INPUT-OUTPUT RANDOM WALK NETWORKS

EDDIE LISTER

Abstract. Motivated by Curtis and Morrow's inverse problem on electrical networks [1], we explore a similar problem on probabilistic networks. We define a new sort of network, with specified "input" and "output" states, generalizing those presented by past REU participants Schroder [7] and DeVries [2].

We develop an analog of Curtis and Morrow's determinant-connection formula, which we show reduces to theirs when we consider random walks on electrical networks. With this formula and other machinery, we classify certain networks which are demonstrably not recoverable and present techniques that may prove useful in general recovery algorithms.

1. Introduction

We are interested in a random walk network, modeled as a Markov chain, on which walks always begin on one particular set of states (to which it never returns) and end as soon as they reach another set of states. Note that we typically visualize Markov chains as graphs whose vertices (or nodes) are given by the states with directed, weighted edges given by the non-zero transition probabilities. As such, we may use the terms state, vertex, or node interchangeably, and it makes sense to refer to edges in a Markov chain.

Definition 1.1 (Input-Output Random Walk Networks)**.** An input-output random walk network $\mathcal{P} = (S, P)$ is a Markov chain (i.e. a set S of states and an $|S| \times |S|$ transition matrix P) which satisfies the following properties:

(1) The state space S is partitioned into input (∂_{in}), interior (int), and output (∂_{out}) subsets such that
  (a) $S = \partial_{in} \cup \text{int} \cup \partial_{out}$;
  (b) ∂_{in}, int, and ∂_{out} are pairwise disjoint; and
  (c) ∂_{in} and ∂_{out} are non-empty.
(2) There is an ordering of states $(s_1, \dots, s_{|S|})$ given, with output states first and input states last.
(3) $j \in \partial_{in} \implies P_{ij} = 0$.
(4) $i \in \partial_{out} \implies P_{ii} = 1$.
(5) For each $s_0 \in \text{int}$, there exists a sequence of states $(s_0, \dots, s_k)$ such that $s_k \in \partial_{out}$ and $P_{s_{i-1}s_i} > 0 \ \forall \ i \in (1, \dots, k)$.

We deal exclusively with input-output random walk networks, so we will often find it convenient to omit "input-output" and refer to them simply as random walk networks. Note that our random walk networks are different from those presented by other authors, such as DeVries [2], so the reader is advised to distinguish them as input-output networks when this omission could cause confusion.

Since the entries of P represent particular probabilities, we have $0 \leq P_{ij} \leq$ and $\sum_{j=1}^{|S|} P_{ij} = 1$, as usual for Markov chains. With the ordering of states in mind, we can write the transition matrix P of the network in block form:

$$P = \begin{array}{c c} & \begin{array}{ccc} \partial_{out} & \text{int} & \partial_{in} \end{array} \\ \begin{array}{c} \partial_{out} \\ \text{int} \\ \partial_{in} \end{array} & \begin{pmatrix} I & 0 & 0 \\ R & Q & 0 \\ S & T & 0 \end{pmatrix} \end{array},$$

where I is the $|\partial_{out}| \times |\partial_{out}|$ identity matrix, R is a $|\text{int}| \times |\partial_{out}|$ matrix, Q is a $|\text{int}| \times |\text{int}|$ matrix, S is a $|\partial_{in}| \times |\partial_{out}|$ matrix, and T is a $|\partial_{in}| \times |\text{int}|$ matrix. Note that a network $\mathcal{P}$ is completely determined by the matrices R, Q, S, and T.

Recalling that the entry P_{ij} represents the probability that a node in state i at time t will be in state j at time $t+1$, it follows that the entry P_{ij}^n of the nth power of P represents the probability that a node in state i at time t will be in state j at time $t+n$. We are interested in taking the limit of P^n as $n \to \infty$;

assuming this limit exists, its ijth entry gives the probability that a random walk starting from state i will get stuck in state j. Our condition that there exists a path from each interior state to the boundary implies convergence:

$$\lim_{n\to\infty} P^n = P^\infty = \begin{array}{c} \\ \partial_{out} \\ \text{int} \\ \partial_{in} \end{array} \begin{array}{c} \begin{array}{ccc} \partial_{out} & \text{int} & \partial_{in} \end{array} \\ \left(\begin{array}{ccc} I & 0 & 0 \\ B & 0 & 0 \\ L & 0 & 0 \end{array} \right) \end{array},$$

where $B = (I - Q)^{-1}R$ and $L = S + TB$. In particular, we are interested in the matrix L, whose entry L_{ij} gives the probability that a random walk starting at input state i will end in output state j. We call L the input-to-output probability map for $\mathcal{P}$.

Since any P^n is a valid transition matrix for some Markov chain, this gives us $0 \le P^\infty_{ij} \le 1$ and $\sum_j P^\infty_{ij} = 1$. Moreover, since we know certain zeros of this matrix, we have $0 \le L_{ij} \le 1$ and $\sum_j L_{ij} = 1$. (The same properties also hold for B, but we are most interested in L.)

That this limit actually converges and that our formula for this infinite product is actually correct are not immediately obvious, but these can be shown analytically. In fact, our random walk network is a special case of an absorbing Markov chain (with the outputs as absorbing states), discussed in great detail by Doyle and Snell [3]. Our formulas for both B and L can be verified by applying block matrix algebra to their techniques.

2. The Determinant-Connection Formula for Input-Output Random Walk Networks

Before deriving an important formula for determinants of submatrices of L, we must first take care to define a particular random process to which they are deeply related. It is helpful to think of our input-output random walk network as a directed graph whose vertices (or nodes) are given by the states of the Markov chain (preserving both ordering and partitioning); a directed edge from vertex i to vertex j exists if and only if $P_{ij} > 0$.

Consider the random process in which we choose exactly one edge directed away from each input and interior vertex. These edges are chosen independently, and a particular edge $e = ij$ is chosen with probability $p(e) = P_{ij}$ taken from the transition matrix. Thus the probability of choosing a particular set of edges E is given by

$$P(E) = \prod_{e\in E} p(e).$$

We see that the collection of all such valid sets of edges and the above probability function give us a well-defined probability space. Note that we do not include the self-loops at output vertices.

We define two more events on this space before we present our desired formula.

Definition 2.1 (Spanning Trees)**.** We let T denote the event that a set of edges E does not contain any cycles. That is, there is no (nonempty) sequence $(v = v_0, \dots, v_n = v)$ of interior nodes such that $v_{i-1}v_i \in E \;\; \forall \;\; i \in (1, \dots, n)$.

