> ἀλλὰ τί ἦ μοι ταῦτα περὶ δρῦν ἢ περὶ πέτρην;
> But what is my business round tree or rock?
>
> Hesiod, *Theogony* (trans. M. L. West)

The main reference for this talk is the book *Probability on Trees and Networks*, by Lyons and Peres.

1 Generating Uniform Spanning Trees

In this talk, all graphs will be finite. Recall that a tree is a connected graph containing no cycles, and that a spanning tree of a graph G is a subgraph that is both a tree and which contains all the vertices of G. Also recall that a walk on a graph is a sequence of vertices with the property that all pairs of subsequent vertices are adjacent in the graph.

Ultimately, the goal of this talk is to understand what we can say about a uniformly random spanning tree of a given graph G (this makes sense because G is finite, so it has only finitely many spanning trees). To get there, we will first see a beautiful method for sampling a uniform spanning tree; note that doing this is not obviously easy—even counting the number of spanning trees is, a priori, a difficult task.

For a walk $v_0, v_1, \ldots, v_n$ in a graph G, we define its *loop erasure* by deleting all cycles that the path traces out in the order they appear. More concretely, if the path contains no cycles, we do nothing; otherwise, there will be some minimal $1 \leq j \leq n$ for which $v_j = v_i$ for some $i < j$. Then we delete $v_i, v_{i+1}, \ldots, v_{j-1}$ from our walk and iterate this process.

Definition. A *rooted tree* is a tree T together with a specified vertex r (called the root). We orient all the edges of T towards r; in other words, every root other than r has exactly one outgoing edge, and r has none.

Definition (Wilson's Algorithm)**.** Given a connected graph G and a vertex r, we define a growing sequence of rooted trees T_i as follows:

1. Set $T_0 = \{r\}$
2. If T_{i-1} is a spanning tree of G, we stop. Otherwise, we pick an arbitrary vertex v not contained in T_{i-1} and start a simple random walk at v. Eventually this walk will hit a vertex in T_{i-1}, and let P be the loop erasure of this path from v to T_{i-1}. Then define $T_i = T_{i-1} \cup P$.

Observe that each step will take finite time a.s., since G is finite. Also observe that each T_i is a tree, since we are adding a pendent loopless path to the previous tree at each step. Also observe that since we only terminate once all vertices have been incorporated, we will get a (rooted) spanning tree at the end of the process. The astonishing thing is its distribution:

Theorem. *Wilson's Algorithm produces a uniformly random rooted spanning tree. In particular, if we forget the root, we get a uniformly random spanning tree of G.*

Remark. Wilson's Algorithm can be similarly defined for any weighted, directed graph, or even more generally, for any Markov chain. It turns out that in all these cases, it will produce the (appropriate analogue of) a uniform spanning tree. The same proof works, but to maintain notational simplicity, we will work with the unweighted, undirected case.

Proof. The key idea is to "pre-process" all the randomness that we might ever use. Specifically, for each non-root vertex v in the graph, form an infinite sequence of independent random variables $\{S_v^{(i)}\}_{i\geq 1}$, where each $S_v^{(i)}$ is uniformly distributed on the neighbors of v. The idea is the following: the first time we get to v and need to pick a neighbor to continue our simple random walk, we will use the neighbor given by $S_v^{(1)}$. The second time we get to v, we will use $S_v^{(2)}$, and so on. This will certainly be the same process as if we had performed the random choice the moment we needed it, since the process doesn't know "when we flipped the coins," as it were. Nevertheless, this is a very useful perspective on what's happening.

Now, we will define a new process, and prove that it is both equivalent to Wilson's Algorithm and that it produces a uniform spanning tree. Specifically, think of placing all the random variables $S_v^{(i)}$ in a stack below each vertex v. Then the top of the stack simply assigns, to each vertex v, one of its neighbors. This naturally has the structure of a directed graph: we draw a directed edge from v to its neighbor $S_v^{(1)}$.

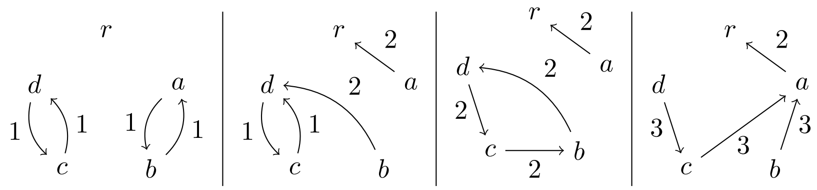
If this directed graph has no cycles, we stop the process. Otherwise, we arbitrarily pick one of the cycles and "pop" it from the stack. This means eliminating the top element of the stack for each vertex in this cycle and moving the rest of the stack upwards (i.e. replacing $S_v^{(i)}$ with $S_v^{(i+1)}$). This gives us a new directed graph, and we repeat this process.

Finally, we will record throughout which level of the stack each directed edge comes from. Recall that a directed edge represents a pair $(v, S_v^{(i)})$, so we simply mark this edge with the "color" i. Thus, in the first step, all edges have color 1, and after we pop the first cycle, all the edges in that cycle will have color 2 while everything else will still have color 1, and so on. Then by a *colored cycle*, we just mean a cycle in one of these directed graphs, together with the marking of which level in the stack each of its edges came from.

Example. Suppose that we have a have a graph with five vertices, called r, a, b, c, d, and suppose that the stacks look like this:

color	a	b	c	d	r
1	b	a	d	c	
2	r	d	b	c	
3	d	a	a	c	
$\vdots$			$\vdots$		

Then we have the following sequence of colored directed graphs, where we first choose to pop the ab cycle:



The following key lemma, which is completely deterministic, is the heart of the proof, and also explains why Wilson's Algorithm always works despite the huge amount of arbitrary choices that it allows us to make.

