1} \\ (t+n)\alpha_1 - \alpha_0 &= 0 \\ &\vdots \\ (t+n)\alpha_{k-1} - \alpha_{k-2} &= 0 \\ t\alpha_k - \alpha_{k-1} &= 0 \end{aligned}$$

which is readily solved to give

$$\begin{aligned}
\alpha_0 &= t(t+n)^{n^k-2} \\
\alpha_1 &= t(t+n)^{n^k-3} \\
&\vdots \\
\alpha_{k-1} &= t(t+n)^{n^k-k-1} \\
\alpha_k &= (t+n)^{n^k-k-1}.
\end{aligned}$$

Finally, in our next result, we are able to give a closed form for $Q(t)$.

Theorem 4.2.13. *The matrix forest generating function for $B(n,k)$ is given by*

$$Q(t) = (t+n)^{n^k-k-1}J + \sum_{j=0}^{k-1} t(t+n)^{n^k-j-2}\mathbb{1}_{n^j} \otimes I_{n^{k-j}} \otimes \mathbb{1}^*_{n^j}.$$

From this form, it is possible to extract the following result.

Proposition 4.2.14. *If $u, v \in [n]^k$ with distance l in $B(n,k)$, then let $m, d \in \mathbb{Z}$, $w, z \in [n]^l$,$y \in [n]^{k-l}$, and $x \in [n]^d$ be defined as in 4.2.11. Then, by Theorem 4.2.13, we have*

$$F^{u\to *v}(t) = (t+n)^{n^k-k-1} + \sum_{j=0}^{m-1} t(t+n)^{n^k-l-jd-2}$$

and so, for $p = 1, \ldots, n^k$,

$$F_p^{u\to *v} = \binom{n^k-k-1}{p-1} n^{n^k-k-p} + \sum_{j=0}^{m-1} \binom{n^k-l-jd-2}{p-2} n^{n^k-l-jd-p}.$$

The above considerations suggest that a couple of generalized networks that might allow similar derivations of their forest matrices. The simplest of these involves the *shifted* de Bruijn graphs defined in [21]. These are simply graphs defined by the 0-1 matrices arising in the powers d of $A_{B(n,k)}$ where $d \leq k$. Additionally, the Kautz graphs [18] and wrapped butterfly graphs [14] possess related, but more complex Laplacian algebra relations and so should be amenable to the kind of approach employed above. Finally, there is a natural weighting of A given by

$$A(\omega) = \mathbb{1} \otimes I \otimes x^T$$

where x is a column vector of n indeterminates. The powers of $A(\omega)$ behave similarly to the powers of A, and yet the resulting algebra is much harder to work with. In general, it would be interesting to study how these weighted Laplacian algebras compare to their unweighted counterparts.

4.3 The Cartesian Product of Complete Graphs

In the previous example, we made use of a nearly cyclic generator for the Laplacian algebra in the form of the adjacency matrix. This allowed us to simplify the general equations determining $Q(t)$ enough to solve explicitly. In this next example, we will see that it is sometimes useful to find a larger matrix algebra containing the Laplacian algebra.

A nice example comes from Cartesian products of complete graphs. Let us fix $k, n_1, \ldots, n_k$ and let $G = K_{n_1} \times \cdots \times K_{n_k}$. As we saw in Section 3.2 we have

$$F_G(t) = \prod_{S \subseteq [k]} (t + n_S)^{\mu(S)},$$

where $\mu(S) = \prod_{i \in S}(i-1)$. This is a much more complicated forest polynomial than that of the de Bruijn graphs in the previous section. As such, we should expect that the forest matrix will be similarly more complicated. To facilitate our arguments, we will derive results for $k = 2$ first and then adapt this approach to the more general case.

Using Example 3.3.5 and the properties of the Cartesian product above, we find that $tI + L_{K_m \times K_n} = tI + (L_{K_m} \oplus L_{K_n}) = (t+m+n)I - J \otimes I - I \otimes J$. Note that $\{I, J \otimes I, I \otimes J, J\}$ forms a basis for a matrix algebra containing $L_{K_m \times K_n}$.

This follows from the fact that this set is linearly independent and that $(J \otimes I)^2 = m(J \otimes I)$, $(I \otimes J)^2$, $(J \otimes I)(I \otimes J) = (I \otimes J)(J \otimes I) = J$, $(J \otimes I)J = mJ$, and $(I \otimes J)J = nJ$. Since $Q_{K_m \times K_n}(t)$ belongs to this larger algebra, we have that

$$Q_{K_m \times K_n}(t) = \alpha(t)I + \beta(t)J \otimes I + \gamma(t)I \otimes J + \delta(t)J.$$

Combining this identity with the equation

$$(tI + L_{K_m \times K_n})Q_{K_m \times K_n}(t) = F_{K_m \times K_n}(t)I$$

from Proposition 3.3.7 we can attempt to solve for the coefficients α, β, γ, and δ.

By applying Proposition 3.2.5 we have that

$$F_{K_m \times K_n}(t) = t(t+m)^{m-1}(t+n)^{n-1}(t+m+n)^{mn-m-n+1}.$$

Together with the above multiplication rules, this results in

$$\begin{aligned}
\alpha(t)(t+m+n)I + (\beta(t)(t+m+n) - \alpha(t) - m\beta(t))J \otimes I \quad &+ \\
(\gamma(t)(t+m+n) - \alpha(t) - n\gamma(t))I \otimes J \quad &+ \\
(\delta(t)(t+m+n) - \beta(t) - \gamma(t) - m\delta(t) - n\delta(t))J \quad &= \quad F_{K_m \times K_n}(t)I.
\end{aligned}$$

This equation is readily solved by equating coefficients. Starting with the coefficient of the identity matrix and working to the right, we see that

$$\alpha \cdot (t+m+n) = F_{K_m \times K_n}(t)$$

so that

$$\alpha = t(t+m)^{m-1}(t+n)^{n-1}(t+m+n)^{mn-m-n}.$$

Now,

$$\beta \cdot (t+n) - \alpha = 0$$

so that

$$\beta = t(t+m)^{m-1}(t+n)^{n-2}(t+m+n)^{mn-m-n}$$

and similarly,

$$\gamma \cdot (t+m) - \alpha = 0$$

so that

$$\gamma = t(t+m)^{m-2}(t+n)^{n-1}(t+m+n)^{mn-m-n}.$$

Finally, we see that

$$\beta + \gamma = t(t+m)^{m-2}(t+n)^{n-2}(t+m+n)^{mn-m-n}(2t+m+n)$$

so that

$$\delta \cdot t - \beta - \gamma = 0$$

implies that

$$\delta = (t+m)^{m-2}(t+n)^{n-2}(t+m+n)^{mn-m-n}(2t+m+n).$$

Putting this all together and letting $\tilde{F} = (t+m)^{m-2}(t+n)^{n-2}(t+m+n)^{mn-m-n}$, we have found that

$$Q_{K_m \times K_n}(t) = \tilde{F}\,(t(t+m)(t+n)I + t(t+m)J \otimes I + t(t+n)I \otimes J + (2t+m+n)J)$$

From here, we can see that for any $u = (a,b), v = (c,d) \in V(K_n \times K_m)$, the v rooted forest polynomial is given by

$$\begin{aligned} F^{\to * v}(t) &= \alpha + \beta + \gamma + \delta \\ &= \tilde{F} \cdot (t(t+m)(t+n) + t(t+m) + t(t+n) + (2t+m+n)) \\ &= \tilde{F} \cdot \left(t^3 + (m+n+2)t^2 + (mn+m+n+2)t + (m+n)\right). \end{aligned}$$

On the other hand, $F^{u \to * v}(t)$ will depend on u and v with three separate cases to consider. Either u and u agree in their first coordinate and not the second, or they agree in their second coordinate and not the first, or they do not agree on either coordinate. In the last case, only J contributes to the forest polynomial and so $F^{u \to * v}(t) = \delta(t)$. In the first two cases, the polynomial depends on J and also on either $J \otimes I$ or $I \otimes J$ respectively. Thus we conclude that if a and b are distinct vertices of K_m and c and d are distinct vertices of K_n, then

$$F^{(a,b)\to*(a,d)}(t) = \gamma + \delta = \tilde{F} \cdot (t^2 + (m+2)t + m + n)$$

$$F^{(a,b)\to*(c,b)}(t) = \beta + \delta = \tilde{F} \cdot (t^2 + (n+2)t + m + n)$$

and

$$F^{(a,b)\to*(c,d)}(t) = \delta = \tilde{F} \cdot (2t + m + n.)$$

Note that $K_n \times K_m$ is vertex transitive but not arc transitive. Indeed, we can see that in this case each of the three different types of vertex pairs has a unique forest polynomial associated with it. This is in contrast to the case of the complete multipartite graphs above. In that case, we had distinct vertex pairs with identical forest polynomials.

The above computation can be generalized to cartesian products of arbitrary collections of complete graphs. While these graphs can have fairly complicated relations determined by their minimal polynomials, the matrices Q_k happen to lie in a larger matrix algebra with a convenient basis. The describe this, we first fix positive integers $n_1, \ldots, n_k$ and consider the graph $G = K_{n_1} \times \cdots \times K_{n_k}$.

The matrix algebra that we will consider is generated by all matrices

$$J_S = \bigotimes_{i=1}^{k} ((1 - \delta_{i \in S})\, I + \delta_{i \in S} J) \tag{4.8}$$

for each $S \subseteq [k]$ where the size of the jth term in the tensor product is n_j and δ is the Kronecker delta. This set of matrices is convenient because of a simple multiplication rule.

Lemma 4.3.1. *If $S, T \subseteq V$, then the matrices defined in equation 4.8 satisfy*

$$J_S J_T = \left(\prod_{j \in S \cap T} n_j \right) J_{S \cup T}$$

for all $S, T \subseteq [k]$.

Proof. Using the multiplicative property of the tensor product, we see that

$$J_S J_T = \bigotimes_{i=1}^{k} \left(\left(1 - \delta_{i \in S}\right) I + \delta_{i \in S} J \right) \left(\left(1 - \delta_{i \in T}\right) I + \delta_{i \in T} J \right).$$

If $i \in S \cap T$, then the ith product is $n_i J$. If $i \in S \Delta T$, the symmetric difference between S and T, then the ith product is simply J. Finally, if $i \notin S \cup T$, then the ith product is I. Using the scalar property of the tensor product and the fact that $(S \cap T) \cup (S \Delta T) = S \cup T$ we recover the lemma. $\square$

Let us label this set by $N(n_1, \ldots, n_k) = \{J_S\}_{S \subseteq [k]}$. Since G is regular, $L_G = n_{[k]} I - \sum_{i=1}^{k} J_{\{i\}}$ we see that $Q_G(t)$ lies in the span of $N(n_1, \ldots, n_k)$. Our strategy will be to follow the argument given above for the $k = 2$ case, however we should first verify that this set is indeed linearly independent.