Note that this definition is equivalent to the graph theoretic notion that the edges in E (viewed as undirected) form a spanning tree (more properly called a spanning forest) for the graph with roots in the output vertices. However, we must take much greater care than shown here to define spanning trees in general.

Definition 2.2 (K-Connections)**.** Suppose we have ordered sets $P = (p_1, \dots, p_k)$ and $Q = (q_1, \dots, q_k)$ of input and output nodes, respectively. Then we say there is a k-connection from P to Q if there exist a permutation τ of $(1, \dots, k)$ and a set of vertex-disjoint paths $\alpha = (\alpha_1, \dots, \alpha_k)$ such that each α_i is a path from p_i to $q_{\tau(i)}$. We call α a k-connection from P to Q and call $\tau = \tau_\alpha$ the associated permutation of α.

Suppose the ordered sets P of input nodes and Q and of output nodes (again of size k) are given, along with a permutation τ of $(1, \dots, k)$. We let A_τ represent the event that a given set of edges contains every edge in some k-connection between P and Q whose associated permutation is τ.

Finally, we are ready to state our main theorem, the determinant-connection formula for input-output random walk networks.

Theorem 2.3 (Determinant-Connection Formula)**.** *Suppose we have an input-output random walk network with input-to-output probability map L. Let $P = (p_1, \ldots, p_k)$ and $Q = (q_1, \ldots, q_k)$ be any ordered subsets of input and output states. Let L_{PQ} be the $k \times k$ submatrix of L whose ijth entry is given by $L_{p_i q_j}$. Then*

$$\det(\mathrm{L_{PQ}}) = \sum_{\tau \in \mathrm{S_k}} \mathrm{sgn}(\tau)\mathrm{P}(\mathrm{A}_\tau|\mathrm{T}),$$

where S_k represents the symmetric group on the first k natural numbers (i.e. the set of permutations of the integers 1 through k), and $\mathrm{sgn}(\tau)$ *is the sign, or signature, of the permutation τ ($+1$ if τ is even and -1 if τ is odd).*

The determinant-connection formula follows from a theorem by Fomin which relates the determinant of L_{PQ} to the probabilities of certain random walks. The rest of the argument employs Wilson's method for generating spanning trees with our desired distribution from random walks, a proof of which is given by Lyons and Peres.

3. Proving the Determinant-Connection Formula

We first begin with a few definitions that are vital to the theorem by Fomin. Despite their importance here and elsewhere, these definitions and the following two theorems will not be necessary for other sections of this paper.

Definition 3.1. A walk between input state v_{in} and output state v_{out} is a sequence of states $\pi = (v_0, \ldots, v_n)$ such that

(1) $v_0 = v_{in}$,
(2) $v_n = v_{out}$, and
(3) $P_{i-1,i} > 0 \;\forall\; i \in (1, \ldots, n)$.

We say that the probability of this walk is given by

$$P(\pi) = \prod_{i=1}^{n} P_{i-1,1}.$$

Intuitively, a walk from v_{in} is any path that a random walk starting in that state might take, and the probability of such a walk is exactly the chance that a random walk would follow that particular sequence. Note that, if there are any cycles in the interior of a random walk network, walks may be arbitrarily long and still have nonzero probability; however, summing over all possible walks from a given input will always give us probability 1. The next definition comes from Fomin [4].

Definition 3.2. Let $LE(\pi)$ denote what we call the loop-erased part of a walk π. If $\pi = (v_0, \ldots, v_n)$ contains each state at most once (that is, it does not intersect itself), then $LE(\pi) = \pi$. Otherwise, take the smallest j such that $v_i = v_j$ and let π' denote the walk formed by removing states $(v_{i+1}, \ldots, v_j)$ from π; then $LE(\pi) = LE(\pi')$.

Though we have defined the process of loop-erasure recursively, it must necessarily terminate because all walks on our input-output random walk networks are necessarily finite. With these two definitions in mind, we are now ready to present a slightly modified version of Fomin's theorem [4].

Theorem 3.3. *Suppose we have an input-output random walk network with input-to-output probability map L. Given subsets $P = (p_1, \ldots, p_k)$ and $Q = (q_1, \ldots, q_k)$ of input and output states, respectively,*

$$\det(\mathrm{L_{PQ}}) = \sum_{\tau \in \mathrm{S_k}} \mathrm{sgn}(\tau) \sum_{(\pi_1,\ldots,\pi_\mathrm{k}):\mathrm{i}<\mathrm{j} \implies \mathrm{LE}(\pi_\mathrm{i})\cap\pi_\mathrm{j}=\emptyset} \prod_{\mathrm{i}=1}^{\mathrm{k}} \mathrm{P}(\pi_\mathrm{i}),$$

where each π_i is a walk from p_i to $q_{\tau(i)}$ and the condition of the second sum is that, in a given set of walks, each successive one does not intersect the loop-erased part of any previous walk.

Proof. Recall from the definition of L that

$$\det(\mathrm{L_{PQ}}) = \sum_{\tau \in \mathrm{S_k}} \mathrm{sgn}(\tau) \sum_{(\pi_1,\ldots,\pi_\mathrm{k})} \prod_{\mathrm{i}=1}^{\mathrm{k}} \mathrm{P}(\pi_\mathrm{i}).$$

With this in mind, we see that Fomin's theorem places a much stronger constraint on the terms in the second summation. Following his proof [4], we will show that each term from the definition that is not in his statement cancels out with a corresponding term of the opposite sign.

Let us first consider the case $k = 2$. Suppose we have a pair of walks π_1 from p_1 to $q_{\tau(1)}$ and π_2 from p_2 to $q_{\tau(2)}$ such that π_2 intersects $LE(\pi_1)$. Let v denote the point of intersection that is closest to p_1 along $LE(\pi_1)$. We partition π_1 at the unique occurrence of v that contributes to $LE(\pi_1)$ to obtain the walks π_1' from p_1 to v and π_1'' from v to $q_{\tau(1)}$ such that π_1'' does not intersect $LE(\pi_1')$ at any point except perhaps v.

For convenience, let $L = LE(\pi_1')$. We now partition π_2 at its first visit to v to obtain the walks π_2' from p_2 to v and π_2'' from v to $q_{\tau(2)}$ such that π_2' only intersects L at its endpoint v and π_2'' only intersects L at v.

Recall that our original pair of paths is $\pi_1 = (\pi_1', \pi_1'')$ from p_1 to $q_{\tau(1)}$ and $\pi_2 = (\pi_2', \pi_2'')$ from p_2 to $q_{\tau(2)}$. We now claim that the original pair cancels with the new pair $\tilde{\pi}_1 = (\pi_1', \pi_2'')$ from p_1 to $q_{\tau(2)}$ and $\tilde{\pi}_2 = (\pi_2', \pi_1'')$ from p_2 to $q_{\tau(1)}$.

