# Matrix-Tree Theorem for Directed Graphs

Jonathan Margoliash

August 31, 2010

**Abstract**

In this paper we provide a tool for counting tree analogues in directed graphs, the theorem proved here being a generalization of Gustav Kirchhoff's Matrix-Tree Theorem. This paper does not presuppose the reader's knowledge of any graph theory, only requiring a modicum of linear algebra. We begin by building from scratch the graph theory necessary to understand the statement of the Matrix-Tree Theorem for Directed Graphs. We then state and prove our generalized result, an endeavor which relates the presence of cycles in functional digraphs and permutation groups.

## 1 Introductory Graph Theory

The following section is meant to accomplish two tasks. Firstly, it should allow a reader unacquainted with graph theory to understand the theorem presented here and its proof. Secondly, it provides a standardized set of notations and definitions to avoid the confusion of a reader already so acquainted.

**Definition 1.1** A *directed-graph*, hereafter referred to as a *digraph*, is a pair $(V, E)$, where $V$ is a nonempty set of nodes or vertices, and $E$ is a set of directed edges between the vertices. To complete this definition, we define a *directed edge* to be an object which has two properties associated with it: a starting node, and an ending node.

**Notation 1.2** Let $G = (V, E)$ be a digraph, and let $i, j$ be vertices in $V$. Then the number of edges in $E$ that start at $i$ and end at $j$ we write as $a_{ij}$. This is normally read as *the number of edges from $i$ to $j$.*

**Definition 1.3** An *undirected graph* $G$ is a digraph where $a_{ii} = 0, a_{ij} = a_{ji}$ and $a_{ij} = 0$ or $1$ for all $i, j \in V$.

Some notes about the distinctions between digraphs and undirected graphs. First, the term digraph is used because in a digraph the edges are directed, $a_{ij}$ need not equal $a_{ji}$. Secondly, digraphs allow for *self-loops*, a self loop being an edge from a vertex to itself, which occurs when $a_{ii} \neq 0$. Lastly, our definition of digraphs allows for the case when $a_{ij} > 1$, when there are multiple edges from $i$ to $j$. These three characteristics distinguish between digraphs and the undirected graphs discussed in other literature.

Because of the topic of this paper, all graphs hereafter are assumed to be finite (in that the number of vertices and edges is finite). However, for those interested in a more general view of graph theory, the preliminary definitions are equally valid for infinite graphs as well finite graphs.

**Definition 1.4** Let $G = (V, E)$ be a digraph. If, for some vertices $i, j \in V$, there exists an edge starting at $i$ and ending at $j$, then we say $j$ is an *outneighbor* of $i$.

**Definition 1.5** Let $G = (V, E)$ be a digraph, and $v$ be a vertex in $V$. We call an edge starting at $v$ an *outedge* of $v$.

**Definition 1.6** Let $G$ be a digraph, and $v$ be a vertex in $V$. Then we say the *outdegree* of $v$ is $\sum_{i \in V} a_{vi}$, the number of outedges of $v$. We write this number as $\deg^+(i)$.

The following definitions are used to characterize vertices by the edges between them.

**Definition 1.7** Let $G$ be a digraph and let $x, y \in V$. A *walk* from $x$ to $y$ is a sequence[1] of vertices $\{x = x_1, x_2, \ldots, x_{k-1}, x_k = y\}$ where $x_{i+1}$ is an outneighbor of $x_i$ for $1 \leq i < k$. An *infinite walk* starting at $x$ is a infinite sequence of vertices $\{x = x_1, x_2, x_3, \ldots\}$ where $x_{i+1}$ is an outneighbor of $x_i$ for all $i \in \mathbb{N}$.

**Definition 1.8** Let $G$ be a digraph and let $x, y \in V$. A *path* from $x$ to $y$ is a sequence of alternating vertices and edges $\{x = x_1, e_1, x_2, \ldots, x_{k-1}, e_{k-1}, x_k = y\}$ where $e_i$ is an edge from $x_i$ to $x_{i+1}$ for all $1 \leq i < k$ and where all the vertices and edges are unique.

Since each vertex in a path must be unique and all graphs have finitely many vertices, all paths are finite by nature.

**Definition 1.9** Let $G$ be a digraph, and let $x, y \in V$. Then $x$ is said to be *connected to* $y$ if there exists a walk from $x$ to $y$.

**Note 1.10** From the definitions above it immediately follows that, for vertices $x, y \in V$, there is a path from $x$ to $y$ iff there is a walk from $x$ to $y$. We choose to draw a distinction between paths for the following reason. We will consider walks when we want to discuss whether or not two vertices are connected. We will consider paths when we want to discuss whether or not that connection is unique.

As defined above, the property of being connected applies only to pairs of vertices, and hence is a local property. It can be generalized to the following global property.

**Definition 1.11** Let $G = (V, E)$ be an undirected graph. We say $G$ is *connected* if every vertex is connected to every other vertex. An equivalent definition is that for all nonempty subsets $A, B \subset V$ where $A \cap B = \emptyset$ and $A \cup B = V$, there exists an edge between a node in $A$ and a node in $B$. An undirected graph is called *disconnected* if it is not connected.