Lemma 4.3.2. *For $k \geq 1$, the set $N(n_1, \ldots, n_k)$ is linearly independent.*

Proof. We argue by induction on k. When $k = 1$ or 2, we can verify directly that the sets $N(n_1)$ and $N(n_1, n_2)$ are linearly independent. Suppose then that $k > 2$ and we have a linear combination

$$\sum_{S \subseteq [k]} \beta_k J_S = 0.$$

We will show that each β_k must equal zero. We can relate members of $N(n_1, \ldots, n_k)$

to members of $N(n_1, \ldots, n_{k-1})$ by noting that, for a subset $S \subseteq [k]$, we have

$$J_S = \tilde{J}_{S-\{k\}} \otimes J$$

when $k \in S$ and

$$J_S = \tilde{J}_S \otimes I$$

when $k \notin S$. Note that we are using $\tilde{J}_S$ to distinguish members of $N(n_1, \ldots, n_{k-1})$.

From this observation, we have

$$\sum_{S \subseteq [k]} \beta_S J_S = \left(\sum_{S \subseteq [k-1]} \beta_S \tilde{J}_S \right) \otimes I + \left(\sum_{S \subseteq [k-1]} \beta_{S \cup \{k\}} \tilde{J}_S \right) \otimes J. \tag{4.9}$$

Now we assume, for our inductive hypothesis, that $N(n_1, \ldots, n_{k-1})$ is linearly independent. This means that either the sum $\sum_{S \subseteq [k-1]} \beta_S \tilde{J}_S$ is non-zero, or each $\beta_S = 0$ for $S \subseteq [k-1]$ and similarly for the sum $\sum_{S \subseteq [k-1]} \beta_{S \cup \{k\}} \tilde{J}_S$. Since, for any matrix, the products $A \otimes I$ and $A \otimes J$ are equal to zero just in case A is equal to zero, we see that either each of the two sums is equal to zero or neither is. To prove our claim, we need only show that the latter case is not possible.

Let us assume, by way of contradiction, that each sum is non-zero. Then, there is an entry of $\sum_{S \subseteq [k-1]} \beta_{S \cup \{k\}} \tilde{J}_S$ that is not zero. Let us say that it is the (a, b) entry. Now, since $n_k > 1$, we have two distinct integers x and y with $0 < x, y < n_k$. This means that the $((a, x), (b, y))$ entry of $\left(\sum_{S \subseteq [k-1]} \beta_{S \cup \{k\}} \tilde{J}_S \right) \otimes J$ is non-zero. On the other hand, this same entry of $\left(\sum_{S \subseteq [k-1]} \beta_S \tilde{J}_S \right) \otimes I$ must equal zero. This contradicts our initial assumption that the entire expression is equal to zero and it follows that each of the two right hand sums in (4.9) is equal to zero. By our inductive hypothesis, we can now conclude that $\alpha_S = 0$ for each $S \subseteq [k]$. $\square$

From Lemma 4.3.2, we can now establish $Q(t)$, with the help of some notation.

Definition 4.3.3. *Let U denote the set of all sums of distinct elements in $\{n_1, \ldots, n_k\}$, including the empty sum, and then $N(t) = \prod_{s \in U}(t+s)$. Given a set $S \subseteq [k]$, we let $\bar{S} = [k] - S$ and $n_S = \sum_{i \in S} n_i$. Finally, given an ordering σ of S, let $m_S^\sigma(t) = \prod_{i=0}^{|S|} t + n_{S_i}$ where $S_0 = \bar{S}$ and $S_i = S_{i-1} \cup \{\sigma(i)\}$ for $i > 0$. So, S_i is the set $\bar{S}$ with the first i elements of S under σ added back to it. Note that if $|S| = l$, then $S_l = [k]$.*

The following lemma is obvious from the preceding definition.

Lemma 4.3.4. *For any subset $S \subseteq [k]$ and any ordering σ of S,*

$$m_S^\sigma(t) \mid N(t).$$

Now, we are ready to establish the forest matrix for the cartesian product of complete graphs.

Theorem 4.3.5. *For $G = K_{n_1} \times \cdots \times K_{n_k}$, we have*

$$Q_G(t) = \tilde{F}_G(t) \sum_{S \subseteq [k]} M_S(t) J_S,$$

where $M_S(t) = \left(\sum_{\sigma \in \mathfrak{O}(S)} \frac{1}{m_S^\sigma(t)} \right) N(t)$ and $\tilde{F}_G(t) = \prod_{S \subseteq [k]} (t + n_S)^{\mu(S)-1}$ [4]*.*

Proof. From Lemma 4.3.2 we have that $Q_G(t) = \sum_{S \subseteq [k]} \alpha_S(t) J_S$ where $\alpha_S(t)$ is a

[4] note that μ is defined at the beginning of this section as well as in proposition 3.2.5

polynomial. Combining this with 3.1.13 and 3.3.7 we have

$$\left((t+n_{[k]})I-\sum_{i=1}^{k}J_i\right)\left(\sum_{S\subseteq[k]}\alpha_S(t)J_S\right)=F(t)I.$$

Note that $m_{-L}(t)=M_{[k]}(t)$ so that, if we set $\tilde{\alpha}_S(t)=\frac{\alpha_S(t)}{\tilde{F}_G(t)}$, we have

$$\left((t+n_{[k]})I-\sum_{i=1}^{k}J_i\right)\left(\sum_{S\subseteq[k]}\tilde{\alpha}_S(t)J_S\right)=N(t)I.$$

By 4.3.1, this becomes

$$(t+n_{[k]})\tilde{\alpha}_{\emptyset}I+\sum_{S\subseteq[k]}\left((t+n_{\bar{S}})\tilde{\alpha}_S(t)-\sum_{i\in S}\tilde{\alpha}_{S-\{i\}}(t)\right)J_S=N(t)I.$$

Thus, we find that $\tilde{\alpha}_{\emptyset}(t)=\frac{N(t)}{t+n_{[k]}}$ and $\tilde{\alpha}_S(t)=\frac{1}{t+n_{\bar{S}}}\sum_{i\in S}\tilde{\alpha}_{S-\{i\}}$. Extending this last recurrence yields the following

$$\begin{aligned}\tilde{\alpha}_S(t) &= \frac{1}{t+n_{\bar{S}}}\sum_{i\in S}\frac{1}{t+n_{\bar{S}+\{i\}}}\sum_{j\in S-\{i\}}\frac{1}{t+n_{\bar{S}+\{i,j\}}}\sum_{k\in S-\{i,j\}}\cdots\frac{1}{t+n_{\bar{\{z\}}}}\sum_{x\in S-(S-\{z\})}\tilde{\alpha}_{\emptyset}\\ &= \left(\sum_{\sigma\in\mathfrak{O}(S)}\frac{1}{m_S^{\sigma}(t)}\right)N(t)\end{aligned}$$

where $\mathfrak{O}(S)$ is the set of all orderings of the set S. □

Note that, by definition, the (a,b) entry of J_S is 1 just in case $T\subseteq S$ and 0 otherwise. This means that the forest polynomial $F_G^{a\to *b}(t)$ depends only on the set T.

Corollary 4.3.6. *If $a=(a_1,\ldots,a_k)$ and $b=(b_1,\ldots,b_k)$ are vertices of $G=K_{n_1}\times$*

$\cdots \times K_{n_k}$ *and* T *is the subset of* $[k]$ *defined by* $a_i \neq b_i$ *if and only if* $i \in T$*, then*

$$F_G^{a \to * b}(t) = \tilde{F}_G(t) \sum_{S: T \subseteq S} M_S(t).$$

Since the set of orderings of $[k]$ is equivalent to the set of permutations of $[k]$, the expression for $M_S(t)$ looks like the determinant of some $|S| \times |S|$ matrix. We will see in section 4.4 below that $Q_G(t)$ can be expressed in terms of the inverse of a matrix whose size is given by the size of a basis for $\mathcal{M}(L)$. Since this is generally much larger than k for $G = K_{n_1} \times \cdots \times K_{n_k}$, this would be a noteworthy fact. However, in general, it does not seem to be the case. To see this, note that the terms in the product $m_S^\sigma(t)$ are of the form $t + s$ where, taking in every possible ordering σ, s ranges over all distinct partial sums of U. This number might be as high as 2^k. On the other hand, there are at most k^2 entries in a $k \times k$ matrix. Without some constraints on the size of the set U, we cannot hope to construct a $k \times k$ matrix with the desired determinant.

On the other hand, in the extreme case of the Hamming graphs $H(c, k)$, where each n_i is equal to some integer c, the set U has exactly $k + 1$ elements, namely, $U = \{0, c, 2c, \ldots, kc\}$. In this case, we have $N(t) = \prod_{j=0}^{k}(t + jc)$ and, for any $S \subseteq [k]$ and ordering σ of S, we see that for S_i, as defined in Definition 4.3.3, $n_{S_i} = (k - |S| + i)c$ so that $m_S^\sigma(t)$ depends only on i and $|S|$. As a result, the forest polynomials of these graphs simplify greatly.

Corollary 4.3.7. *If* $G = H(c, k)$ *is a Hamming graph and* a, b *are vertices of* G *with*

T defined as in Corollary 4.3.6 with $|T| = l$, then

$$F_G^{a\to *b}(t) = \tilde{F}_G(t) \sum_{j=0}^{k-l} \binom{k-l}{j} (j+l)! \left(\prod_{i=0}^{k-l-j-1} (t+ic) \right).$$

Proof. Let S be a subset of $[k]$. Applying the observation in the previous paragraph, we see that

$$\begin{aligned} \sum_{\sigma\in\mathfrak{O}(S)} \frac{1}{m_S^\sigma(t)} &= \sum_{\sigma\in\mathfrak{O}(S)} \prod_{j=0}^{|S|} \frac{1}{(t+(k-|S|+j)c)} \\ &= \frac{|S|!}{\prod_{j=0}^{|S|} (t+(k-|S|+j)c)} \end{aligned}$$

and therefore, $\tilde{\alpha}_S = |S|! \prod_{j=0}^{k-|S|-1} (t+jc)$. Now, there are exactly $\binom{k-l}{j}$ subsets S containing T with size $j+l$. □

4.4 The Laplacian Matrix Algebra and the Univariate Forest Matrix

In the previous two sections, we have made use of a convenient description of the Laplacian matrix algebra to derive a formula for the entries of the matrix $Q(t)$. In each case, this resulted in a system of equations that could be solved directly. In general, we can relate these system of equations to a matrix equation.

Proposition 4.4.1. *If L belongs to a matrix algebra with basis $\{B_1, \ldots, B_k\}$ satisfying*

$$LB_i = \sum_{j=1}^{k} p_{ij} B_j, \quad I = \sum_{i=1}^{k} q_i B_i, \quad \text{and} \quad Q(t) = \sum_{i=1}^{k} \alpha_i(t) B_i$$

then

$$(tI + P^T)\alpha(t) = F_G(t)q$$

where $P_{ij} = p_{ij}$*, while* $\alpha(t)$ *and* q *are the column vectors determined by the* α_i *and the* q_i *respectively.*

Proof. From Proposition 3.3.7 and our assumed representations of Q and I, we have

$$(tI + L)\sum_{i=1}^{k} \alpha_i B_i = F \sum_{i=1}^{k} q_i B_i,$$

or, after expanding each product LB_i and collecting like terms,

$$\sum_{i=1}^{k} \left(t\alpha_i + \left(\sum_{j=1}^{k} p_{ji}\alpha_j \right) - Fq_i \right) B_i = 0.$$

Since each coefficient on the left must equal zero, we can assemble the vector equation

$$(tI + P^T)\alpha(t) = Fq,$$

as desired. □

4.5 The Minimal Polynomial and Lazy Random Walks

The simplest way to characterize $\mathcal{M}(L)$ is through the minimal polynomial $m_L(t)$. We will instead use the closely related minimal polynomial of $-L$ since it divides $F(t)$ and has non-negative coefficients. As such, we set $m(t) = m_{-L}(t)$. If we let $d = \deg(m)$ and $m = t^d + \sum_{i=0}^{d-1} m_i t^i$, then we know that $\{I, L, L^2, \ldots, L^{d-1}\}$ is a basis for $\mathcal{M}(L)$ with $L^d = \sum_{i=0}^{d-1}(-1)^{d-i-1} m_i L^i$. Applying Proposition 4.4.1 we see that P^T is the companion matrix of m so that α may be solved for directly. Now, it will be convenient to define a polynomial that we will show is a common factor of all of the univariate $i \to *j$ forest polynomials of G.