It is clear that these two pairs are associated with permutations of opposite signs and carry the same probability, so they will exactly cancel in the sum. Applying the same procedure to the new pair $(\tilde{\pi}_1, \tilde{\pi}_2)$ will recover (π_1, π_2), so there is a one-to-one correspondence of unwanted terms of opposite signs, which completes the proof for $k = 2$. We must now generalize this proof to arbitrary k.

Consider a family of walks $(\pi_1, \ldots, \pi_k)$ which we wish to remove from the sum. Consider the nonempty set of triples (i, j, v) where $1 \leq i < j \leq k$ and $v \in \mathrm{int}$ such that π_j intersects $LE(\pi_i)$ at v. From this set, we choose the triple (i, j, v) such that

(1) i is as small as possible;
(2) given i, v is closest to p_i along $LE(\pi_i)$ for all intersections with π_j where $j > i$; and
(3) given i and v, j is the first value after i for which π_j passes through v.

With the paths π_i and π_j now chosen, we repeat the above procedure with these two paths, keeping all other $k - 2$ paths fixed. This again produces a one-to-one correspondence between undesired terms, with the procedure recovering exactly the original paths when applied to the new ones. □

With Fomin's theorem in hand, we must now present a relationship between random walks and spanning trees. Wilson's method, as presented by Lyons and Peres, provides an algorithm for generating spanning trees (as described previously) from random walks [6]:

(1) Begin with an empty tree T.
(2) Choose any input or interior vertex v that is not contained in T.
(3) Start a random walk from v and continue until it reaches an output node or any vertex already contained in T. Call the walk given by this path π.
(4) Add $LE(\pi)$, the loop erasure of π, to T.
(5) If T is a spanning tree, we are done. Otherwise, return to step 2.

It should be clear that Wilson's algorithm will always generate a valid spanning tree. The distribution of these trees, however, is not obvious. Lyons and Peres state and prove the following theorem about the distribution [6].

Theorem 3.4. *The distribution of spanning trees generated by Wilson's method is identical to that generated by independently choosing one edge from each input and interior vertex based on random walk probabilities and conditioning on the event that a valid spanning tree is produced.*

Perhaps the most remarkable part of this theorem is that the choice of starting points for walks in Wilson's algorithm is completely arbitrary. In fact, it may even depend on the outcome of previous random walks without changing the distribution of spanning trees! Before proving this theorem, we will first state and prove a lemma by Lyons and Peres that handles the problem of ordering [6].

Since we are dealing with a Markov chain, the states which follow every visit to a state x are independent and identically distributed. For every state x, we can imagine this as an infinite stack $(S_1^x, S_2^x, , S_3^x, \ldots)$, where S_i^x is the state we visit after the ith visit to x. We imagine these stacks as being generated randomly according to the Markov chain, but given ahead of time.

We think of each pair (x, S_i^x) as a directed edge in the graph of the network, and we say that it has color i. Suppose we have a set of stacks whose first entries give us at least one cycle in the graph. We call each cycle, along with the colors of its edges, a colored cycle, and we remove them by "popping" (removing) the first entry of each stack in the cycle. We can repeat this procedure as long as there are cycles in the graph. Note that we can pop a colored cycle at most once and that the first cycle removed has color 1 for every edge.

Lemma 3.5. *The order in which we pop cycles from the graph cannot change the end result. That is, any ordering will either pop the same finitely many colored cycles or continue popping cycles indefinitely.*

Proof. Suppose C is any colored cycle that we can pop. Then there is a sequence of colored cycles $(C_1, \ldots, C_n = C)$ which we can pop in that order. If we first pop some colored cycle $C' \neq C_1$, we must show that C can still be popped. If $C' = C$ or C' is disjoint from each $(C_1, \ldots, C_n)$, then the proof is trivial.

In the non-trivial case, let C_k be the first of $(C_1, \ldots, C_n)$ with which C' shares a vertex. Since C' is popped first, all its edges have color 1. If we let x be any vertex in $C' \cap C_k$, then $x \notin \bigcup_{i=1}^{k-1} C_i$ implies that the edge from x in C_k has color 1. As a vertex and a coloring uniquely determine an edge, we see that the edges from x in both C' and C_k lead to the same vertex.

Since the same argument applies for the successor of x, we have that $C' = C_k$. Thus C can be popped by the sequence $C_k = C', C_1, \ldots, C_{k-1}, C_{k+1}, \ldots, C_n = C$. Therefore, the process will either never terminate or it will terminate only after the same colored cycles have been removed, regardless of the order in which we choose to pop cycles. □

We note that, after removal of all cycles, the first entries of each stack give us a valid spanning tree. With this lemma and the construction of using stacks to run a Markov chain, we are now ready to present Lyons and Peres' proof that Wilson's method yields the desired distribution of Markov chains [6].

Proof of Wilson's Method. Given our stipulation that each interior vertex has a path to the output, the removal of cycles from stacks will stop at a spanning tree with probability 1. We now consider Wilson's algorithm as being run with stacks for each state. In Wilson's method, we remove cycles in the order that they are encountered by a random walk; as we have seen, this is just one way of choosing which cycles to remove and does not change the underlying spanning tree.

Certainly, we are able to cover any valid spanning tree with any finite set of valid cycles. Due to the independence inherent in a Markov chain, we see that the probability of obtaining a given spanning tree with Wilson's method is independent of whatever cycles happen to occur in our random walks. □

Thanks to the vital link between random walks and spanning trees given by Wilson's algorithm, the proof of the determinant-connection formula for input-output random walk networks follows almost trivially from Fomin's theorem.