The first of the two equivalent definitions above embodies the idea of being able to walk from one node to any other node, while the second embodies the notion that $V$ cannot be separated into two unrelated subsections $A$ and $B$. These two notions are equivalent only because of the bidirectionality of edges in undirected graphs, and are not equivalent for digraphs.

**Definition 1.12** Let $G = (V, E)$ be a digraph. We say $G$ is *strongly connected* if every vertex is connected to every other vertex. We say $G$ is *weakly connected* if for all nonempty subsets $A, B \subset V$ where $A \cap B = \emptyset$ and $A \cup B = V$, there exists an edge from a node in one subset to a node in the other. A digraph is called *disconnected* if it is not weakly connected.

**Note 1.13** As the names suggest, strong connectivity is a sufficient but not necessary condition for weak connectivity.

In this paper, we are more interested in the distinction between being weakly connected and being disconnected than in the notion of being strongly connected. However, we are more interested in working with walks and paths than sets of vertices. As such, we proceed to give an equivalent definition of weak connectivity in terms of walks.

[1] In this paper we will use the term sequence to denote a finite, nonempty, ordered set.

**Definition 1.14** Let $G$ be a digraph, and let $x, y \in V$. A *direction-ignoring walk* from $x$ to $y$ is a sequence of vertices $\{x = x_1, x_2, \ldots, x_{k-1}, x_k = y\}$ where for all $1 \leq i < k$, either $x_i$ is an outneighbor of $x_{i+1}$ or $x_{i+1}$ is an outneighbor of $x_i$.

**Definition 1.15** Let $G$ be a digraph. We say $G$ is *weakly connected* if there is a direction-ignoring walk from every vertex to every other vertex.

The two definitions of weak connectivity are equivalent for the same reason that there is no distinction between strong connectivity and weak connectivity in undirected graphs.

Lastly we wish to formalize the idea of starting with a digraph, and then taking edges out of it to form a new digraph.

**Definition 1.16** Let $G = (V, E)$ be a digraph. We say $H = (V, F)$ is a subgraph of G if $F \subset E$. Note that $G$ and $H$ share the same vertex set.

## 2 History and Generalization

We have now covered enough basic graph theory to discuss the object that this paper is interested in.

**Definition 2.1** Let $G$ be an undirected graph. We say $G$ is a *tree* if there is a unique path from every vertex to every other vertex.

Trees are useful objects in many areas of graph theory. For example, trees are the only undirected graphs that have the interesting property that they are connected, but removing any edge from them will disconnect them. Thus, for an undirected graph $G$, the number of its subgraphs that are trees is the number of different ways one can pare down $G$ to a minimally connected graph. The goal of this paper, however, is not to explore the interesting uses of trees, but to provide for those uses a tool for counting trees. In 1847, Kirchhoff found a way to count how many subgraphs of a connected undirected graph are trees[3], and a more modern proof can be found in Chaiken's article[2]. We wish to find a similar method for counting trees in digraphs.

Generalizing this problem to digraphs requires some care. In digraphs, connectedness is a directed property; it is not symmetric. The above definition of a tree does not respect that aspect of digraphs. As such, we must find an analogous object to discuss in the setting of directed graphs.

**Definition 2.2** Let $G$ be a digraph. We say $s \in V$ is a *sink* if, whenever $v \in V$ is an outneighbor of $s$, then $v = s$. An equivalent definition is $s \in V$ is a sink iff $\deg^+(s) - a_{ss} = 0$.

**Definition 2.3** For our purposes, a digraph $G$ will be called a *sink-rooted digraph* if there is a sink $s \in V$ to which all other nodes are connected. Moreover, for the sake of convenience, we require the sink to have one and only one self-loop, that $a_{ss} = 1$. This sink will be called the *root*.

**Definition 2.4** Let $G$ be a sink-rooted digraph. We say $G$ is a *reverse arborescence* if, for all nodes $v \in V$ that are not the root, there is a unique path from $v$ to the root.

It turns out that reverse arborescences are the closest analogues to trees in directed graphs, engendering a result that is nearly identical Kirchhoff's original theorem. Our goal is then, given a sink-rooted digraph $G$, count the number of reverse arborescence subgraphs it has. First a couple of minor observations to ease our notation.

**Observation 2.5** *Let $G = (V, E)$ be a sink-rooted digraph, with root $r$. Then $r$ is the unique sink in $V$.*

**Proof:** Suppose $s \in V$ is a sink. Then $s$ has no outneighbors other than itself, and hence is connected only to itself. But $s$ is connected to $r$ since $r$ is the root of the graph. Thus $s = r$. $\square$

**Notation 2.6** Let $G$ be a sink-rooted digraph with root $s \in V$. Since the root is the unique sink in $G$, we will use the two terms interchangeably. Moreover, the uniqueness of the sink allows us to differentiate between it and every other node, which we will call *sites.* We write $V_0 = V \setminus \{s\}$, the set of sites.