Definition 4.5.1. *Let G be a graph with univariate forest polynomial $F(t)$ and let $m(t) = m_{-L}(t)$. Then we define*

$$\tilde{F}(t) = \frac{F(t)}{m(t)}.$$

Proposition 4.5.2. *With the assumptions of the previous paragraph and supposing that $Q(t) = \sum_{i=1}^{d} \alpha_i L^{i-1}$, then*

$$\alpha_i = (-1)^i \tilde{F}(t) \cdot \left(\sum_{j=0}^{d-i} m_{i+j} t^j \right)$$

where we take m_d to be 1.

Proof. With our assumptions, the matrix equation in Proposition 4.4.1 has the form

$$\begin{bmatrix} t & 0 & \dots & 0 & (-1)^{d-1} m_0 \\ 1 & \ddots & \ddots & \vdots & \vdots \\ 0 & \ddots & \ddots & 0 & \vdots \\ \vdots & & \ddots & t & -m_{d-2} \\ 0 & \dots & 0 & 1 & t + m_{d-1} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_d \end{bmatrix} = \begin{bmatrix} F(t) \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

By multiplying each row i by t^{i-1} and then, for $i = 1, \dots, d-1$, successively adding the negative of each row i to row $i+1$ this system of equations is equivalent to

$$\begin{bmatrix} t & 0 & \dots & 0 & (-1)^{d-1}\bar{m}_0(t) \\ 0 & t^2 & \ddots & \vdots & (-1)^{d-2}\bar{m}_1(t) \\ \vdots & \ddots & \ddots & 0 & \vdots \\ \vdots & & \ddots & t^{d-1} & -\bar{m}_{d-2}(t) \\ 0 & \dots & \dots & 0 & t^d + \bar{m}_{d-1}(t) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_d \end{bmatrix} = F(t) \begin{bmatrix} 1 \\ -1 \\ \vdots \\ (-1)^{d-1} \end{bmatrix}$$

where $\bar{m}_k(t) = \sum_{i=0}^{k-1} m_i t^i$. Since $t^d + \bar{m}_{d-1}(t) = m_{-L}(t)$, the bottom row implies that

$$\alpha_d(t) = (-1)^{d-1}\tilde{F}$$

and in general,

$$\alpha_i(t) = (-1)^i \frac{F(t) - \tilde{F}\bar{m}_i(t)}{t^i}.$$

□

One important feature of the Laplacian matrix is that it can be translated into a non-negative matrix $B = \Delta I - L$. One reason to prefer the matrix B over the Laplacian matrix L is that we can make use of the theory of non-negative matrices. The powers of B also have a simple combinatorial interpretation. That is, $\left(B^l\right)_{i,j}$ counts the number of length l walks from i to j in the modified network in which each vertex u has exactly $\Delta - d_s(u)$ loops added. These walks are related to lazy random walk Markov processes [2] with the probability of remaining at a given vertex v equalling $\frac{\Delta - d_s(v)}{\Delta}$. We will call these Δ-lazy walks.

Definition 4.5.3. *Let $w_{ij}^{(k)} = \left(B^l\right)_{i,j}$ denote the number of Δ-lazy walks in G.*

Theorem 4.5.4. *With the same assumptions as in Proposition 4.5.2 and B defined as above, for $i, j \in V$,*

$$F^{i\to *j}(t) = \tilde{F}(t) \sum_{l=0}^{d-1} \left(\sum_{p=0}^{d-l} (-1)^p \eta(l,p) w_{ij}^{(p)} \right) t^l$$

where $\eta(l,p) = \sum_{k=0}^{d-l-p} \Delta^k \binom{k+p}{p} m_{k+l+p}$.

Proof. Substituting $L = \Delta I - B$ into Proposition 4.5.2 and applying the binomial theorem, we have

$$\begin{aligned} Q(t) &= \tilde{F}(t) \sum_{k=1}^{d} \left[(-1)^k \left(\sum_{l=0}^{d-k} m_{k+l} t^l \right) \left(\sum_{p=0}^{k} \binom{k}{p} (-1)^p \Delta^{k-p} B^p \right) \right] \\ &= \tilde{F}(t) \sum_{l=0}^{d-1} \left[\sum_{k=1}^{d-l} \sum_{p=0}^{k} (-1)^k m_{k+l} \binom{k}{p} (-1)^p \Delta^{k-p} B^p \right] t^l \\ &= \tilde{F}(t) \sum_{l=0}^{d-1} \left[\sum_{p=0}^{d-l} (-1)^p \left(\sum_{k=0}^{d-l-p} \Delta^k \binom{k+p}{p} m_{k+l+p} \right) B^p \right] t^l \end{aligned}$$

where the last line is the result of interchanging the p and k summations and then substituting $k+p$ for k. $\square$

This formula allows us to compute the lth coefficient of $F^{i\to *j}(t)$ from $\{m_l, \ldots, m_d\}$ as well as $w_{ij}^{(p)}$ for $p = 0, \ldots, d-l$. Thus the larger l gets, we see Q_{l+1} containing less information in general about G.

5 Tractable Multivariate Forest Polynomials

In this final chapter we return to multivariate forest polynomials. Our basic interest is to find examples edge and vertex variable specializations that lead to forest polynomials with factored, or at least compact, expressions. We will consider a few graph families that posses such specializations explicitly and then turn our attention to graphs that posses what we call externally equitable partitions. We find that the presence of such a partition Π ensures that a specialization of $F(\tau, \omega)$ can be factored into at least two nontrivial pieces, namely, the forest polynomial of a quotient graph and another factor that we name $h_{G,\Pi}$. Then we define a further condition a prove a theorem that ensures that h can be can be written as a product of related graph polynomials. We close the chapter by applying this theorem to a few examples and show that it generalizes a number of further results in the recent graph theory literature.

5.1 Reducible and Nearly Reducible Examples

We begin this section with some examples of graphs whose multivariate forest generating functions have simple closed expressions. The easiest of these are graphs with many transient edges, as illustrated in the following lemma.

Lemma 5.1.1. *If e is a transient edge, then*

$$F_G^{\to *}(\tau, \omega) = F_{G-e}^{\to *}(\tau + \omega_e \mathbb{1}_{s(e)}, \omega).$$

Proof. The forests in $\mathcal{F}_{G-e}^{\to *}$ are in one to one correspondence with the forests in $\mathcal{F}_G^{\to *}$ that do not contain edge e. On the other hand, the forests in $\mathcal{F}_G^{\to *}$ that do contain edge e are in one to one correspondence with the forests in $\mathcal{F}_{G-e}^{\to * s(e)}$. That is,

$$F_G^{\to *}(\tau, \omega) = \mathcal{F}_{G-e}^{\to *}(\tau, \omega) + \omega^e \mathcal{F}_{G-e}^{\to * s(e)}(\tau, \omega).$$

The result now follows from the multilinearity of F_{G-e} and $F_{G-e}^{\to * s(e)}$. $\square$

Since every edge in an acyclic graph is transient, we can repeatedly apply Lemma 5.1.1 to obtain the following result.

Proposition 5.1.2. *If G is an acyclic graph, then*

$$F(\tau,\omega) = \prod_{v\in V} (\tau_v + \phi_v(\omega))$$

and

$$F^{i\to *j}(\tau,\omega) = \sum_{p\in P(i,j)} \omega^p \prod_{v\in V-V(p)} (\tau_v + \phi_v(\omega)),$$

where $P(i,j)$ denotes the set of directed paths in G from i to j and $\phi_v(\omega) = \sum_{e\in s^{-1}(v)} \omega_e$.

Said another way, in an acyclic graph, every spanning functional digraph is also a spanning forest. Using this proposition we can, in principal, assemble the forest matrix $Q(\tau,\omega)$ although it will not generally factor any farther without some special structure on the part of G.

We say that a graph G has k disjoint directed cycles if there is a partition $V = U + C_1 + \cdots + C_k$ so that the graph induced on U is acyclic and the graph induced on each C_i is a directed cycle. If G is not acyclic, but if its cycles are disjoint, then the forest generating function and the generating function for functional digraphs are almost equal.

Proposition 5.1.3. *If G has a unique directed cycle C then*

$$F(\tau,\omega) = \left(\prod_{v\in V-V(C)} (\tau_v + \phi_v(\omega))\right)\left(\prod_{v\in V(C)} (\tau_v + \phi_v(\omega)) - \omega^C\right)$$

This can also be easily generalized.

Proposition 5.1.4. *If G has exactly k disjoint directed cycles $C_1, \ldots, C_k$ then*

$$F(\tau,\omega) = \left(\prod_{v\in U} (\tau_v + \phi_v(\omega))\right) \prod_{j=1}^{k} \left(\prod_{v\in V(C_j)} (\tau_v + \phi_v(\omega)) - \omega^{C_j}\right).$$

Notice that these compact expressions do not always involve an explicit factorization. For example, a single directed cycle will have an irreducible forest polynomial, by virtue of Proposition 2.2.11, but can still be compactly represented because it *nearly* factors. Another example of this phenomena comes from the complete graph with the target uniform edge weighting defined in Section 2.3.

Proposition 5.1.5. *For complete graph $G = K_n$ with vertex set V, we let $x = (x_i)_{1\le i\le n}$ with $|x| = x_1 + x_2 + \cdots + x_n$. Then,*

$$F_G(\tau, Tx) = \prod_{i\in V}(\tau_i + |x|) - \sum_{i=1}^{n} x_i \prod_{j\in V-\{i\}} (\tau_j + |x|).$$

Proof. From $L_{K_n}(\tau, Tx) = D(\tau + |x|\mathbb{1}) - \mathbb{1}x^T$, we apply lemma 4.1.4 and 4.1.1. $\square$

Organizing the monomials in $F(\tau, Tx)$ around their τ variables generalizes a result from Moon [45].

Corollary 5.1.6.

$$F(\tau, Tx) = \sum_{S\subseteq V} |x|^{n-1-|S|} \left(\sum_{i\in S} x_i\right) \tau^S.$$

We can also specialize Proposition 5.1.5 by identifying certain of the τ variables. This results in a rather compelling factorization that we will explain in the following two sections.