Proof of the Determinant-Connection Formula. Given an input-output random walk network with input-to-output probability map L, along with sets of input and output vertices $P = (p_1, \ldots, p_k)$ and $Q = (q_1, \ldots, q_k)$, respectively, Fomin's theorem gives us

$$\det(\mathrm{L_{PQ}}) = \sum_{\tau \in \mathrm{S_k}} \mathrm{sgn}(\tau) \sum_{(\pi_1, \ldots, \pi_k): \mathrm{i<j} \implies \mathrm{LE}(\pi_\mathrm{i}) \cap \pi_\mathrm{j} = \emptyset} \prod_{\mathrm{i=1}}^{\mathrm{k}} \mathrm{P}(\pi_\mathrm{i}).$$

We can expand the second sum of the above formula in terms of k-connections $\alpha = (\alpha_1, \ldots, \alpha_k)$ from P to Q associated with the current permutation τ:

$$\det(\mathrm{L_{PQ}}) = \sum_{\tau \in \mathrm{S_k}} \mathrm{sgn}(\tau) \sum_{\alpha: \tau_\alpha = \tau} \sum_{(\pi_1, \ldots, \pi_k): \mathrm{LE}(\pi_\mathrm{i}) = \alpha_\mathrm{i}, \mathrm{i<j} \implies \alpha_\mathrm{i} \cap \pi_\mathrm{j} = \emptyset} \prod_{\mathrm{i=1}}^{\mathrm{k}} \mathrm{P}(\pi_\mathrm{i}).$$

We now consider running Wilson's method starting from each of the nodes in P. With the relation between random walks and spanning trees, we see that the last summation gives exactly the probability that a spanning tree for the network contains the k-connection α. We now have

$$\det(\mathrm{L_{PQ}}) = \sum_{\tau \in \mathrm{S_k}} \mathrm{sgn}(\tau) \sum_{\alpha : \tau_\alpha = \tau} \mathrm{P}(\alpha | \mathrm{T}).$$

Rather than sum over all possible k-connections, we let A_τ denote the event that a set of edges contains a k-connection from P to Q associated with the permutation τ. Encapsulating all the terms in the final sum, this gives us our desired form and completes the proof:

$$\det(\mathrm{L_{PQ}}) = \sum_{\tau \in \mathrm{S_k}} \mathrm{sgn}(\tau) \mathrm{P}(\mathrm{A}_\tau | \mathrm{T}).$$

□

4. Random Walks on Electrical Networks

We now consider electrical networks, which consist of a set of vertices V, partitioned into disjoint boundary and interior vertices, typically denoted by ∂V and $\mathrm{int} V$, respectively, a set of undirected edges E joining pairs of vertices, and a positive-valued conductance function γ defined on the edges. The network is denoted by $\Gamma = (V, E, \gamma)$.

We will assume that there is at most one edge between each pair of vertices and no self-loops. We let γ_{ij} denote the conductivity of the edge between vertices i and j, or zero if there is no such edge. Like our transition matrix P for input-output random walk networks, an electrical network is characterized the Kirchhoff matrix K, whose entries are given by

$$K_{ij} = \begin{cases} -\gamma_{ij} & \text{if } i \neq j, \\ \sum_{k \neq i} \gamma_{ik} & \text{if } i = j. \end{cases}$$

For an electrical network with n boundary vertices and m interior vertices, the Kirchhoff matrix is an $(n+m) \times (n+m)$ symmetric matrix, with entries corresponding to boundary vertices given first and those associated with interior vertices given last. It is often convenient to write K in the block form

$$K = \begin{array}{cc} & \begin{array}{cc} \partial V & \mathrm{int} V \end{array} \\ \begin{array}{c} \partial V \\ \mathrm{int} V \end{array} & \begin{pmatrix} A & B \\ B^T & C \end{pmatrix} \end{array},$$

where A is an $n \times n$ symmetric matrix, B is an $n \times m$ matrix, and C is an $m \times m$ symmetric matrix.

For a given electrical network, we define its associated input-output random walk network by considering a boundary-to-boundary random walk on the electrical network. We disallow self-loops and set transition probabilities proportional to conductances, as we imagine a random walker (perhaps a charged particle) will tend to follow the path of least resistance. That is, we want our transition probabilities to be given by

$$P_{ij} = \frac{\gamma_{ij}}{\sum_k \gamma_{ij}}.$$

However, if we define the random walk as such, it will never actually terminate. To solve this problem, we partition each boundary vertex into input and output states.

For notational convenience, we define a modified form of the transition matrix in block form by

$$M = \begin{pmatrix} S & T \\ R & Q \end{pmatrix}$$

We also find it useful to split the Kirchhoff matrix into its diagonal (D) and off-diagonal ($\tilde{K} = D - K$) entries. In block form,

$$D = \begin{pmatrix} D_A & 0 \\ 0 & D_C \end{pmatrix} \text{ and } \tilde{K} = \begin{pmatrix} \tilde{A} & \tilde{B} \\ \tilde{B}^T & \tilde{C} \end{pmatrix}.$$

This construction allows us to define a simple relation between the Kirchhoff matrix of an electrical network and the transition map of its associated input-output random walk network. We find $M = D^{-1}\tilde{K}$; that is,

$$\begin{pmatrix} S & T \\ R & Q \end{pmatrix} = \begin{pmatrix} D_A^{-1} & 0 \\ 0 & D_C^{-1} \end{pmatrix} \begin{pmatrix} \tilde{A} & \tilde{B} \\ \tilde{B}^T & \tilde{C} \end{pmatrix} = \begin{pmatrix} D_A^{-1}\tilde{A} & D_A^{-1}\tilde{B} \\ D_C^{-1}\tilde{B}^T & D_C^{-1}\tilde{C} \end{pmatrix}.$$

Since we now know Q, R, S, and T, we have a full characterization of this particular random walk network.

An important object of study with regard to electrical networks is the response matrix (also called the Dirichlet-to-Neumann map), denoted by Λ and defined by the Schur complement of C in K. Explicitly, we have

$$\Lambda = A - BC^{-1}B^T.$$

From this definition, we are able to derive a relationship between Λ and the input-to-output probability map L. This form was given by Fomin [4], though a weaker version was shown earlier by Schroder [7].

Proposition 4.1. *Suppose we have an electrical network with response matrix Λ and diagonal matrix D_A corresponding to the boundary portion of the Kirchhoff matrix. Then the input-to-output probability map of its associated input-to-output random walk network is*

$$L = I - D_A^{-1}\Lambda.$$

Proof. This is a straightforward algebraic verification from the definition of L.

$$\begin{aligned}
L &= S + T(I-Q)^{-1}R \\
&= D_A^{-1}\tilde{A} + D_A^{-1}\tilde{B}(I - D_C^{-1}\tilde{C})^{-1}D_C^{-1}\tilde{B}^T \\
&= D_A^{-1}\tilde{A} + D_A^{-1}\tilde{B}(D_C(I - D_C^{-1}\tilde{C}))^{-1}\tilde{B}^T \\
&= D_A^{-1}\tilde{A} + D_A^{-1}\tilde{B}(D_C - \tilde{C})^{-1}\tilde{B}^T \\
&= D_A^{-1}(D_A - A) + D_A^{-1}(-B)C^{-1}(-B)^T \\
&= I - D_A^{-1}(A - BC^{-1}B^T) \\
&= I - D_A^{-1}\Lambda
\end{aligned}$$

□

With the relationship between L and Λ established, we would like to apply the determinant-connection formula to electrical networks. Though what we will show is a well-known formula due to Curtis and Morrow [1], we provide an alternative proof. First, however, we require a theorem due to Lewandowski [5].