We are now ready to proceed with the result of this paper. To do so we need the following constructs, which may seem unmotivated now, but should be made clearer in the proof.

**Definition 2.7** Let $G$ be a sink-rooted digraph, and let the vertices in $V$ be indexed from 1 to $n$, where the nth node is the sink. The *adjacency matrix* of $G$ is the $n \times n$ matrix $A = [a_{ij}]$, where $a_{ij}$ is as defined in Notation 1.2 . The *diagonal matrix* of $G$ is the $n \times n$ diagonal matrix $D = [d_{ij}]$, where

$$d_{ij} = \begin{cases} \deg^+(i), & \text{if } i = j \\ 0, & \text{otherwise} \end{cases}$$

The *Laplacian* of $G$ is the matrix $D - A$. The *sink-reduced Laplacian* of $G$ is the $(n-1) \times (n-1)$ matrix $L$ formed by removing the nth row and column from the Laplacian of $G$, the row and column corresponding to the sink. To clarify, the sink-reduced Laplacian of $G$ is the matrix $L = [l_{ij}]$, where, for all $1 \le i, j \le n-1$,

$$l_{ij} = \begin{cases} \deg^+(i) - a_{ii}, & \text{if } i = j \\ -a_{ij}, & \text{otherwise} \end{cases}$$

This allows us to state the result of this paper:

**Theorem 2.8** (Matrix-Tree Theorem for Digraphs) *Let $G = (V, E)$ be a sink-rooted digraph. Then the number of reverse arborescence subgraphs of $G$ is equal to the determinant of the sink-reduced Laplacian of $G$.*

# 3 Proof of Theorem

Before we begin directly tackling the proof of the theorem, we must broaden our understanding of reverse arborescences as a specific type of a functional digraph.

**Definition 3.1** A functional digraph $F$ is a digraph where $\deg^+(x) = 1$ for all nodes $x \in V$. An equivalent definition: $F = (V, E)$ is a functional digraph if there exists a function $f : V \to V$ such that

$$a_{ij} = \begin{cases} 1 & \text{if } f(i) = j, \\ 0 & \textit{otherwise} \end{cases}$$

**Notation 3.2** Let $G$ be a sink-rooted digraph. Then we denote the set of its functional subgraphs by $\mathbf{X}$.

We introduce functional digraphs because they are much easier to count than reverse arborescences. For example, while it is difficult to count the number of reverse arborescence subgraphs of a digraph $G$, it is easy to count the number of functional subgraphs. Any given functional subgraph can be specified by choosing exactly one outedge of each vertex in $G$. Since these choices can be made independently from one another -

as the only requirement to be a functional digraph is about the number of outedges of each vertex, not their directions - we get that

$$|\mathbf{X}| = \prod_{v \in V} \deg^+(v) = \deg^+(s) * \left( \prod_{v \in V_0} \deg^+(v) \right) = \prod_{v \in V_0} \deg^+(v)$$

But counting functional subgraphs is only useful in conjunction with the following fact:

**Fact 3.3** *A reverse arborescence is a functional digraph.*

**Proof:** Let $G = (V, E)$ be a reverse arborescence. To show it is a functional digraph, we must show that there is exactly one edge leaving from every node in $V$. First consider the sink. Since there is exactly one self-loop at the sink, there is at least one edge leaving the sink. Since the sink is connected to no nodes besides itself, there are no other edges leaving the sink. Hence the sink has only one outedge.

Now consider a node $x \in V_0$. Suppose $x$ has more than one edge leaving from it. Then $x$ has two different edges leaving it, $e_1$ and $e_2$. Let $y_1, y_2$ be the two vertices these edges terminate at. (Note that $y_1$ is not necessarily a different vertex then $y_2$). Since $G$ is a reverse arborescence, there exist paths $P_1$ and $P_2$ from $y_1$ and $y_2$, respectively, to the sink. Then $x \to y_1$ via edge $e_1$, and then to the sink via $P_1$, is a different path to the sink than the path $x \to y_2$ via edge $e_2$, and then to the sink via $P_2$. This contradicts the fact that, since $G$ is a reverse arborescence, there must be a unique path from $x$ to the sink. Thus $x$ cannot have more than one outedge. But it must have at least one outedge, because it is connected to the sink. Hence it has exactly one outedge.

Therefore all nodes in $V$ have exactly one outedge, and $G$ is a functional digraph. $\square$

Since we can count functional subgraphs, and know that reverse arborescence subgraphs are a type of functional subgraph, our goal then becomes to extract the number of reverse arborescence subgraphs from the number of functional subgraphs. To do this, we must understand what characteristics distinguish the two types of subgraphs. This leads us to a discussion about cycles.

**Definition 3.4** A *cycle* in a digraph $G$ is a nonempty sequence of vertices $C = \{x_1, \ldots, x_k\}$ where $x_{i+1}$ is an outneighbor of $x_i$ for $1 \leq i < k$ and $x_1$ is an outneighbor of $x_k$. We require every vertex in $C$ to be unique.