Proposition 5.1.7. *Let $G = K_n$ and Π be any partition of V_G with $|\Pi_i| = c_i$, then*

$$F_G(\Pi\tau, \mathbb{1}) = \left(\prod_{i=1}^{k}(\tau_i + n)^{c_i - 1}\right)\left(\prod_{i=1}^{k}(\tau_i + n) - \sum_{i=1}^{k} c_i \prod_{\substack{j=1 \\ j\neq i}}^{k}(\tau_j + n)\right).$$

Note that the rightmost factor bears a striking resemblance to the formula in Proposition 5.1.5, but with k replacing n.

In fact, the factors in the left hand product are also related to the weighted *univariate* forest polynomial of K_n.

Corollary 5.1.8. *With $G = K_n$, $\omega = Tx$ as in proposition 5.1.5, and $\tau = t\mathbb{1}$,*

$$F_G(t, Tx) = t(t + |x|)^{n-1}$$

For now, the relation between these different factors will remain mysterious, however, we will account for each of them in the main theorem of section 5.3. The next section will prepare us for the proof of this theorem.

5.2 Externally Equitable Vertex Partitions and Graph Quotients

In what follows, we will take S to be a set of the vertices of a graph and assume Π has some special structure.

Definition 5.2.1. *Let G be a graph and Π be a partition of V. We say that Π is* Externally Equitable *if, for any i, j with $i \neq j$ and $v \in \Pi_i$, the number $|N^t(v) \cap \Pi_j|$ depends only on i and j. In other words, every member of Π_i has the same number of neighbors in Π_j. In this case, we let c_{ij} denote this number.*

Note that this definition imposes no constraint on the structure of G inside of Π_i.

Definition 5.2.2. *Let Π be an externally equitable partition of $V(G)$. Then Π determines an edge partition Ψ defined first by setting each edge e with both $s(e)$ and $t(e)$ contained in the same cell of Π into its own singleton cell $\Psi_e = \{e\}$. For each i, j with $i \neq j$ and $v \in \Pi_i$, we apply some ordering to $s^{-1}(v) \cap t^{-1}(\Pi_j) = \{e_1^{(v,j)}, e_2^{(v,j)}, \ldots, e_{c_{ij}}^{(v,j)}\}$. This ordering can be arbitrarily chosen, however, unless otherwise specified, we assume the induced ordering described in Section 0.1. With this in place, we can define the remaining cells of Ψ by $\Psi_{ijl} = \{e_l^{(v,j)}\}_{v \in \Pi_i}$ for each $1 \leq i, j \leq k$ and $1 \leq l \leq c_{ij}$. If an edge is wholly contained in a cell of Π, then it is* Π-internal. *Otherwise it is* Π-external.

The cells of Ψ can be naturally viewed as union of the set of Π internal edges of G and the set of edges of $\tilde{G}$.

Definition 5.2.3. *If G is a graph and Π is an externally equitable partition of V_G, then the* quotient graph *$\tilde{G}$, induced by Π has vertex set Π with an edge set Ψ. Singleton sets $\{e\}$ of Π-internal edges, e, are loops with*

$$s(\{e\}) = t(\{e\}) = \Pi(s(e))$$

and for a set of Π-external edges, Ψ_{ijl}, we have $s(\Psi_{ijl}) = \Pi_i$ and $t(\Psi_{ijl}) = \Pi_j$.

In general, $\tilde{G}$ has multiple edges and loops. Note that, as we have defined things,

G and $\tilde{G}$ are both independent of any particular edge weight assignments. Since loops play no role in our analysis below, we will mostly ignore those edges. We include them in the definition of $\tilde{G}$ because they allow us to state certain results more cleanly below.

Lemma 5.2.4. *For any partition Π of $[n]$ into k parts with associated variable $\tau = (\tau_1, \ldots, \tau_k)^T$, we have*

$$D(\Pi\tau)\Pi = \Pi D(\tau).$$

Proof. Both matrices are $n \times k$ with i, j entry equal to τ_j just in case $i \in \Pi_j$. $\square$

Lemma 5.2.5. *If Π is an externally equitable partition of G with Ψ and $\tilde{G}$ defined as above, then the following hold.*

(i) $S_G\Pi = \Psi S_{\tilde{G}}$

(ii) $T_G\Pi = \Psi T_{\tilde{G}}$

(iii) $\Psi^T S_G = S_{\tilde{G}}\Pi^T$

Proof. For item (i), we can check that

$$(S_G\Pi)_{e\tilde{v}} = \sum_{v\in V(G)} (S_G)_{ev}\,\Pi_{v\tilde{v}}.$$

This is clearly 1 if $s_G(e) \in \tilde{v}$. The same is obviously true of

$$(\Psi S_{\tilde{G}})_{e\tilde{v}} = \sum_{\tilde{e}\in E(\tilde{G})} (S_{\tilde{G}})_{\tilde{e}\tilde{v}}\,\Psi_{e\tilde{e}}.$$

The other two equations follow similarly. $\square$

Example 5.2.6. Let us consider the cube graph Q_3. Here we will represent its bi-directed edge pairs as undirected edges.

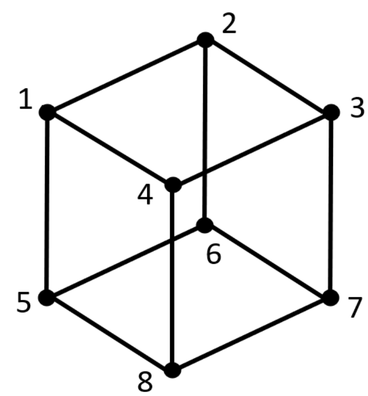


Figure 5.6: The cube graph Q_3 with vertices labeled by [8].

Now, we will consider the partition $\Pi = \{\{1,2\},\{3,4\},\{5,6\},\{7,8\}\}$ so that, for example, $\Pi_2 = \{3,4\}$. In Figure 5.7, we label each vertex by the index of its cell in Π. We will ignore the dotted lines for now. The reader can verify that Π is an EEP with quotient pictured in Figure 5.8.

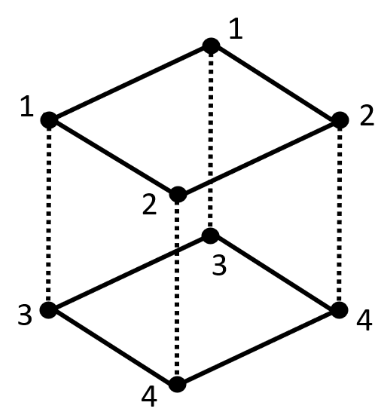


Figure 5.7: Q_3 with vertices labeled according to a partition.

Proposition 5.2.7. *With $G, \tilde{G}, \Pi$, and Ψ as above, and τ, ω indexed over Π and Ψ*

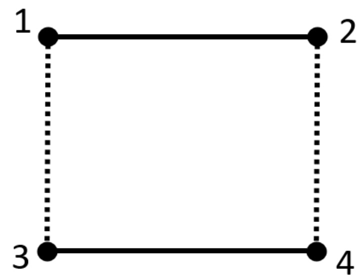


Figure 5.8: The quotient graph from the partition in Figure 5.7.

respectively, we have

$$L_G(\Pi\tau, \Psi\omega)\Pi = \Pi L_{\tilde{G}}(\tau, \omega).$$

Proof. Making use of each of the identities in Lemmas 5.2.5 and 5.2.4 we can compute directly using Definition 3.1.1, as follows:

$$\begin{aligned}
L_G(\Pi\tau, \Psi\omega)\Pi &= \left(D(\Pi\tau) + S_G' D(\Psi\omega)(S_G - T_G)\right)\Pi \\
&= \Pi D(\tau) + S_G' D(\Psi\omega)\Psi(S_{\tilde{G}} - T_{\tilde{G}}) \\
&= \Pi D(\tau) + S_G' \Psi D(\omega)(S_{\tilde{G}} - T_{\tilde{G}}) \\
&= \Pi D(\tau) + \Pi S_{\tilde{G}}' D(\omega)(S_{\tilde{G}} - T_{\tilde{G}}) \\
&= \Pi L_{\tilde{G}}(\tau, \omega).
\end{aligned}$$

□

Note that evaluating the second argument of L_G at $\Psi\omega$ assigns ω variables to the edges of G based on the cells of Ψ. To facilitate our discussion, we will need some conventional way of referring to the ω variable assigned to members of Ψ_{ijl}. Technically, this is an edge variable from $\tilde{G}$, however there will generally be c_{ij} differently weighted edges in $\tilde{G}$ pointing from Π_i to Π_j. To keep track of this we will say that members of Ψ_{ijl} will be weighted with variable $\omega_l^{i,j}$.

Example 5.2.8. In Figure 5.9, we see an example of the labeling scheme imposed by Π and Ψ on the edges pointing from a member of Π_i to three members of Π_j.

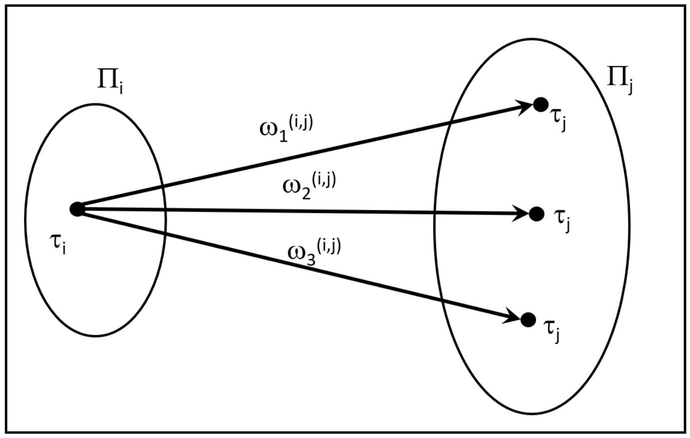


Figure 5.9: Illustration of the edge and vertex variables assigned by $\Pi\tau$ and $\Psi\omega$.

For the remainder of this section we will assume that G is a graph on with partition Π into k parts and that τ and ω are variables indexed by Π and Ψ respectively.

Theorem 5.2.9. *If Π is an externally equitable partition of G with quotient $\tilde{G}$, then*

$$F_{\tilde{G}}(\tau, \omega) \mid F_G(\Pi\tau, \Psi\omega).$$

Proof. Let $L = L_G(\Pi\tau, \Psi\omega)$ and $\tilde{L} = L_{\tilde{G}}(\tau, \omega)$. Since the set of columns of matrix Π is linearly independent, it can be extended to a basis for $\mathbb{C}[\tau, \omega]^n$ by adding $n - k$ additional independent vectors. Let us suppose that $n \times n - k$ matrix X . Since

$D_\Pi^{-1}\Pi^T\Pi = I$, there exists a matrix Y so that

$$[\Pi D_\Pi^{-1} \mid Y]^T = [\Pi \mid X]^{-1}.$$

In that case, we have $Y^T X = I$, $Y^T\Pi = 0$, and $\Pi^T X = 0$. Let $W = [\Pi \mid X]$ and then consider

$$W^{-1}LW \quad = \quad \begin{bmatrix} D_\Pi^{-1}\Pi^T L\Pi & D_\Pi^{-1}\Pi^T LX \\ Y^T L\Pi & Y^T LX \end{bmatrix}$$

Now, Proposition 5.2.7 ensures that $Y^T L\Pi = Y^T\Pi\tilde{L}(\tau,\omega) = 0$ and $D_\Pi^{-1}\Pi^T L\Pi = \tilde{L}$. Since $\det W^{-1}LW = \det L$, we conclude that $\det L = \det\tilde{L}\det Y^T LX$. It remains to verify that $\det Y^T LX$ is a polynomial. This follows from the fact that every entry of Π is constant. Thus, X and Y can be chosen to extend the columns of Π to a basis of $\mathbb{C}^n$. Then, their entries are elements of $\mathbb{C}$ and so the entries of $Y^T LX$ are still polynomials in ω and τ. $\square$

Theorem 5.2.9 shows that, when Π is externally equitable, $F_G(\Pi\tau,\Psi\omega)$ factors into a product of polynomials, one of which is another forest generating function. The proof does not make clear if there might be a similar interpretation for the quotient $\frac{F_G(\Pi\tau,\Psi\omega)}{F_{\tilde{G}}(\tau,\omega)}$. We will see below that this is indeed the case when Π satisfies an additional property. For now, let us give this quotient a name and derive a simple corollary from Theorem 5.2.9.