Theorem 4.2. *Suppose Γ is an electrical network with Kirchhoff matrix*

$$K = \begin{pmatrix} A & B \\ B^T & C \end{pmatrix}.$$

Then, letting $\mathcal{T}$ be the set of spanning trees of Γ rooted in its boundary vertices,

$$\det(C) = \sum_{T\in\mathcal{T}} \prod_{e\in T} \gamma(e).$$

Noting that the Kirchhoff matrix alone (without partitioning into the canonical block form) does not distinguish between interior and boundary vertices, we can easily generalize Lewandowski's theorem.

Corollary 4.3. *Suppose P is a subset of the vertices V of an electrical network Γ with Kirchhoff matrix K. Letting K_P denote the principal submatrix of K corresponding to the vertices in P,*

$$\det(K_P) = \sum_{T\in\mathcal{T}} \prod_{e\in T} \gamma(e),$$

where $\mathcal{T}$ is the set of spanning trees of Γ rooted in $V \setminus P$.

Before giving the determinant-connection formula, we must first, of course, define what we mean by k-connections in electrical networks. The definition is taken to be the same as for our random walk networks, except that they are between disjoint sets of boundary vertices rather than input and output states. Without the partitioning of boundary states, it is difficult to interpret what it would mean for these sets to overlap; moreover, the fact that they are disjoint is vital to the proof.

Theorem 4.4. *Suppose we have an electrical network Γ with Kirchhoff matrix K (represented in canonical block form) and response matrix Λ. Let $P = (p_1, \ldots, p_k)$ and $Q = (q_1, \ldots, q_k)$ be disjoint sets of boundary vertices. Then, if Λ_{PQ} is the $k \times k$ submatrix of Λ whose ijth entry is given by $\Lambda_{p_i q_j}$,*

$$\det(\Lambda_{PQ}) \det(C) = (-1)^k \sum_{\tau \in S_k} \operatorname{sgn}(\tau) \sum_{\alpha : \tau_\alpha = \tau} \det D_\alpha \prod_{e \in \alpha} \gamma(e),$$

where D_α is the determinant of the principal submatrix of K corresponding to interior nodes of Γ not used in the k-connection α (or 1 if this set is empty).

Proof. For this proof, let $D_{A(P}$ denote the principal submatrix of D_A corresponding to the boundary vertices in P. Furthermore, let $\mathcal{T}(S)$ denote the set of spanning trees of Γ (or the associated network $\mathcal{P}$) rooted in $V \setminus S$.

We begin with the relation between Λ and L from the previous proposition. Note that the identity matrix makes no difference since P and Q are disjoint.

$$\begin{aligned}
\det(\Lambda_{PQ}) &= \det((D_A(I - L))_{PQ}) \\
&= \det(-D_{A(P)} L_{PQ}) \\
&= (-1)^k \det(D_{A(P)}) \det(L_{PQ})
\end{aligned}$$

We now expand $\det(D_{A(P)})$ in terms of its definition from the Kirchhoff matrix and $\det(L_{PQ})$ by the determinant-connection formula.

$$\begin{aligned}
\det(\Lambda_{PQ}) &= (-1)^k \left(\prod_{i \in P} \sum_{j \neq i} \gamma_{ij} \right) \left(\sum_{\tau \in S_k} \operatorname{sgn}(\tau) P(A_\tau | T) \right) \\
&= (-1)^k \left(\prod_{i \in P} \sum_{j \neq i} \gamma_{ij} \right) \left(\sum_{\tau \in S_k} \operatorname{sgn}(\tau) \sum_{\alpha : \tau_\alpha = \tau} P(\alpha | T) \right) \\
&= (-1)^k \left(\prod_{i \in P} \sum_{j \neq i} \gamma_{ij} \right) \left(\sum_{\tau \in S_k} \operatorname{sgn}(\tau) \sum_{\alpha : \tau_\alpha = \tau} \frac{P(\alpha) P(T | \alpha)}{P(T)} \right) \\
&= (-1)^k \left(\prod_{i \in P} \sum_{j \neq i} \gamma_{ij} \right) \left(\sum_{\tau \in S_k} \operatorname{sgn}(\tau) \sum_{\alpha : \tau_\alpha = \tau} \frac{\left(\prod_{ij \in \alpha} P_{ij} \right) \left(\sum_{t \in \mathcal{T}(\operatorname{int} V \setminus \alpha)} \prod_{ij \in t} P_{ij} \right)}{\left(\sum_{t \in \mathcal{T}(\operatorname{int} V)} \prod_{ij \in t} P_{ij} \right)} \right)
\end{aligned}$$

In order to turn the probabilities into conductivities, we distribute the conductances from the front and introduce matching terms corresponding to the products of the sums of conductances leaving each interior vertex into the numerator and denominator.

$$\det(\Lambda_{PQ}) = (-1)^k \sum_{\tau \in S_k} \operatorname{sgn}(\tau) \sum_{\alpha : \tau_\alpha = \tau} \frac{\left(\prod_{ij \in \alpha} P_{ij} \sum_{j \neq i} \gamma_{ij}\right)\left(\sum_{t \in \mathcal{T}(\operatorname{int} V \setminus \alpha)} \prod_{ij \in t} P_{ij} \sum_{j \neq i} \gamma_{ij}\right)}{\sum_{t \in \mathcal{T}(\operatorname{int} V)} \prod_{ij \in t} P_{ij} \sum_{j \neq i} \gamma_{ij}}$$

$$= (-1)^k \sum_{\tau \in S_k} \operatorname{sgn}(\tau) \sum_{\alpha : \tau_\alpha = \tau} \frac{\left(\prod_{ij \in \alpha} \gamma_{ij}\right)\left(\sum_{t \in \mathcal{T}(\operatorname{int} V \setminus \alpha)} \prod_{ij \in t} \gamma_{ij}\right)}{\sum_{t \in \mathcal{T}(\operatorname{int} V)} \prod_{ij \in t} \gamma_{ij}}$$

Finally, we apply Lewandowski's theorem and its corollary to relate the tree terms to determinants of submatrices of K.

$$\det(\Lambda_{PQ}) = (-1)^k \sum_{\tau \in S_k} \operatorname{sgn}(\tau) \sum_{\alpha : \tau_\alpha = \tau} \frac{D_\alpha \prod_{e \in \alpha} \gamma(e)}{\det(C)}$$

From here, multiplying through by $\det(C)$ gives us the desired form. $\square$

5. Recovering Boundary Edges and Spikes

A major problem in the study of electrical networks is the recovery of a network from its response matrix Λ, usually along with some geometric information. That is, under what circumstances can we derive information about K from Λ, and how can we do so? We will now examine the inverse problem for our input-output random walk networks. It is natural, given the close relation between K and P and between Λ and L, to suspect that we may be able to recover P from L under certain conditions and in a similar fashion.