Note that this allows $\{x\}$ - the sequence consisting of a single vertex - to be a cycle if and only if there is a self-loop at $x$. Specifically, the sink in a sink-rooted digraph is a cycle. We will call this the *sink self-loop cycle.*

**Definition 3.5** A digraph $G$ is called unicyclic if it contains exactly one cycle.

The following are two facts that follow immediately from our definition of cycles.

**Fact 3.6** *Let $F$ be a functional digraph. Then the following are true:* 1) *$F$ contains a cycle* 2) *If $x_0$ is a vertex in $V$, then $x_0$ is connected to the vertices of a cycle.*

**Proof:** Since $F$ is a functional digraph, $x_0$ has a single outneighbor $x_1$. Similarly $x_1$ has a single outneighbor $x_2$. Continue inductively to create an infinite walk $\{x_0, x_1, x_2, \ldots\}$. Since $G$ is finite, there exists $n \in \mathbb{N}$ such that $x_n = x_k$ for some $k < n$. Then $\{x_k, x_{k+1}, \ldots, x_{n-1}\}$ is a cycle in $F$, and $x_0$ is connected to those vertices. $\square$

We know have enough information to show that reverse arborescence subgraphs of a digraph $G$ are exactly its unicyclic functional sugraphs.

**Lemma 3.7** *Let $G$ be a sink-rooted digraph. Then all of its functional subgraphs contain the sink self-loop cycle.*

**Proof:** Let $F$ be a functional subgraph of $G$. Then the sink $s$ must have an outedge in $F$. This outedge must be one of the outedges of $s$ in $G$. But the sink has only one outedge in $G$, a self-loop. Hence the sink has a self-loop in $F$. $\square$

**Fact 3.8** *Let $G$ be a sink-rooted digraph, and $F$ a unicyclic functional subgraph of $G$. Then $F$ is a reverse arborescence.*

**Proof:** To show a digraph is a reverse arborescence, we must show that (1) It has a sink with exactly one self-loop, and (2) There exists a unique path from every site to the sink. The above lemma has already proven (1). So now we must show (2). Take $v \in V_0$. Then $v$ is connected to the vertices of a cycle by Fact 3.6. But $F$ is unicyclic, so the only cycle in $F$ is the sink self-loop cycle. Thus $v$ is connected to the sink, and there exists a path $P$ from $v$ to the sink. Since $F$ is a functional subgraph, all paths starting at $v$ must coincide with $P$. Thus $P$ is the unique path from $v$ to the sink, and we have shown (2). $\square$

Now we will show the converse, that if $F$ is a reverse arborescence subgraph of a sink-rooted digraph $G$, then $F$ is a unicyclic functional subgraph.

**Fact 3.9** *Let $F$ be a weakly connected functional digraph. Then $F$ is unicyclic.*

**Proof:** As shown in Fact 3.6, all functional digraphs have at least one cycle. Now we will use weak connectivity to show that this cycle is unique. We do this through the contrapositive. Suppose there exist two distinct cycles $C_1$ and $C_2$ in $F$, and there exists a direction-ignoring walk $\{x_1, \ldots, x_n\}$ where $x_1 \in C_1$ and $x_n \in C_2$. Let $a = \max\{k \in \mathbb{N} : x_k \in C_1\}$ and $b = \min\{k \in \mathbb{N} : x_k \in C_2\}$. Then $x_{a+1} \notin C_1$. Since $x_a$ is an element of $C_1$, its unique outneighbor must also be a member of that cycle. Then $x_{a+1}$ is not the outneighbor of $x_a$, so $x_a$ is the outneighbor of $x_{a+1}$.

Suppose we can find $c = \min\{k > a : x_{k+1} \text{ is an outneighbor of } x_k\}$. Then $x_c$ is not an outneighbor of $x_{c-1}$, so $x_{c-1}$ is an outneighbor of $x_c$. But $x_{c+1}$ is also an outneighbor of $x_c$. Since $G$ is a functional digraph, $x_c$ must have a unique outneighbor, so this is a contradiction. Hence $c$ does not exist and for all $a \leq i < n$, $x_i$ is an outneighbor of $x_{i+1}$.

Then $x_{b-1}$ is an outneighbor of $x_b$. But $x_{b-1} \notin C_2$, so it cannot be the outneighbor of an element of $C_2$. This contradiction implies that there is no direction-ignoring walk from an element of $C_1$ to an element of $C_2$. Hence if $F$ is not unicyclic, it is not weakly connected. $\square$

**Fact 3.10** *Let $G$ be a sink-rooted digraph, and let $F$ be a reverse arborescence subgraph of $G$. Then $F$ is a unicyclic functional digraph.*

**Proof:** Fact 3.3 already shows that $F$ is a functional subgraph of $G$, so all we must show is that $F$ is unicyclic. Since $F$ is a reverse arborescence, $F$ is weakly connected, and thus is a weakly connected functional subgraph. But the above fact shows that weakly connected functional digraphs are unicyclic. $\square$

We have now completed our foray into functional digraphs, having shown that reverse arborescence subgraphs are exactly the unicyclic functional subgraphs, and are now ready to count the reverse arborescence subgraphs of a sink-rooted digraph. To proceed with this task, we must borrow a principle from combinatorics.