Definition 5.2.10. *If Π is an externally equitable partition of G with quotient $\tilde{G}$, then we let*

$$h_{G,\Pi}(\tau,\omega) = \frac{F_G(\Pi\tau,\Psi\omega)}{F_{\tilde{G}}(\tau,\omega)}.$$

Corollary 5.2.11. *If Π is an externally equitable partition of G, then*

$$Q_G(\Pi\tau, \Psi\omega)\Pi = h_{G,\Pi}(\tau,\omega)\Pi Q_{\tilde{G}}(\tau,\omega).$$

Proof. Starting with Proposition 5.2.7, we multiply by $Q_G(\Pi\tau, \Psi\omega)$ on the right and by $Q_{\tilde{G}}(\tau,\omega)$ on the left. The result now follows from Corollary 3.1.14. $\square$

It will be convenient for us to settle on a particular choice of X and Y from Theorem 5.2.9 for the argument that follows.

Definition 5.2.12. *Given a partition Π of $[n]$ into k parts with $|\Pi_i| = p_i$, suppose that $\Pi_i = \{v(i,1), \ldots, v(i,p_i)\}$. Then we define the $n \times (p_i - 1)$ matrix Z_i whose jth column is equal to a difference of standard basis vectors $e_{v(i,j)} - e_{v(i,j+1)}$. Then, we define the $n \times n - k$ matrix given by*

$$X_\Pi = [Z_1 \mid Z_2 \mid \cdots \mid Z_k].$$

The matrix Y_Π is determined by the relations $Y_\Pi^T X_\Pi = I$ and $Y^T\Pi = 0$. But we can also give an explicit description. Again, we define an $n \times (p_i - 1)$ matrix U_i by

$$(U_i)_{j,l} = \begin{cases} 1 - {}^{l}\!/_{p_i} & j = v(i,m) \text{ and } m \le l \\ -{}^{l}\!/_{p_i} & j = v(i,m) \text{ and } m > l \\ 0 & j \notin \Pi_i \end{cases}$$

Now we can assemble

$$Y_\Pi = [U_1 \mid U_2 \mid \cdots \mid U_k].$$

When no confusion is likely, we will drop subscripts and refer simple to the matrices X and Y.

Proposition 5.2.13. *For partition Π, the matrices X and Y satisfy $\Pi^T X = 0$, $Y^T \Pi = 0$, and $Y^T X = I$*

Proof. The ij entry of $\Pi^T X$ is the inner product of the ith column of Π with the jth column of X. By definition, the jth column of X has two non-zero entries corresponding to a pair of vertices both belonging to some cell of Π. If this cell is not Π_i, then the two vectors have disjoint support. Otherwise, Π_i has entry 1 at both vertices and the inner product is obviously 0.

Indeed, Π^T times any vector whose entries sum to zero on each cell of Π will be zero. So, we can establish our second claim by verifying that Y has this property as well. By definition, the support of each column is contained in a single cell of Π. Let us consider column l of U_i. From the definition, there are l entries in rows $v(i,1),\ldots,v(i,l)$ and then $p_i - l$ entries in rows $v(i,l+1),\ldots,v(i,p_i)$. Adding up the corresponding entries, we have

$$l\left(1 - \frac{l}{p_i}\right) + (p_i - l)\frac{-l}{p_i} = 0.$$

Finally, the ij entry of $Y^T X$ is the inner product of column i from Y and column j from X. Note that a column from U_l and Z_m will have disjoint support unless $l = m$. Thus, it suffices to show that, for each m, $U_m^T Z_m = I$. The inner product of column l of U_m with column j of Z_m is given by

$$\left(1 - \frac{l}{p_m}\right) - \left(1 - \frac{l}{p_m}\right) = 0 \text{ when } j < l,$$

$$\left(-\frac{l}{p_m}\right) - \left(-\frac{l}{p_m}\right) = 0 \text{ when } j > l, \text{ and}$$

$$\left(1-\frac{l}{p_m}\right) - \left(-\frac{l}{p_m}\right) = 1 \text{ when } j = l.$$

□

It will be useful to work with an ordering of the vertices of G that is compatible with Π in a particular sense given in the following lemma.

Lemma 5.2.14. *Let G be a graph and let Π be a partition of G. If the vertices of G are ordered so that $\Pi_1 = \{1, \ldots, p_1\}$, and for $i > 1$, $\Pi_i = \{p_1 + \cdots + p_{i-1} + 1, \ldots, p_1 + \cdots + p_i\}$, then Π, X, and Y are all block diagonal matrices.*

Proof. The proposed ordering ensures that vertices in the same cell lie in adjacent rows of each matrix. Thus,

$$\Pi = [\delta_{ij}\mathbb{1}_{p_i}]_{1\leq i,j\leq k}\,.$$

Similarly, the support of each column of U_i and Z_i are contained entirely in Π_i. Thus if $\hat{U}_i$ is the $p_i \times p_{i-1}$ submatrix of U_i consisting of only those rows indexed by vertices contained in Π and $\hat{Z}_i$ is defined similarly, then

$$X = \left[\delta_{ij}\hat{U}_i\right]_{1\leq i,j\leq k} \text{ and } Y = \left[\delta_{ij}\hat{Z}_i\right]_{1\leq i,j\leq k}.$$

□

This notion of compatibility is useful enough to make into an explicit definition.

Definition 5.2.15. *A graph G with partition Π has a* compatible vertex order *if its vertices satisfy the conditions of Lemma 5.2.14*

Of course, the matrix $L(\Pi\tau, \Psi\omega)$ can also be expressed as a $k \times k$ block matrix with ij block having size $p_i \times p_j$. With a compatible ordering of vertices as defined above, we get some special structure for the off diagonal blocks.

Lemma 5.2.16. *If G has an EEP, say, Π, and a compatible vertex order with $L(\Pi\tau, \Psi\omega) = [C_{ij}]_{1\leq i,j\leq k}$, then for $i \neq j$,*

$$C_{ij}\mathbb{1} = \left(\sum_{l=1}^{c_{ij}} \omega_l^{ij}\right)\mathbb{1}.$$

Proof. From our Definition 5.2.2, if $i \neq j$, then each row of C_{ij} contains exactly the ω variables $\omega_1^{ij}, \ldots \omega_{c_{ij}}^{ij}$. □

It will also be useful to characterize the diagonal blocks of the matrix as well. The following lemma does not require any special structure for Π. To minimize the need for cumbersome and unnecessary indices, we will employ the following convention.

Let x be a vector whose entries are indexed over some set S and suppose that $g(y)$ is a polynomial in y where y is a vector of variables whose entries are indexed of a set T with $T \subseteq S$. Then, we will allow ourselves to evaluate g at the vector x by simply evaluating g at the vector $(x_i)_{i\in T}$. Thus, in particular, given a subgraph H of G, we can assert that τ and ω are indexed by V_G and E_G respectively and still compare $L_G(\tau, \omega)$ and $L_H(\tau, \omega)$ without needing to introduce new variables.

Lemma 5.2.17. *Let G have partition Π and a compatible vertex order with C_{ij} defined as in the previous lemma. Then,*

$$C_{ii} = L_{G_i}(\tau_i\mathbb{1} + d_i(\omega), \omega)$$

where $G_i = G\,|_{\Pi_i}$*, and* $d_i(\omega) = \sum_{j \neq i} C_{ij}\mathbb{1}$*.*

Proof. Let $v, w \in \Pi_i$. From Remark 3.1.2, the v, w entry of C_{ii} is equal to $-\omega^e$ if $v \neq w$ and $e = v \to w$ is an edge of G and 0 otherwise. If $v = w$, then

$$(C_{ii})_{vv} = \tau_i + \sum_{e \in s_G^{-1}(v)} \omega_e = \tau_i + \sum_{e \in s_{G_i}^{-1}(v)} \omega_e + \sum_{e \in U} \omega_e$$

where U is the edge set $s_G^{-1}(v) - s_{G_i}^{-1}(v)$ of edges whose source is v and whose target lies outside of Π_i. Now, $\tau_i + \sum_{e \in s_{G_i}^{-1}(v)} \omega_e$ is the vv entry of $L_{g_i}(\tau_i\mathbb{1}, \omega)$ and $\sum_{e \in U} \omega_e$ is precisely the v entry of the vector $\sum_{j \neq i} C_{ij}\mathbb{1}$. $\square$

Finally, establish an important relation between the polynomial h of G and some of its subgraphs. For this, we require a straightforward but useful lemma.

Lemma 5.2.18. *If* G *has EEP* Π *and* $S \subseteq \Pi$ *is a collection of cells of* Π*, then* S *is an EEP of* $G\,|_{\bigcup S}$*.*

Proof. Consider cells S_i and S_j of S and let $v \in S_i$. The number of neighbors of v in S_j is unchanged whether S_j is thought of as a subset of vertices of G or $G\,|_{\bigcup S}$. $\square$

Proposition 5.2.19. *Let* G *be a graph with EEP* Π *and a compatible vertex order. Let* X *and* Y *be defined as in Definition 5.2.12. Let* H *be a subgraph of* G *whose vertex set is a union of cells of* Π*. Let* Θ *be the partition of the vertices of* H *induced by* Π *and let* K *denote the set of indices* $j \in [k]$ *so that* Π_j *is a cell of* Θ*. Then,*

$$h_{H,\Theta}(\tau, \omega) = \det \hat{Y}^T L_H(\tau, \omega) \hat{X}$$

where $\hat{X}$ and $\hat{Y}$ are given by

$$\hat{X} = \left[\delta_{ij}\hat{U}_i\right]_{i,j\in K} \text{ and } \hat{Y} = \left[\delta_{ij}\hat{Z}_i\right]_{i,j\in K}.$$

Proof. In light of Lemma 5.2.18, Θ is an EEP of H. Therefore, our proof need only show that $\hat{X}$ and $\hat{Y}$ satisfy the definition of X and Y for the partition Θ of H.

In fact, this is clear from the definitions of $\hat{U}_i$ and $\hat{Z}_i$ whose nonzero blocks depend only on the size of each cell Π. $\square$

5.3 The Forest Polynomial Quotient Factorization Theorem

In this section, we consider a special class of EEPs that allow us to factor $h_{G,\Pi}$ into a product of ratios of forest polynomials of subgraphs of G and $\tilde{G}$.