Before attempting to recover any random walk networks, we will build up machinery similar to that used by Curtis and Morrow [1] to approach the inverse problem for electrical networks.

Definition 5.1 (Well-Behaved)**.** An input-output random walk network is said to be well-behaved if it has that property that, for any equally sized subsets P and Q of input and output states, there exists a k-connection from P to Q if and only if $\det(L_{PQ}) \neq 0$.

We say an edge in a random walk network is a boundary edge if it connects an input state directly to an output state. If the above condition is satisfied, we are able to calculate the probability associated with such an edge.

Theorem 5.2 (Boundary Edge Formula)**.** *Suppose $\mathcal{P}$ is a well-behaved input-output random walk network with a boundary edge between input state p and output state q. Let $P = (p_1, \ldots, p_k)$ and $Q = (q_1, \ldots, q_k)$ be sets of input and output vertices, respectively, which do not contain p or q, and let $P' = (p, p_1, \ldots, p_k)$ and $Q' = (q, q_1, \ldots, q_k)$. Suppose that P and Q are k-connected and that P' and Q' are (k+1)-connected, but that removing the edge from p to q breaks the connection from P' to Q'. Then*

$$P_{pq} = \frac{\det(L_{P'Q'})}{\det(L_{PQ})} = L_{pq} - L_{pQ} L_{PQ}^{-1} L_{Pq}.$$

Proof. For equally sized sets of input and output states P and Q, define the square matrix M_{PQ} by

$$M_{PQ} = \begin{pmatrix} S_{PQ} & T_P \\ R_Q & Q - I \end{pmatrix}$$

and note that L_{PQ} is given by the Schur complement of $(Q - I)$ in M_{PQ}. That is,

$$L_{PQ} = S_{PQ} - T_P (Q - I)^{-1} R_Q.$$

From the properties of the Schur complement, we have

$$\det(L_{PQ}) = \frac{\det(M_{PQ})}{\det(Q-I)} \implies \det(L_{PQ}) = 0 \text{ if and only if } \det(M_{PQ}) = 0.$$

Now consider $\det(M_{P'Q'})$ as a linear function $F(z)$ of the first row z of $M_{P'Q'}$. We can write $z = x + y$, where $x = (P_{pq}, 0, \ldots, 0)$ and $y = (0, a_1, \ldots, a_k)$. We know that $F(y) = 0$ since removing the boundary edge pq breaks the (k+1)-connection. We now have

$$\begin{aligned}
\det(M_{P'Q'}) &= F(z) \\
&= F(x+y) \\
&= F(x) + F(y) \\
&= F(x) \\
&= P_{pq}\det(M_{PQ}) \\
\det(L_{P'Q'})\det(Q-I) &= P_{pq}\det(L_{PQ})\det(Q-I) \\
\det(L_{P'Q'}) &= P_{pq}\det(L_{PQ}).
\end{aligned}$$

We recognize L_{PQ} as a principal submatrix of $L_{P'Q'}$ and use the Schur complement to rewrite our equation in the desired form:

$$P_{pq} = \frac{\det(L_{P'Q'})}{\det(L_{PQ})} = L_{pq} - L_{pQ}L_{PQ}^{-1}L_{Pq}.$$

□

We say an edge in a random walk network is a boundary spike if it connects an interior node to an output node and is the only edge leading into that particular output state. Under certain conditions, we are able to solve for the probability associated with such an edge directly from L.

Theorem 5.3 (Boundary Spike Formula). *Suppose $\mathcal{P}$ is a well-behaved input-output random walk network with a boundary spike from interior state r to output state q. Let $P = (p_1, \ldots, p_k)$ and $Q = (q_1, \ldots, q_k)$ be sets of input and output vertices, respectively, where Q does not contain q. Suppose that there is a k-connection from P to Q and that any such k-connection has the property that r is not used in the k-connection and any edge leaving r results in a valid spanning tree whenever the edges in the k-connection are already fixed. Finally, suppose there exists an input state p such that any spanning tree necessarily contains a path from p to r. Letting $P' = (p, p_1, \ldots, p_k)$ and $Q' = (q, q_1, \ldots, q_k)$, then*

$$P_{rq} = \frac{\det(L_{P'Q'})}{\det(L_{PQ})} = L_{pq} - L_{pQ}L_{PQ}^{-1}L_{Pq}.$$

Proof. First, note that $p \notin P$, or else we would contradict the assumption of independence, since a k-connection from P to Q would pass through r and necessarily avoid q.

Note that any (k+1) connection between P' and Q' must connect p to q through r, since paths connecting to either one must both use r. Letting α denote (k+1)-connections from P' to Q' and letting β denote k-connections from P to Q,

$$\begin{aligned}
\det(\Lambda_{P'Q'}) &= \sum_{\tau \in S_{k+1}} \operatorname{sgn}(\tau) \sum_{\alpha:\tau_\alpha = \tau} P(\alpha|T) \\
&= \sum_{\tau \in S_{k+1}:\tau(1)=1} \operatorname{sgn}(\tau) \sum_{\alpha:\tau_\alpha=\tau} P(\alpha|T) \\
&= \sum_{\tau' \in S_k} \operatorname{sgn}(\tau') \sum_{\beta:\tau_\beta=\tau'} P_{rq}P(\beta|T) \\
&= P_{rq} \sum_{\tau' \in S_k} \operatorname{sgn}(\tau') \sum_{\beta:\tau_\beta=\tau'} P(\beta|T) \\
&= P_{rq}\det(\Lambda_{PQ})
\end{aligned}$$

Again, we recognize L_{PQ} as a principal submatrix of $L_{P'Q'}$ and use the Schur complement to rewrite our equation in the desired form:

$$P_{rq} = \frac{\det(L_{P'Q'})}{\det(L_{PQ})} = L_{pq} - L_{pQ} L_{PQ}^{-1} L_{Pq}.$$

□

This boundary spike formula can actually be extended to recover any edge, assuming we can find two k-connections that differ only by its inclusion. Unfortunately, applying this formula requires intuition for the particular network and oftentimes a clever choice of connections. This limitation, along with the current lack of any way to "contract" a boundary spike (assuming this can even be done in a way analogous with an electrical network) makes recoverability difficult in general.