**Fact 3.11** (Inclusion-Exclusion Priniciple) *Let $A_1, A_2, \ldots A_m$ be subsets of a finite universe $X$, the set which complements are taken with respect to. Let $[m] = \{1, 2, \ldots, m\}$. Define $B = X \setminus \bigcup_{i \in [m]} A_i$. Then*

$$|B| = \sum_{I \subset [m]} (-1)^{|I|} \cdot \left| \bigcap_{i \in I} A_i \right|,$$

*the sum over all subsets $I$ of $[m]$, where*

$$\bigcap_{i \in \emptyset} A_i = X$$

*by convention.*

This principle will not be proved here, if the reader wants, a more detailed analysis of it is given here[1]. However, the following is an informal explanation of the idea behind this principle. We want to calculate the size of the set $B = X \setminus \bigcup_{i \in [m]} A_i$, the set left over when the $A_i$ are removed from $X$. We begin this calculation with an approximation, taking the size of $X$, and subtracting from it the sizes of each of the $A_i$, giving us the expression $|B| \approx |X| - \sum_{i \in [m]} |A_i|$. However, the last term in this approximation exaggerates the total size of $\bigcup_{i \in [m]} |A_i|$ by over counting everywhere the $A_i$ overlap. So we must correct our approximation with an additional term, getting us to

$$|B| \approx |X| - \sum_{i \in [m]} |A_i| + \sum_{\substack{i,j \in [m] \\ i \neq j}} |A_i \cap A_j|.$$

But the latest term in this approximation exaggerates the overlap of the $A_i$, by over counting everywhere three members of the $A_i$ intersect. So the approximation must proceed to a term of order three, leaving us with

$$|B| \approx |X| - \left( \sum_{i \in [m]} |A_i| \right) + \left( \sum_{\substack{i,j \in [m] \\ i \neq j}} |A_i \cap A_j| \right) - \left( \sum_{\substack{I \in m \\ |I| = 3}} \left| \bigcap_{i \in I} A_i \right| \right)$$

Successively adding higher order terms leaves us with the Inclusion-Exclusion principle.

We are conceptually ready to apply this principle and begin counting arborescences, but just need a bit more notation to ease the process.

**Notation 3.12** Let $G$ be a sink-rooted digraph. Recall $\mathbf{X} = \{$functional subgraphs of $G\}$. We define $R_A = \{$reverse arborescence subgraphs of $G\}$. We will enumerate the cycles in $G$ that are not the sink self-loop cycle by $C_1, \ldots, C_m$. Then we define $A_i = \{$functional digraphs which contain $C_i\}$ for all $1 \leq i \leq m$. Lastly, for any such cycle $C_i$ and vertex $v \in C_i$ let $w$ be the outneighbor of $v$ along the cycle $C_i$. Then we will write $a_i(v) = a_{vw}$, and read this symbol as *the number of outedges of $v$ along the cycle $C_i$*.

Since all functional subgraphs of a sink-rooted digraph contain the sink self-loop cycle, the only unicyclic functional subgraphs are those that contain no other cycles. Thus $R_A = \mathbf{X} \setminus \bigcup_{i \in [m]} A_i$. Applying the above principle we get:

$$|R_A| = \sum_{I \subset [m]} (-1)^{|I|} \cdot \left| \bigcap_{i \in I} A_i \right| \tag{1}$$

Now to write out this formula in a more concrete form. First, recall the discussion above Fact 3.3, where we show that

$$\left| \bigcap_{i \in \emptyset} A_i \right| = |\mathbf{X}| = \prod_{v \in V_0} \deg^+(v)$$

This describes the zeroth order term of the summation. The next terms in the summation are the first order terms, which are just the individual $A_i$. Our goal then becomes to count the functional subgraphs of $G$ that contain a given cycle $C_i$. For every vertex not in the cycle, a functional subgraph containing the cycle has exactly one of its outedges. For every vertex in the cycle, the functional subgraph would have exactly one outedges from that vertex to the next vertex in the cycle. This leaves us with

$$|A_i| = \left( \prod_{v \notin C_i} \deg^+(v) \right) * \left( \prod_{v \in C_i} a_i(v) \right)$$

We then proceed to evaluate the size of the larger order terms.

**Fact 3.13** *Let $G$ be a sink-rooted digraph, and let $C_i$ and $C_j$ be two distinct cycles in $G$. Then $A_i \cap A_j = \emptyset$ if $C_i \cap C_j \neq \emptyset$.*

**Proof:** Let $S$ be a subgraph of $G$ containing both cycles. Supposing $C_i \cap C_j \neq \emptyset$, we can find a vertex $v \in C_i \cap C_j$. Write $C_i = \{w_1, \ldots w_k\}$ and $C_j = \{z_1, \ldots, z_l\}$ where $w_1 = z_1 = v$, and $w_{i+1}$ is an outneighbor of $w_i$ for $1 \leq i < k$ and $z_{j+1}$ is an outneighbor of $z_j$ for $1 \leq j < l$. Let $r = \min\{i \in \mathbb{N} : w_i \neq z_i\}$. Then $z_{r-1} = w_{r-1}$ is a single vertex which has two distinct outneighbors $w_i$ and $z_i$. That implies it has at least two outedges in $S$, so $S$ cannot be a functional digraph. □