Definition 5.3.1. *Let $\Pi = \{\Pi_1, \ldots, \Pi_k\}$ be a partition of G into k parts. Then, we say Π has a* target uniform urdering *(TUO) if there is a partial ordering $\preceq$ of $[k]$ so that $j \preceq i$ implies that, for each $v \in \Pi_i$, the set $N^t(v) \cap \Pi_j$ does not depend on v. The ordering $\preceq$ induces an ordered partition of Π which we label $\Gamma = \{\Gamma_1, \ldots, \Gamma_r\}$.*

Example 5.3.2. Continuing with the cube graph from Example 5.2.6, note that the partition given there does not possess a TUO. The solid lines in 5.7 represent one attempt at ordering the cells of the partition by placing cells 1 and 2 preceding 3 and 4. This fails to be a TUO because, for example, each vertex in cell 1 points to a different vertex of cell 3.

In contrast, let $\Pi = \{\{1,3\},\{2,4\},\{5,7\},\{6,8\}\}$. Now we claim that ordering cells 1 and 3 before cells 2 and 4. This is a TUO because, for example, each vertex in cell 1 points to the same two vertices in cell 2 and neither vertex of cell 4.

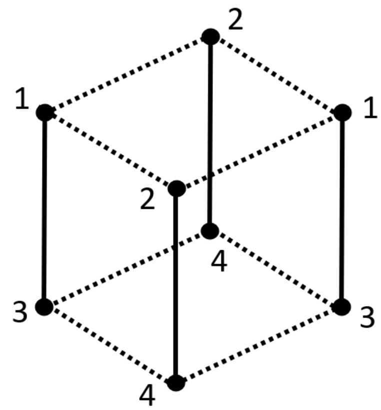


Figure 5.10: Q_3 with vertices labeled according to a partition with a TUO.

Also, since, for example, each vertex in 1 points to two vertices in 2, the quotient will have doubled edges.

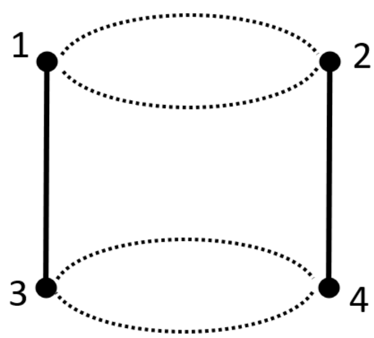


Figure 5.11: The quotient graph from the partition in figure 5.10.

Lemma 5.3.3. *Let G be a graph with an externally equitable partition Π having a TUO with partition Γ. Assume further that the vertices of G are ordered compatibly with Π. Then, for $j \preceq i$, the blocks C_{ij} defined in 5.2.16 satisfy $C_{ij} = \mathbb{1}\eta^T$ where η is a size p_j vector with l entry equal to ω_l^{ij}.*

Proof. Definition 5.3.1 ensures that whenever $j \preceq i$, we have, for each $v \in \Pi_j$ that

$N^s(v) \cap \Pi_i$ is either all of Π_i or empty. Therefore, each column of block C_{ij} is constant.

For a given $v \in \Pi_i$, the ordering of $s^{-1}(v) \cap t^{-1}(\Pi_j)$ assumed in Definition 5.2.2 implies that the lth nonzero entry of the v row of C_{ij} is ω_l^{ij}. $\square$

Note that Γ is also a partition of the vertices of $\tilde{G}$. While Γ is not in general externally equitable, Lemma 5.2.18 ensures that each Γ_i is an externally equitable partition of $G \mid_{\bigcup \Gamma_i}$.

Example 5.3.4. In Figure 5.12, we see a somewhat larger example of a graph with EEP Π possessing TUO Γ. Note that Γ is not an EEP of G since for example some vertices in Γ_2 have a single neighbor in Γ_3 and others have none.

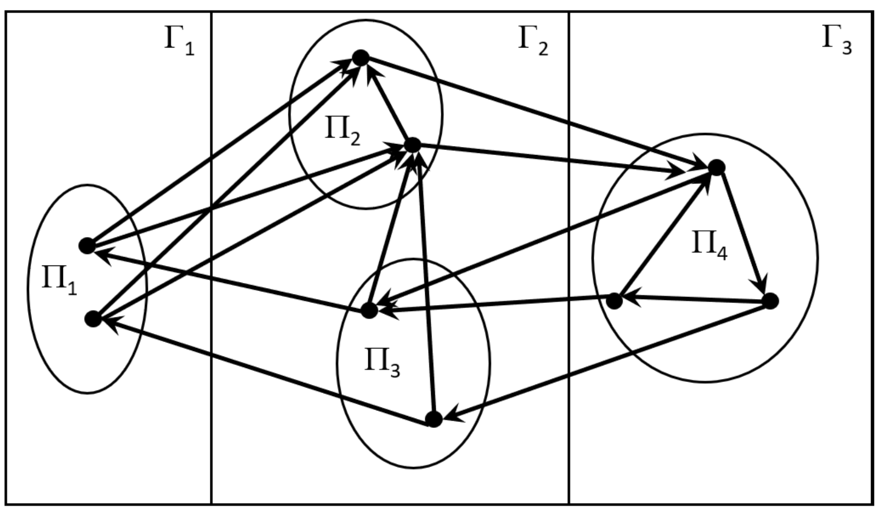


Figure 5.12: A graph with an EEP Π and a TUO with induced partition Γ.

Definition 5.3.5. *Given graph G with EEP Π possessing a target uniform ordering Γ, we define G_i to be the subgraph of G induced by $\bigcup \Gamma_i$. That is $G_i = G \mid_{\bigcup \Gamma_i}$. Now, $\tilde{G}_i$ is defined to be the quotient of G_i by Γ_i.*

Now, Theorem 5.2.9 can be applied to G_i and $\tilde{G}_i$. Note that Γ_i and its associated edge partition are just restrictions of Π and Ψ. It follows that

$$F_{\tilde{G}_i}(\tau, \omega) \mid F_{G_i}(\Pi\tau, \Psi\omega)$$

and as a result, the polynomial $h_{G_i,\Gamma_i}(\tau, \omega)$ can be defined. With this observation, we can express our main theorem for this chapter.

Theorem 5.3.6. *Let Π be an EEP of graph G into k parts with Γ a TUO of Π into r parts. For $i \in [r]$, let G_i be as in Definition 5.3.5. Then,*

$$h_{G,\Pi}(\tau, \omega) = \prod_{i=1}^{r} h_{G_i,\Gamma_i}(\tau + d_i, \omega)$$

In this formula, d_i is defined as in Lemma 5.2.17, in terms of the graph G and the partition Γ.

Proof. Let L and $\tilde{L}$ be defined as in Theorem 5.2.9 so that $\det L = h_{G,\Pi}(\tau, \omega) \cdot \det \tilde{L}$. We will prove the theorem by conjugating L into an upper block triangular form, taking advantage of the special structure ensured by lemma 5.3.3.

First, we will set a useful ordering for V_G. Since applying a permutation to the columns and rows of L will not change the value of its determinant, we are free to choose any ordering. So, let us assume that our vertex ordering is compatible with Π, in the sense of Lemma 5.2.16, and satisfies the further condition that $v \leq w$ whenever $v \in \Pi_i$,$w \in \Pi_j$ and $i \preceq j$ under the partial ordering defined in Definition 5.3.1. This amounts to ensuring that the ordering of the blocks of Π respect the partial order $\preceq$.

Now, the matrices Π, X, and Y all have the block diagonal structure given in Lemma 5.2.14. Thus, with W defined as in Theorem 5.2.9 and applying the results of that theorem, the matrix product

$$W^{-1}LW$$

has a 2×2 block form

$$\begin{bmatrix} \tilde{L} & D_{\Pi}^{-1}\Pi^T LX \\ 0 & Y^T LX \end{bmatrix}.$$

It follows that $h_{G,\Pi} = \det Y^T LX$. Putting this matrix product into block form, we have

$$Y^T LX \quad = \quad \left[\hat{Z}_i^T C_{ij} \hat{U}_j\right]_{1 \le i,j \le k}$$

Now, by Lemma 5.3.3, whenever $j \le i$, we have

$$\hat{Z}_i^T C_{ij} \hat{U}_j = \hat{Z}_i^T \mathbb{1} \eta^T \hat{U}_j = \left(\hat{Z}_i^T \mathbb{1}\right) \left(\eta^T \hat{U}_j\right) = 0.$$

The last equality follows from the fact that $\hat{Z}_i^T \mathbb{1} = 0$ which is a consequence of Lemma 5.2.16 and Proposition 5.2.13 via the identity $Y^T \Pi = \left[\delta_{ij} \hat{Z}_i^T \mathbb{1}\right]$.

Therefore, $Y^T LX$ is $r \times r$ upper block triangular so that

$$\det Y^T LX = \prod_{i=1}^{r} R_i$$

with R_i being the ii block of $Y^T LX$. Now, if we let K_i denote the set of indices $j \in [k]$ so that $\Pi_j \in \Gamma_i$, we can define the matrices

$$\hat{X}_i = \left[\delta_{jl}\hat{U}_j\right]_{j,l\in K_i} \text{ and } \hat{Y}_i = \left[\delta_{jl}\hat{Z}_i\right]_{j,l\in K_i}.$$

By Proposition 5.2.19, if we let $L_i = L_{G_i}(\tau + d_i, \omega)$, we have $R_i = \hat{Y}_i^T L_i \hat{X}_i$ so that

$$\det R_i = h_{G_i,\Theta_i}(\tau + d_i, \omega).$$

□

We can of course also straightforwardly use this theorem to get a characterization of $F_G(\Pi\tau, \Psi\omega)$.

Corollary 5.3.7. *With the same assumptions as in Theorem 5.3.6 and with $\tilde{G}$ defined as in Definition 5.2.3,*

$$F_G(\Pi\tau, \Psi\omega) = F_{\tilde{G}}(\tau, \omega)\prod_{i=1}^{r} h_{G_i,\Gamma_i}(\tau + d_i, \omega).$$

As we shall see below, this is particularly useful when the restricted quotient graphs G_i have a simple structure. In addition, we will often encounter examples in which the TUO is actually a total ordering on $[k]$. In this case, $r = k$ and $\Gamma_i = \Pi_i$ so that our theorem simplifies.

Corollary 5.3.8. *With the same assumptions as in Theorem 5.3.6, if $r = k$ then each d_i is constant. If we let $d_i = \tilde{d}_i\mathbb{1}$, then*

$$h_{G,\Pi}(\tau, \omega) = \prod_{i=1}^{k} \frac{F_{G_i,\Pi_i}((\tau_i + \tilde{d}_i)\mathbb{1}, \omega)}{\tau_i + \tilde{d}_i}.$$

In this case, we can actually factor $h_{G,\Pi}$ into a product of polynomials that are each univariate in their τ variables. It is also significant that the coefficients of each right hand factor are easily expressible in terms of the converging forests of the subgraphs G_i. From a computational standpoint, Corollary 5.3.8 implies that computing h directly might be significantly easier than computing F in these circumstances.