6. Transforming Random Walk Networks

In this section, we introduce a generalized Schur complement to deal with non-square matrices. Its properties are virtually identical to its counterpart in the square case.

Definition 6.1. Given an $(n + d) \times (m + d)$ matrix M written in block form as

$$M = \begin{pmatrix} A & B \\ C & D \end{pmatrix},$$

where D is a $d \times d$ invertible matrix, we say the $n \times m$ matrix $M/D = A - BD^{-1}C$ is the generalized Schur complement of D in M.

Proposition 6.2 (Determinants of Submatrices of Generalized Schur Complements)**.** *Suppose M is an $(n + d) \times (m + d)$ matrix written as above, with D invertible. If $P = (p_1, \ldots, p_k)$ and $Q = (q_1, \ldots, q_k)$ are ordered sets of indices, then*

$$\det (M/D)_{PQ} = \frac{\det M_{PQ}}{\det D},$$

where $(M/D)_{PQ}$ is the $k \times k$ matrix whose ijth entry is $(M/D)_{p_i q_j}$ and M_{PQ} is the $(k + d) \times (k + d)$ matrix given by

$$M_{PQ} = \begin{pmatrix} A_{PQ} & B_P \\ C_Q & D \end{pmatrix}.$$

The ijth entry of A_{PQ} is $A_{p_i q_i}$, the ith row of B_P is the p_ith row of B, and the jth column of C_Q is the q_jth column of C.

Proof. By our construction, $(M/D)_{PQ}$ is exactly the ordinary Schur complement of D in M_{PQ}. The determinant formula is a well-known property of the Schur complement. □

Given an input-output random walk network, we have at times found it convenient to describe it by the matrix $M = \begin{pmatrix} S & T \\ R & Q - I \end{pmatrix}$, which we will now call the characteristic matrix. From this, we have that the probability map L is given by the generalized Schur complement of $Q - I$ in M.

Sometimes, we may wish to remove a particular interior state from a vertex in such a way that does not change L and preserves connections in the network. Using terminology form the electrical network case, we call this a Star-K transform.

Definition 6.3 (Star-K Transform)**.** Suppose we have a network $\mathcal{P}$ with some interior state k. Then, for any (i, j) with neither i nor j equal to k, we define

$$p'_{ij} = p_{ij} + p_{ik} \frac{p_{kj}}{1 - p_{kk}}.$$

Then the set of all such p'_{ij} define transition probabilities for a response-equivalent network $\mathcal{P}'$ with state k removed, which we call the Star-K transform of k in $\mathcal{P}$.

Note that a Star-K transform may (and often will) introduce new edges to the network. In many cases, performing a Star-K transform will introduce self-loops, even if there were none originally, or cause a planar graph to become non-planar.

Proposition 6.4. *Suppose without loss of generality that k is the last interior vertex in some network $\mathcal{P}$ with characteristic matrix M. Then the characteristic matrix M' for the Star-K transform of k in $\mathcal{P}$ is given by the generalized Schur complement of $(Q-I)_{kk}$ in M.*

Proof. This follows immediately from the definitions of M and of the Star-K transform. □

Corollary 6.5. *The response map L for an input-output random walk network $\mathcal{P}$ is equivalent to the characteristic matrix M' for the network $\mathcal{P}'$ that results when we iteratively take the Star-K of each interior state in $\mathcal{P}$.*

Proof. Suppose M is the characteristic matrix for $\mathcal{P}$. Then we know that L is exactly the Schur complement of $Q-I$ in M. Without loss of generality, we can assume that we take the Star-K transform of each interior state in reverse order. By the proposition, this is equivalent to repeatedly taking the Schur complement of the last entry in M, which is guaranteed to be non-zero since every vertex at each stage must have some path to the output. Since this operation is well-defined, we know from properties of the Schur complement that the procedure used to find M' is equivalent to that used to find L, so they are equivalent. □

We also want a formula for removing a boundary edge, which we hope will be useful in recovering networks. Of course, we assume that there is at least one other edge leaving the node with this boundary edge, or else we might as well remove this input state altogether.

Proposition 6.6 (Removing a Boundary Edge)**.** *Suppose we have a network $\mathcal{P}$ with response map L and a boundary edge pq that we wish to remove. Removing this edge and scaling all others by $\frac{1}{1-P_{pq}}$ will yield a new network $\mathcal{P}'$ with response matrix L', where $L'_{pi} = L_{pi}\frac{1-\delta_{iq}P_{pq}}{1-P_{pq}}$ and all other entries are unchanged.*

Proof. Removing the edge is equivalent to setting S_{pq} to 0 and scaling all other entries in the *pth* rows of S and T by $\frac{1}{1-P_{pq}}$. The formula for L' follows from the response map formula given our new S and T. □

This formula for removing a boundary edge is rather nice in the fact that it does not significantly change the geometry of the network's underlying graph. If recovery of our probabilistic networks is similar to that of electrical networks, this formula may play an important role. Curtis and Morrow present an algorithm for recovering critical circular planar networks that involves finding a boundary edge or boundary spike, removing the edge or contracting the spike, and then iteratively repeating this procedure on the resulting network until we are done [1].

7. Some Thoughts on Recoverability

As with the electrical network, we wish to solve the inverse problem of recovering a network ($\mathcal{P}$, fully described by the transition matrix P) from its boundary data (L). We imagine that we know the underlying graph of the network (without edge weights) along with the matrix L, which is equivalent to knowing the dimensions and zero entries of the transition matrix P.

Definition 7.1 (Recoverable)**.** We say a random walk network with transition matrix P is recoverable if we can uniquely determine P with knowledge of only the boundary map L and of the dimensions and zero entries of P. That is, there does not exist some other transition matrix $P' \neq P$ that yields L such that $P'_{ij} = 0$ if and only if $P_{ij} = 0$.

We will consider the recoverability of two special types of networks that can be reduced to previously explored cases. Though we fall short of a full characterization of recoverable and non-recoverable networks, we hope that they may still shed light on recoverability in the general case.

Definition 7.2. We say an input-output random walk network $\mathcal{P}$ is simple if it satisfies the following:

(1) for each interior state j, there exists a unique input state i such that $P_{ij} = 1$; and
(2) there are no other input states except those required by the first condition.

Note that we have implicitly assumed that the set of interior states in a simple network is non-empty. Otherwise, we would have no input states and the network would be invalid.