The above result can be generalized to the following statement: if the $\{C_i\}_{i \in I}$ are not pairwise disjoint for any nonempty subset $I \subset [m]$, then $\cap_{i \in I} A_i = \emptyset$. Thus all we need to calculate is the the number of functional digraphs that contains a collection of pairwise disjoint cycles. But to do this, we treat the members of each cycle separately, as we did above for the individual $A_i$, and then choose any outedge of the remaining vertices. Thus for any nonempty subset $I \subset [m]$, we get that

$$\left| \bigcap_{i \in I} A_i \right| = \begin{cases} \left( \displaystyle\prod_{v \notin \bigcup_{i \in I} C_i} \deg^+(v) \right) * \displaystyle\prod_{i \in I} \left( \prod_{v \in C_i} a_i(v) \right) & \text{if the } C_i \text{ are pairwise disjoint} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

This, along with equation 1, gives us a concrete, if messy, way of counting reverse arborescences.

We are momentarily done counting reverse arborescences, and will begin analyzing the determinant of the sink-reduced Laplacian. For that we will need some theory of permutation groups, which will be used both to define the determinant and to make use of that definition. We will not build up this theory from scratch, but only state some of its key results and hope the reader is already acquainted with them. If the reader is not so acquainted, then they may find a useful reference here[4].

**Definition 3.14** Let $M = [m_{ij}]$ be a $k \times k$ matrix. We then say that the *determinant of $M$* is

$$\det(M) = \sum_{\sigma \in S_k} \operatorname{sgn}(\sigma) \prod_{v=1}^{k} m_{v,\sigma(v)}$$

where $S_k$ is the set of permutations on $k$ objects, and $\operatorname{sgn}(\sigma)$ is the sign of the permutation $\sigma$, equal to 1 if $\sigma$ can be decomposed into an even number of transpositions and $-1$ otherwise.

We are interested in the determinant because we are able to expand the permutations that make it up into cycles. It is through these cycles that we will relate the determinant to our count of reverse arborescences.

**Definition 3.15** Let $\phi$ be a permutation on $k$ objects. We say $\phi$ is a *cyclic permutation* if there exists an ordering of the objects $v_1, \ldots, v_k$ where $\phi(v_i) = v_{i+1}$ for $1 \leq i < k$ and $\phi(v_k) = v_1$. We say $\phi$ is a *permutation on a subset $S$ of $k$ elements*, if $\phi$ restricted to $S$ is a cyclic permutation, and $\phi$ restricted to $[k] \setminus S$ is the identity. We say that $\phi$ is a *proper cyclic permutation* if it is not the identity.

**Fact 3.16** *Let $\phi$ be a proper cyclic permutation on a subset $S$ of $k$ objects. Then $sgn(\phi) = (-1)^{|S|-1}$.*

The following fact is a basic result in the theory of permutations that is essential to our discussion of the determinant of the sink-reduced Laplacian.

**Fact 3.17** *Let $\sigma$ be a permutation on $k$ elements that is not the identity. Then there exist proper cyclic permutations $\phi_1, \ldots, \phi_N$ over pairwise disjoint subsets $S_1, \ldots, S_N$ of the $k$ elements, where $\sigma = \phi_1 \circ \cdots \circ \phi_N$.*

Since the $\phi_i$ act on pairwise disjoint subsets, it immediately follows that the order of the composition of the $\phi_i$ doesn't matter. It also follows from the disjointed property of the $S_i$ that for any permutation $\sigma$, the choice of the $\phi_i$ is unique. Thus what we are doing is *decomposing* $\sigma$ into its cyclic components.

**Fact 3.18** *Let $\sigma$ be a permutation on $k$ objects, and let its decomposition into proper cyclic permutations be $\phi_1, \ldots, \phi_N$. Then $sgn(\sigma) = \prod_{i=1}^{N} sgn(\phi_i)$.*

Now let us return to the problem at hand. If $G = (V, E)$ is a sink-rooted digraph with $V$ indexed from 1 to $n$ (the nth node being the sink), recall that the sink-reduced Laplacian of $G$ is the $(n-1) \times (n-1)$ matrix $L$ whose entries are

$$l_{ij} = \begin{cases} \deg^+(i) - a_{ii} & \text{if } i = j \\ -a_{ij} & \text{otherwise} \end{cases}$$

where $i, j$ are vertices that range from 1 to $n-1$. Then

$$\det(L) = \sum_{\sigma \in S_{n-1}} \operatorname{sgn}(\sigma) \prod_{v=1}^{n-1} l_{v,\sigma(v)}$$