The main theorem of this section was inspired by the result of [54]. This paper includes a coordinate free proof of a similar fact related to adjacency matrices and equitable partitions.

5.4 Applications

In this final section we will apply Theorem 5.3.6 to a few concrete cases and then note a number of other potential applications. Let us first consider the complete graph $G = K_n$. This graph has the special property that any partition $\Pi = \{\Pi_1, \ldots, \Pi_k\}$ is an EEP with a total TUO. The resulting quotient will be a complete multigraph with $c_j = |\Pi_j|$ edges from Π_i to Π_j. In fact, since each cell Π_i induces a complete subgraph of G, we can apply 5.1.8 the characterize h. This requires us to be a little careful with our definition of Tx however. We will need to specialize Ψ so that all Π_i internal edges e have weights given by $x_{t_G(e)}$ while edges from Π_i to Π_j have weights given by $x_{t_{\tilde{G}}(e)}$ where $\tilde{G}$ is the quotient of G by Π. This is actually a specialization of the edge partition Ψ induced by Π and so we can apply our theorem.

Proposition 5.4.1. *With $\tilde{T}x$ defined as in the above paragraph*

$$h_{K_n,\Pi}(\tau, \tilde{T}x) = \prod_{i=1}^{k} (\tau_i + |x|)^{c_i - 1}.$$

With this proposition, we are able to better understand the factorization in Proposition 5.1.5. In particular, each factor of h is a univariate, weighted forest polynomial of a complete graph, evaluated at an appropriate $\tau_i + a(x)$ polynomial.

Similar considerations lead us to the h polynomial for the complete multipartite graphs.

Proposition 5.4.2. *Let $G = K_{c_1,\ldots,c_k}$, then define Π to be the partition defined by the maximal independent sets of G. Then, if we let $x^{(i)} = \sum_{j \notin \Pi_i} x_j$, we have*

$$h_{G,\Pi}(\tau, \tilde{T}x) = \prod_{i=1}^{k} (\tau_i + x^{(i)})^{c_i - 1}.$$

This result generalizes the polynomial defined in [24]. Note that the conclusions of this paper apply to the more general class of *inversion* graphs that includes the complete multipartite graphs. Further, by specializing some of the x_i to 0, we recover the forest generating functions for the *almost* multipartite graphs discussed in [12].

Further specializing the above formula, we take $x = \mathbb{1}$. Now, we can express the forest polynomial of the complete multipartite graph as

$$F_G(\Pi\tau, \mathbb{1}) = \left(\prod_{i=1}^{k} (\tau_i + n - c_i)^{c_i - 1} \right) \left(\prod_{i=1}^{k} (\tau_i + n) - \sum_{i=1}^{k} c_i \prod_{\substack{j=1 \\ j \neq i}}^{k} (\tau_j + n) \right)$$

With some considerable effort, we can expand this product and recover an expression for the coefficients of each τ monomial in this generating function. Note that in the below theorem, we have let $c = (c_1, \ldots, c_k)$. In addition, given vectors a, b of the same size l, we let $\binom{a}{b} = \prod_{i=1}^{l} \binom{a_i}{b_i}$

Theorem 5.4.3. *If $G = K_{c_1c_2\ldots c_k}$, then the number of spanning forests of G with exactly η_i roots in c_i is given by*

$$\sum_{1\le\mu\le c} \binom{c-1}{\mu-1}\binom{\mu}{\eta}(-c)^{c-\mu}n^{|\mu-\eta|}\left(\sum_{i=1}^{k}\frac{c_i\eta_i}{n\mu_i}\right).$$

Proof. We will defer the proof of this claim to Appendix A as it is long and the derivation itself does not really add to the exposition. □

The cost of evaluating this expression depends both on c and η. For each $i = 1,\ldots,k$, our sum runs independently through the values $\max(1,\eta_i) \le \mu_i \le c_i$. This is because the product $\binom{\mu}{\eta} = 0$ whenever $\eta_i > \mu_i$. This means that the number of terms in the sum over μ may be exponential in n.

If each $c_i \approx \frac{n}{k}$ and each $\eta_i \le 1$, then this makes $(\frac{n}{k}-1)^k$ terms in the sum to compute. Interestingly, this is not exponential in n unless k is proportional to n. So, this formula will be practical to evaluate if the number of cells in the partition is fixed and the size of some cells grows with n. On the other hand, it may be applicable to derive asymptotic results such as [49].

Another nice application of Theorem 5.3.6 is to the line graphs defined in 1.1. [40, 19] These graphs are the subject of some early interest if spanning tree polynomials [39]. If G is a graph with n vertices, then $DL(G)$ has a natural vertex partition Π into n parts which places edge $i \to j$ into cell Π_j. For general i, j, either there are no edges in $DL(G)$ from Π_i to Π_j or some member of Π_j is pointed to by every member of Π_i. Thus, the partition is an EEP and further has a TUO that is a total order.

Proposition 5.4.4. *With Π defined as in the previous paragraph,*

$$h_{DL(G),\Pi}(\tau,\omega) = \prod_{i=1}^{n} (\tau_i + d_s(i))^{|N^t(i)|}$$

Proof. We have already observed that we can apply Theorem 5.3.6. The Quotient graph of $LD(G)$ by Π is clearly G. On the other hand, the cells of Π are independent sets. Finally, we can see that $C_{ij} = \delta_{i\to j\in E(G)}\omega^{(ij)}\mathbb{1}\mathbb{1}_i$ since every edge $a \to i$ in Π_i is adjacent only to $i \to j \in \Pi_j$ [5]. Thus $C_{ij}\mathbb{1} = \delta_{i\to j\in E(G)}\omega^{(ij)}\mathbb{1}$, and so $d_i = \sum_{j\neq i}\delta_{i\to j\in E(G)}\omega^{(ij)}\mathbb{1}$. □

In fact, since *any* ordering of the cells Π_i satisfies the conditions of a TUO, the matrix product $Y^T L_{DL(G)} X$ is not just upper triangular but actually diagonal. This allows us to express $Q_{DL(G)}(t)$ directly in terms of X, Y, and $Q_G(t)$. Unfortunately, in the interest of space we will have to leave this result for future work.

Another interesting example comes from studying graphs with an involutional automorphism ϕ. That is, a graph automorphism of order 2. In this case, we can partition the vertex set into cells corresponding to the orbits of ϕ and then form a *TUO* by placing the fixed point cells behind the order 2 cells. This generalizes a result proved in [56] for spanning trees of undirected graphs.

This example shows that there may be other special classes of automorphisms that give rise to similar factorizations.

The threshold graphs discussed in Section 3.4 also admit a partition with a TUO. One construction breaks the vertices up into cells based on contiguous blocks of the a or b vectors with identical entries. The application of this observation to generating a symbolic characteristic adjacency matrix polynomial is discussed in the undirected

[5] We have dropped the l subscript from $\omega_1^{(ij)}$ as it is redundant.

case in [33]. For directed graphs, this case is interesting because the Π internal edges of such a partition always form transitive tournament graphs while the quotient graph is always another directed threshold graph.

We conclude with a final comment about Theorem 5.3.6. One way to interpret the result is that it tells us that in graphs satisfying it's hypothesis, we can say exactly how the addition and removal of Π internal edges will affect the partitioned forest generating function. Indeed, the theorem implies that these edges only appear in the factor of h corresponding to the graph restricted to the appropriate cell of Γ. One consequence of this observation is that all of the formulas in this section can be easily perturbed by the addition of such edges.

References

[1] Martin Aigner. *A Course in Enumeration*, volume 1. Springer, 2007.

[2] Christos G Bampis, Petros Maragos, and Alan C Bovik. Graph-driven diffusion and random walk schemes for image segmentation. *IEEE Transactions on Image Processing*, 26(1):35–50, 2016.

[3] Jørgen Bang-Jensen and Gregory Gutin. *Classes of directed graphs*, volume 11. Springer, 2018.

[4] Türker Biyikoglu, Josef Leydold, and Peter F Stadler. *Laplacian eigenvectors of graphs: Perron-Frobenius and Faber-Krahn type theorems*. Springer, 2007.

[5] Collin Patrick Bleak, Peter J Cameron, and Feyisayo Olukoya. Automorphisms of shift spaces and the higman-thompson groups: the one-sided case. *Discrete Analysis*, 2021.

[6] Béla Bollobás. *Modern graph theory*, volume 184. Springer Science & Business Media, 1998.

[7] Domingos M Cardoso, Charles Delorme, and Paula Rama. Laplacian eigenvectors and eigenvalues and almost equitable partitions. *European journal of combinatorics*, 28(3):665–673, 2007.

[8] J. S. Caughman and J. J. P. Veerman. Kernels of directed graph Laplacians. *The Electronic Journal of Combinatorics*, 13, 2006.

[9] Seth Chaiken. A combinatorial proof of the all minors matrix tree theorem. *SIAM Journal on Algebraic and Discrete Methods*, September 1982.

[10] Pavel Chebotarev and Rafig Agaev. Forest matrices around the Laplacian matrix. *Linear Algebra and its Applications*, (356):253–274, 2002.

[11] Yiou Chen, Jianhao Hu, and Xiang Ling. De bruijn graph based 3d network on chip architecture design. In *2009 International Conference on Communications, Circuits and Systems*, pages 986–990. IEEE, 2009.

[12] Sujing Cheng, Wuxian Chen, and Weigen Yan. Counting spanning trees in almost complete multipartite graphs. *Journal of Algebraic Combinatorics*, 56(3):773–783, 2022.

[13] Sunita Chepuri, Christopher J Dowd, Andrew Hardt, Gregory Michel, Sylvester W Zhang, and Valerie Zhang. Arborescences of covering graphs. *Algebraic Combinatorics*, 5(2), 2022.

[14] Francesc Comellas, Miquel Angel Fiol, Joan Gimbert, and Margarida Mitjana. The spectra of wrapped butterfly digraphs. *Networks: An International Journal*, 42(1):15–19, 2003.

[15] Dragoš Cvetković and Slobodan K Simić. Towards a spectral theory of graphs based on the signless laplacian, i. *Publications de l'Institut Mathematique*, 85(105):19–33, 2009.

[16] João Domingos Gomes da Silva Junior, Carla Silva Oliveira, and Liliana Manuela Gaspar Cerveira da Costa. On the characteristic polynomial of the a α-matrix for some operations of graphs. *Computational and Applied Mathematics*, 42(5):206, 2023.

[17] Patrick De Leenheer. An elementary proof of a matrix tree theorem for directed graphs. *SIAM Review*, 62(3):716–726, 2020.

[18] Charles Delorme and J-P Tillich. The spectrum of de bruijn and kautz graphs. *European Journal of Combinatorics*, 19(3):307–319, 1998.

[19] Fengming Dong and Weigen Yan. Expression for the number of spanning trees of line graphs of arbitrary connected graphs. *Journal of Graph Theory*, 85(1):74–93, 2017.

[20] Keith Dufault-Thompson and Xiaofang Jiang. Applications of de bruijn graphs in microbiome research. *iMeta*, 1(1):e4, 2022.

[21] Ragnar Freij. Shifted de bruijn graphs. In *ICMCTA*, pages 195–202. Springer, 2014.