Proposition 7.3. *A simple network is recoverable if and only if it is possible to recover R and Q from B.*

Proof. Without loss of generality, we assume that input vertices and their associated interior vertices have the same indices in their respective orderings. Then we know that S is the all zeros matrix and T is the identity matrix. This gives us that $L = S + TB = 0 + IB = B$. Since $L = B$ and S and T are already known, the problem of recovering P from L reduces to that of recovering R and Q from B. $\square$

Using B to recover R and Q has already been studied by DeVries, who considered networks without special input states, but we have defined our simple networks in such a way that his results can be extended trivially via the previous proposition. He presents a conjecture, due to Ryan Card, which would provide a full characterization of recoverable simple networks, as well as a proof of one direction [2].

Conjecture 7.4 (Card's Conjecture)**.** *A simple random walk network is recoverable if and only if all the edges leaving any interior node can be simultaneously extended to vertex-disjoint paths to output nodes.*

Theorem 7.5. *A simple random walk network is not recoverable if there exists an interior node such that the edges leaving that node can not be simultaneously extended to vertex-disjoint paths to output nodes.*

A proof of the opposite direction of Card's conjecture would be a great achievement, as it would give us a simple geometric test for recoverability of a non-trivial class of random walk networks.

We again consider random walks defined on electrical networks. Using Schroder's arguments [7], it becomes possible to relate recoverability of electrical networks to recoverability of their associated random walk networks.

Theorem 7.6. *Suppose we have an electrical network Γ and its associated random walk network $\mathcal{P}$. If Γ is recoverable, then $\mathcal{P}$ is recoverable.*

Proof. Suppose that the response map L for $\mathcal{P}$ is known. We employ a method demonstrated by Schroder to find the electrical network's response matrix Λ (up to a positive scalar multiple) from knowledge of L [7]. Once we know Λ, the proof is nearly complete.

We know from earlier that $\Lambda = D(I - L)$, where D is the diagonal matrix corresponding to the submatrix of the Kirchhoff matrix K with boundary connection data. However, without K, it is impossible for us to determine D. Fortunately, from Curtis and Morrow, we know that Λ is symmetric, its diagonal entries are positive, its off-diagonal entries are nonpositive, and its row (and column) sums are zero [1].

Since the matrix $I - L$ already has entries with the right signs and row sums zero, we need only find a diagonal matrix D' with positive entries such that $D'(I - L)$ is symmetric. Any such matrix must be unique up to a positive scalar multiple, and we know that at least one (D) exists, so we choose a D' and note that $D' = cD$ for some unknown $c > 0$.

With D' in hand, we can find Λ up to the same scalar multiple: $c\Lambda = cD(I - L) = D'(I - L)$. Since Γ is recoverable, this allows us to determine K up to a scalar multiple; that is, we can find cK. Note that knowledge of K uniquely determines P and that P only depends on the ratios of entries of K. Therefore, from cK we can find P, and thus $\mathcal{P}$ is recoverable. $\square$

Although the constraints on a random walk network are rather strong for it to have an associated electrical network, this theorem is extremely useful in that electrical networks are well-understood. In fact, Curtis and Morrow have developed a simple algorithm for recovering critical, circular planar electrical networks [1], so an algorithm for recovering their associated random walk networks requires only slight modifications.

It would be tempting to say that an electrical network is recoverable if and only if its associated random walk network is recoverable, but this does not hold true in general. It is not difficult, surprisingly, to construct networks which prove this statement false. For instance, consider an electrical network which consists of just two boundary vertices connected in series with either one or two interior vertices in between.

Though edges in series are always impossible to recover in the electrical case, it is trivial to recover the associated random walk network. However, if we insert any more than two interior vertices in this series connection, it becomes unrecoverable, as we will later see. We will now examine certain features that would make a random walk network unrecoverable.

Theorem 7.7. *Suppose that a random walk network $\mathcal{P}$ contains at least one interior state with a self-loop. Then $\mathcal{P}$ is not recoverable.*

Proof. We assume, without loss of generality, that the last interior state, k, has a self-loop. We consider the characteristic matrix M for $\mathcal{P}$ and the characteristic matrix M' for the network $\mathcal{P}'$ that we form by removing the self-loop at k and scaling the remaining transition probabilities from k by $\frac{1}{1-p_{kk}}$. We will show that the response matrix L for $\mathcal{P}$ is equal to the response matrix L' for $\mathcal{P}'$.

We established earlier that we may find L from M by iteratively taking the Schur complement of its last entry, and the same applies. to L' and M'. Since M and M' differ only in the last row, it is sufficient to show that the Schur complement of $p_{kk} - 1$ in M is equal to the Schur complement of the final -1 in M'. Taking the Schur complement,

$$(M/(p_{kk}-1))_{ij} = M_{ij} - M_{ik}\frac{1}{p_{kk}-1}M_{kj} = M'_{ij} - M'_{ik}\frac{1}{-1}M'_{kj} = (M'/-1)_{ij}.$$

M' is independent of the value of p_{kk}, depending only on the ratios of all other transition probabilities from k. Thus we can form infinitely many response-equivalent networks with the same geometry as $\mathcal{P}$ by choosing a value for p_{kk} from $(0, 1)$ and scaling other probabilities accordingly, so $\mathcal{P}$ is not recoverable. $\square$

8. Acknowledgements

I would like to thank Jim Morrow and Will Johnson for their help in understanding the determinant-connection formula and related topics for electrical networks, which gave my initial motivation for studying probabilistic networks. I am also grateful to Eric Nitardy and Zach Lindsey for several interesting discussions about probability; they invariably provided me with new problems to explore and gave insight into how to solve my current ones.

References

[1] Edward B. Curtis and James A. Morrow, *Inverse Problems on Electrical Networks*, World Scientific, New Jersey, 2000.
[2] Timothy DeVries, *Recoverability of Random Walk Networks*, UW Math REU (2003).
[3] Peter G. Doyle and J. Laurie Snell, *Random Walks and Electrical Networks*, Mathematical Association of America, Washington, D.C., 2000.
[4] Sergey Fomin, *Loop-Erased Walks and Total Positivity*, Transactions of the AMS (2001).
[5] Matthew J. Lewandowski, *Determinant of a Principle Proper Submatrix of the Kirchhoff Matrix*, UW Math REU (2008).
[6] Russell Lyons and Yuval Peres, *Probability on Trees and Networks*, Cambridge University Press, Cambridge, 2011.
[7] Konrad Schroder, *Interpretation of Electrical Networks as Probability Networks (and vice versa)*, UW Math REU (1993).