We now wish to expand this expression into something workable. For every permutation $\sigma \in S_{n-1}$, let us decompose it into proper cyclic permutations $\phi_1, \ldots, \phi_N$ over disjoint subsets $S_1, \ldots, S_N$ of the $[n-1]$ nodes. Then we get that $\sigma(v) = v$ exactly when $v \notin \bigcup_{i=1}^{N} S_i$. Thus for $\sigma \in S_{n-1}$,

$$\operatorname{sgn}(\sigma) \prod_{v=1}^{n-1} l_{v,\sigma(v)} = \operatorname{sgn}(\sigma) \left( \prod_{v \notin \bigcup_{i=1}^{N} S_i} (\deg^+(v) - a_{vv}) \right) * \prod_{i=1}^{N} \left( \prod_{v \in S_i} -a_{v,\phi_i(v)} \right)$$

Now,

$$\operatorname{sgn}(\sigma) = \prod_{i=1}^{N} \operatorname{sgn}(\phi_i) = \prod_{i=1}^{N} (-1)^{|S_i|-1}$$

and,

$$\prod_{i=1}^{N} \left( \prod_{v \in S_i} -a_{v,\phi_i(v)} \right) = \left( \prod_{i=1}^{N} (-1)^{|S_i|} \right) * \prod_{i=1}^{N} \left( \prod_{v \in S_i} a_{v,\phi_i(v)} \right)$$

so when we multiply the two together many of the -1's will cancel. Thus we can rewrite the determinant as

$$det(L) = \sum_{\sigma \in S_{n-1}} (-1)^N \left( \prod_{v \notin \bigcup_{i=1}^{N} S_i} (\deg^+(v) - a_{vv}) \right) * \prod_{i=1}^{N} \left( \prod_{v \in S_i} a_{v,\phi_i(v)} \right) \tag{3}$$

where it is implicit in the notation that the choice of the $S_i$ and the $\phi_i$ is dependent on the individual $\sigma$.

For a sink-rooted digraph, we now have an expression for the number of its reverse arborescence subgraphs in equations 1 and 2, and an expression for the determinant of its sink-reduced Laplacian in equation 3. We then must show that these represent the same number. First we shall concentrate on the similarities between these two formulas. Let $C_i$ indexed over $I \subset [m]$ be a collection of pairwise disjoint cycles in $G$, where none of the cycles is a self-loop. Then the term corresponding to them in the count of the reverse arborescences is

$$(-1)^{|I|}\left|\bigcap_{i\in I} A_i\right| = (-1)^{|I|}\left(\prod_{v\notin\bigcup_{i\in I} C_i} \deg^+(v)\right) * \prod_{i\in I}\left(\prod_{v\in C_i} a_i(v)\right)$$

Now for each cycle $C_i$, let $\phi_i$ be the cyclic permutation whose image of a vertex $v \in C_i$ is the outneighbor of $v$ along $C_i$. Since the $C_i$ are not self-loops, the $\phi_i$ are proper cyclic permutations. Since the $C_i$ are disjoint, the $\phi_i$ uniquely determine a permutation $\sigma$ of the sites in $G$. Then there is a term in the expansion of the determinant corresponding to $\sigma$, which is

$$\operatorname{sgn}(\sigma)\prod_{v=1}^{n-1} l_{v,\sigma(v)} = (-1)^N\left(\prod_{v\notin\bigcup_{i=1}^N S_i} (\deg^+(v) - a_{vv})\right) * \prod_{i=1}^{N}\left(\prod_{v\in S_i} a_{v,\phi_i(v)}\right)$$

These two terms should seem very similar. Since each $\phi_i$ sends vertices along the cycle $C_i$, we get that $a_{v,\phi_i(v)} = a_i(v)$. There are also some notational differences; one term is indexed over $I$ while the other is indexed from 1 to $N$, and in one term the cycles are called $C_i$ while in the other they are called $S_i$, but the number these two terms represent is almost the same. The one caveat is that the latter term has a product of $\deg^+(v) - a_{vv}$ for vertices not in any cycle, while the former only multiplies by $\deg^+(v)$. We will explain this away latter.

For now we will concentrate on the differences between the domains of the sums in each expression. Our count of reverse arborescences sums over the $C_i$, representing all possible combinations of the cycles found in $G$. Our expression for the determinant sums over the permutations $\sigma$, representing all possible combinations of disjoint cycles of $n-1$ vertices. The differences in these sums leaves two discrepancies.

First, our count of reverse arborescences has a term for every collection of cycles $C_i$ for $I \subset [m]$, not just those collections that are pairwise disjoint. When a collection of cycles is not pairwise disjoint there can be no corresponding term in the expansion of the determinant. This is because there must be a node which has two different outneighbors under two different cycles, a phenomena that no permutation can represent. However, we already showed that such terms do not contribute to the count of reverse arborescences, because there are no functional digraphs which contain two distinct but intersecting cycles. As such, we can dismiss this concern.