[22] Martin Fürer. Efficient computation of the characteristic polynomial of a threshold graph. *Theoretical Computer Science*, 657:3–10, 2017.

[23] Lucia Valentina Gambuzza and Mattia Frasca. A criterion for stability of cluster synchronization in networks with external equitable partitions. *Automatica*, 100:212–218, 2019.

[24] Yibo Gao and Siyu Liu. Tree enumeration polynomials on separable permutations. *Enumerative Combinatorics and Applications*, 4(1), 2024.

[25] Ira Gessel and Gérard Viennot. Binomial determinants, paths, and hook length formulae. *Advances in mathematics*, 58(3):300–321, 1985.

[26] C. D. Godsil. *Algebraic Combinatorics*. Chapman and Hall, New York, 1993.

[27] C. D. Godsil and G. Royle. *Algebraic Graph Theory*. Springer-Verlag, New York, 2001.

[28] J. C. Gower. A modified Leverrier-Faddeev algorithm for matrices with multiple eigenvalues. *Linear Algebra and its Applications*, 31:61–70, 1980.

[29] Peter Grindrod, Desmond J Higham, and Vanni Noferini. The deformed graph Laplacian and its applications to network centrality analysis. *SIAM Journal on Matrix Analysis and Applications*, 39(1):310–341, 2018.

[30] Erik R Hauge and Tor Helleseth. De bruijn sequences, irreducible codes and cyclotomy. *Discrete Mathematics*, 159(1-3):143–154, 1996.

[31] Roger Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2 edition, 2012.

[32] Shun-Pin Hsu. Minimal Laplacian controllability of directed threshold graphs. *IEEE Control Systems Letters*, 6:2413–2418, 2021.

[33] David Pokrass Jacobs, Vilmar Trevisan, and Fernando Tura. Computing the characteristic polynomial of threshold graphs. *J. Graph Algorithms Appl.*, 18(5):709–719, 2014.

[34] Brian Jacobson, Andrew Niedermaier, and Victor Reiner. Critical groups for complete multipartite graphs and cartesian products of complete graphs. *Journal of Graph Theory*, 44(3):231–250, 2003.

[35] Osamu Kada. Characteristic polynomials and zeta functions of equitably partitioned graphs. *Linear Algebra and its Applications*, 588:471–488, 2020.

[36] Alexander Kelmans, Igor Pak, and Alexander Postnikov. Tree and forest volumes of graphs. *available as DIMACS Technical Report*, 3, 2000.

[37] Oliver Knill. Counting rooted forests in a network. `https://arxiv.org/abs/1307.3810`.

[38] Oliver Knill. Cauchy-Binet for pseudo-determinants. *Linear Algebra and its Applications*, (459):522–547, 2014.

[39] Donald E Knuth. Oriented subtrees of an arc digraph. *Journal of Combinatorial Theory*, 3(4):309–314, 1967.

[40] Lionel Levine. Sandpile groups and spanning trees of directed line graphs. *Journal of Combinatorial Theory, Series A*, 118(2):350–364, 2011.

[41] Xiaogang Liu and Shunyi Liu. On the aα-characteristic polynomial of a graph. *Linear Algebra and its Applications*, 546:274–288, 2018.

[42] Matúš Mihalák, Przemysław Uznański, and Pencho Yordanov. Prime factorization of the Kirchhoff polynomial: Compact enumeration of arborescences. In *2016 Proceedings of the Thirteenth Workshop on Analytic Algorithmics and Combinatorics (ANALCO)*, pages 93–105. SIAM, 2016.

[43] Inom Mirzaev and David M. Bortz. Laplacian dynmaics with synthesis and degredation. *Bulletin of Mathematical Biology*, 77(6):1013–1045, 2015.

[44] Inomzhon Mirzaev and Jeremy Gunawardena. Laplacian dynamics on general graphs. *Bulletin of Mathematical Biology*, 75(11):2118–2149, 2013.

[45] J. W. Moon. *Counting Labeled Trees*. Canadian Mathematical Congress, 1970.

[46] Fabio Morbidi. The deformed consensus protocol. *Automatica*, 49(10):3049–3055, 2013.

[47] Jim Pitman and Wenping Tang. Tree formulas, mean first passage times, and Kemeny's constant of a Markov chain. *Bernoulli*, 24:1942–1972, 2018.

[48] Yi Qi, Yuhao Yi, and Zhongzhi Zhang. Topological and spectral properties of small-world hierarchical graphs. *The Computer Journal*, 62(5):769–784, 2019.

[49] Dudley Stark. The asymptotic number of spanning forests of complete bipartite labelled graphs. *Discrete Mathematics*, 313(11):1256–1261, 2013.

[50] Andrei Tarfulea and Robert Perlis. An Ihara formula for partially directed graphs. *Linear Algebra and its Applications*, 431(1-2):73–85, 2009.

[51] AJ Van der Schaft, Shodhan Rao, and Bayu Jayawardhana. A network dynamics approach to chemical reaction networks. *International Journal of Control*, 89(4):731–745, 2016.

[52] JJP Veerman, Tessa Whalen-Wagner, and Ewan Kummel. Chemical reaction networks in a Laplacian framework. *Chaos, Solitons & Fractals*, 166:112859, 2023.

[53] Wen-Huan Wang, Lei Zhong, and Lian-Jiang Zheng. The signless Laplacian coefficients and the incidence energy of the graphs without even cycles. *Linear Algebra and its Applications*, 563:476 – 493, 2019.

[54] Yaokun Wu. Invariant subspace, determinant and characteristic polynomials. *Linear algebra and its applications*, 428(7):1753–1760, 2008.

[55] Pencho Yordanov and Jörg Stelling. Efficient manipulation and generation of Kirchhoff polynomials for the analysis of non-equilibrium biochemical reaction networks. *Journal of the Royal Society Interface*, 17(165):20190828, 2020.

[56] Fuji Zhang and Weigen Yan. Enumerating spanning trees of graphs with an involution. *Journal of Combinatorial Theory, Series A*, 116(3):650–662, 2009.

A Derivation of Theorem 5.4.3

As a reminder, we assume that $G = K_{c_1 c_2 \cdots k}$ is the complete multipartite graph with independent sets of size $c_1, c_2, \ldots,$ and c_k. We will also make use of the following identity.

$$\begin{aligned}
\sum_{a \leq \mu \leq b} \left(\prod_{i=1}^{k} \alpha_{i\mu_i} (\tau_i + n)^{\mu_i} \right) &= \sum_{a \leq \mu \leq b} \alpha(\mu)(\tau + n1)^{\mu} \\
&= \sum_{a \leq \mu \leq b} \alpha(\mu) \left(\prod_{i=1}^{k} \sum_{l=0}^{\mu_i} \binom{\mu_i}{l} n^{\mu_i - l} \tau_i^l \right) \\
&= \sum_{a \leq \mu \leq b} \alpha(\mu) \sum_{0 \leq \eta \leq \mu} \binom{\mu}{\eta} n^{|\mu - \eta|_1} \tau^{\eta} \\
&= \sum_{0 \leq \eta \leq b} \left(\sum_{a \leq \mu \leq b} \alpha(\mu) \binom{\mu}{\eta} n^{|\mu - \eta|_1} \right) \tau^{\eta}
\end{aligned}$$

$$
\begin{aligned}
F_G(\Pi\tau,\mathbb{1}) &= \left(\prod_{i=1}^{k}(\tau_i+n-c_i)^{c_i-1}\right)\left(\prod_{i=1}^{k}(\tau_i+n)-\sum_{i=1}^{k}c_i\prod_{\substack{j=1\\ j\neq i}}^{k}(\tau_j+n)\right)\\
&= \left(1-\sum_{i=1}^{k}\frac{c_i}{\tau_i+n}\right)\left(\prod_{i=1}^{k}\left((\tau_i+n-c_i)^{c_i}+c_i(\tau_i+n-c_i)^{c_i-1}\right)\right)\\
&= \left(1-\sum_{i=1}^{k}\frac{c_i}{\tau_i+n}\right)\cdot\\
&\quad \prod_{i=1}^{k}\left(\sum_{j=0}^{c_i}\binom{c_i}{j}(-c_i)^{c_i-j}(\tau_i+n)^j+c_i\sum_{l=0}^{c_i-1}\binom{c_i-1}{l}(-c_i)^{c_i-1-l}(\tau_i+n)^l\right)\\
&= \left(1-\sum_{i=1}^{k}\frac{c_i}{\tau_i+n}\right)\left(\prod_{i=1}^{k}\left(\sum_{j=1}^{c_i}\binom{c_i-1}{j-1}(-c_i)^{c_i-j}(\tau_i+n)^j\right)\right)\\
&= \left(1-\sum_{i=1}^{k}\frac{c_i}{\tau_i+n}\right)\left(\sum_{1\le\mu\le c}\binom{c-1}{\mu-1}(-c)^{c-\mu}(\tau+n1)^{\mu}\right)\\
&= \sum_{1\le\mu\le c}\binom{c-1}{\mu-1}(-c)^{c-\mu}(\tau+n1)^{\mu}-\\
&\qquad\qquad \sum_{i=1}^{k}\sum_{1-e_i\le\nu\le c-e_i}\binom{c-1}{\nu+e_i-1}(-c)^{c-\nu}(\tau+n1)^{\nu}\\
&= \sum_{0\le\eta\le c}\left(\sum_{1\le\mu\le c}\binom{c-1}{\mu-1}\binom{\mu}{\eta}(-c)^{c-\mu}n^{|\mu-\eta|}\right)\tau^{\eta}-\\
&\qquad\qquad \sum_{i=1}^{k}\sum_{0\le\zeta\le c-e_i}\left(\sum_{1-e_i\le\nu\le c-e_i}\binom{c-1}{\nu+e_i-1}\binom{\nu}{\zeta}(-c)^{c-\nu}n^{|\nu-\zeta|}\right)\tau^{\zeta}\\
&= \sum_{0\le\eta\le c}\left(\sum_{1\le\mu\le c}\binom{c-1}{\mu-1}\binom{\mu}{\eta}(-c)^{c-\mu}n^{|\mu-\eta|}\right)\tau^{\eta}-\\
&\qquad\qquad \sum_{0\le\zeta<c}\sum_{i:\zeta_i<c_i}\left(\sum_{1\le\nu\le c}\binom{c-1}{\nu-1}\binom{\nu-e_i}{\zeta}(-c)^{c-\nu+e_i}n^{|\nu-e_i-\zeta|}\right)\tau^{\zeta}\\
&= \sum_{0<\eta\le c}\left(\sum_{1\le\mu\le c}\binom{c-1}{\mu-1}\binom{\mu}{\eta}(-c)^{c-\mu}n^{|\mu-\eta|}\left(1-\sum_{i:\eta_i<\mu_i}\frac{c_i(\mu_i-\eta_i)}{n\mu_i}\right)\right)\tau^{\eta}\\
&= \sum_{0<\eta\le c}\left(\sum_{1\le\mu\le c}\binom{c-1}{\mu-1}\binom{\mu}{\eta}(-c)^{c-\mu}n^{|\mu-\eta|}\left(\sum_{i=1}^{k}\frac{c_i\eta_i}{n\mu_i}\right)\right)\tau^{\eta}
\end{aligned}
$$