Our second concern is that there exist permutations in $S_{n-1}$ whose cyclic decompositions contain cycles of the $n-1$ vertices that are not present in $G$. For each of these terms in the expansion of the determinant, there can be no corresponding term in our count of reverse arborescences. However, we will be able to dismiss this worry just as we did above. Suppose $\phi : V_0 \to V_0$ is a proper cyclic permutation corresponding to a sequence of vertices $C = \{c_1, \dots c_k\}$, where $C$ is not a cycle in $G$. This implies that there must be a node $c_j$ which has no outedges towards $\phi(c_j)$. Let $\sigma$ be any permutation in $S_{n-1}$ whose decomposition into proper cyclic permutations contains $\phi$. Then the term for $\sigma$ in our expansion will be a product which contains $\prod_{v\in C} a_{v,\phi(v)}$, which in turn contains $a_{c_j,\phi(c_j)}$, which is zero. Thus such terms are zero and do not contribute to the determinant of the sink-reduced Laplacian.

Thus, despite appearances otherwise, there is an exact correspondence between the non-zero terms of each sum. However, there are two last deviations between the two formulae that must be accounted for. First of all, where each term in the count of reverse arborescences multiply by

$$\prod_{v\notin\bigcup_{i\in I} C_i} \deg^+(v)$$

the terms in the determinant expansion multiply by

$$\prod_{v \notin \bigcup_{i=1}^{N} S_i} (\deg^+(v) - a_{vv})$$

Second, the terms of the sum in the expansion of the determinant cannot represent self-loop cycles, for the decomposition of a permutation is only into proper cyclic permutations. Where a term in the expansion of the determinant would need to represent a self-loop cycle at $v$ by multiplying by $a_{vv}$, it instead multiplies by $l_{vv} = \deg^+(v) - a_{vv}$.

In fact, these two deviations end up reconciling one another. To see this, take $I \subset [m]$ so that the $C_i$ are pairwise disjoint and none of them are self-loops. Index the sites not in any of the $C_i$ by $s_1, \dots s_k$. Then there will be a term for $I$ in the count of reverse arborescences, namely

$$(-1)^{|I|} \left( \prod_{i=1}^{k} \deg^+(s_i) \right) * \prod_{i \in I} \left( \prod_{v \in C_i} a_i(v) \right)$$

Let $D_1$ be the self-loop cycle at $s_1$. Then there will be a term corresponding to the functional subgraphs that contain the $C_i$ and $D_1$, namely

$$(-1)^{|I|-1} * a_{s_1 s_1} * \left( \prod_{i=2}^{k} \deg^+(s_i) \right) * \prod_{i \in I} \left( \prod_{v \in C_i} a_i(v) \right)$$

Adding the two leaves us with

$$(-1)^{|I|} * \left( \deg^+(s_1) - a_{s_1 s_1} \right) * \left( \prod_{i=2}^{k} \deg^+(s_i) \right) * \prod_{i \in I} \left( \prod_{v \in C_i} a_i(v) \right)$$

Let $D_2$ be the self-loop cycle at $s_2$. Then there will be a term for the functional digraphs that contain the $C_i$ and $D_2$, and a term for those that contain the $C_i$ and $D_1$ and $D_2$. Adding those two terms together leaves,

$$(-1)^{|I|-1} * \left( \deg^+(s_1) - a_{s_1 s_1} \right) * a_{s_2 s_2} * \left( \prod_{i=3}^{k} \deg^+(s_i) \right) * \prod_{i \in I} \left( \prod_{v \in C_i} a_i(v) \right)$$

Adding this to our original computation leaves us with

$$(-1)^{|I|} * \left( \prod_{i=1}^{2} \left( \deg^+(s_i) - a_{s_i s_i} \right) \right) * \left( \prod_{i=3}^{k} \deg^+(s_i) \right) * \prod_{i \in I} \left( \prod_{v \in C_i} a_i(v) \right)$$

Continuing this process inductively will eventually leave us with

$$(-1)^{|I|} * \left( \prod_{i=1}^{k} \left( \deg^+(s_i) - a_{s_i s_i} \right) \right) * \prod_{i \in I} \left( \prod_{v \in C_i} a_i(v) \right)$$

which is a term in the expansion of the determinant. In other words, each term of the expansion of the determinant already accounts for the adjustments due to self-loops of the corresponding term in the count of reverse arborescences. This being the last discrepancy between the two formulae, we conclude that they are equal, and that for a digraph $G$, the number of its subgraphs that are reverse arborescences is the determinant of its sink-reduced Laplacian.

**Acknowledgements** For helping me to write a coherent paper, I gladly thank my mentors Zachary Madden and Professor László Babai. I would also like to thank Professor Babai for the course he taught this REU, in which I was introduced to this theorem and its context within the fascinating Abelian Sandpile Model.

# References

[1] Andrews, G. E: Number Theory. Philadelphia, PA: Saunders, 1971.

[2] Chaiken, S: *A Combinatorial Proof of the All-Minors Matrix Tree Theorem* SIAM J. Alg. Disc. Methods 3, 319-329, 1982.

[3] Kirchhoff, G: *Über die Auflösung der Gleichungen, auf welche man bei der untersuchung der linearen verteilung galvanischer Ströme geführt wird.* Ann. Phys. Chem. 72, 497-508, 1847.

[4] Sally, P. Jr: Tools of the Trade: Introduction to Advanced Mathematics. American Mathematical Society, 2008