Lemma (Key lemma)**.** *In the above process, it doesn't matter what order we pop the cycles in. More formally, suppose each vertex comes with an (arbitrary) infinite stack, and we perform the above process. Then one of the following two cases occurs:*

1. *Any order of popping will never terminate (i.e. infinitely many cycles will be popped)*
2. *In any order of popping, the same set of colored cycles will be popped.*

Proof. Suppose that C is any colored cycle that can be popped at some point in some popping order. That means that there is a sequence of colored cycles $C_1, \ldots, C_k = C$ such that we can pop the cycles in that order. Let $C' \neq C_1$ be any other cycle that can be popped at the beginning. Then we will show that either $C' = C$ or else C can still be popped via a sequence that begins with C'. This suffices, since it implies that any alternative choice of popping sequence either preserves the infinite number of poppings or preserves the (finite) set of cycles to be popped.

If C' is vertex-disjoint from $C_1 \cup \cdots \cup C_k$, then popping it first will not affect any of the stacks in the sequence $C_1, \ldots, C_k$, so we will still be able to pop that sequence. Therefore, let $1 \leq m \leq k$ be the first index for which $C' \cap C_m \neq \emptyset$. Let x be some vertex shared by C' and C_m. Then since C' can be popped at the first stage, all its edges must have color 1. Since x was not in the sequence $C_1, \ldots, C_{m-1}$, and since C_m can be popped after all of those, the color on the edge leaving x in C_m must also be 1, since x was never popped. Thus, the successor of x in C_m is the same as its successor in C'. Applying this same argument to the successor of x, and then to its successor, and so on, implies that $C' = C_m$. But then C_m is vertex-disjoint from $C_1, \dots, C_{m-1}$, so either $C' = C$ or else we can pop these cycles in the order $C_m, C_1, \dots, C_{m-1}, C_{m+1}, \dots, C_k$. □

Going back to Wilson's Algorithm, we see that if we pre-process all the randomness, we do just have some method for popping the stacks in some order (namely loop-erasing the random walks). So Wilson's Algorithm will produce the same spanning tree distribution as any other method of popping the stacks, since we just proved that the specific method doesn't matter.

Since Wilson's Algorithm terminates in finite time a.s., we see that when we sample our stacks randomly, we will a.s. only need to pop finitely many cycles. Thus, we can think of our collection of stacks as a union of finitely many colored cycles O sitting on top of a spanning tree T, and then a bunch of junk beneath T (which we'll never see since we terminate the process once we hit T). Observe that if (O, T) is such a pair that can arise in our stacks, then (O, T') can also arise for any other spanning tree T': indeed, any spanning tree can live under the set of cycles. But this implies that the appearance of O is independent of the appearance of T, and any T is equally likely, so the distribution of the final spanning tree is uniform. □

2 Harmonic Functions

Definition. Given a graph $G = (V, E)$, a function $f : V \to \mathbb{R}$ is called *harmonic* at a vertex v if its value at v is the average of its values on the neighbors of v, i.e.

$$f(v) = \frac{1}{\deg(v)} \sum_{u: uv \in E} f(u)$$

or equivalently

$$\sum_{u: uv \in E} (f(u) - f(v)) = 0$$

We say f is harmonic on a set W of vertices if it's harmonic at w for all $v \in W$.

We won't need this interpretation, but f is harmonic if and only if it's in the kernel of the graph Laplacian, so this definition matches our usual notion of harmonicity.

Harmonic functions turn out to be extremely useful; unfortunately, they don't exist:

Lemma. *If G is a finite, connected graph, then the only functions harmonic on all of V are constants functions.*

Proof. The idea is that harmonic functions satisfy a maximal principle, very similar to the maximal principle we know for harmonic functions on $\mathbb{R}^n$.

Let $f : V \to \mathbb{R}$ be harmonic on all of V, and let $W \subseteq V$ be the set of vertices where f attains its maximum (f attains a maximum since V is finite). Then $W \neq \emptyset$, and if $W = V$, then f is constant. So suppose not; since G is connected, there is some $v \in W$ adjacent to some $u \notin W$. Thus, $f(u) < f(v)$, so the average of f on the neighbors of v will also be strictly less than $f(v)$, so f is not harmonic at v. □

Note that we can run the same argument with the minimum instead of the maximum, and we can thus find that a non-constant function must be non-harmonic at at least two vertices. This turns out to be the only restriction:

Lemma. *For any $A \subseteq V$ with $|A| \geq 2$ and any $f_0 : A \to \mathbb{R}$, there is a unique $f : V \to \mathbb{R}$ such that $f|_A = f_0$ and f is harmonic on $V \setminus A$.*

Note that this is a discrete analogue of the Dirichlet problem: we specify boundary conditions (on A), and can find a unique harmonic extension to the rest of the domain. Also note that the assumption $|A| \geq 2$ isn't strictly necessary—if $|A| = 1$, then the unique harmonic extension will just be the constant function agreeing with f_0.

Proof. We first prove uniqueness. Observe that if f, g are harmonic at v, then any linear combination of f and g is also harmonic at v. Therefore, suppose we had two harmonic extensions f, g of f_0. Then $f - g$ is also harmonic on $V \setminus A$, and $(f-g)|_A \equiv 0$. But by the previous lemma, we know that $f - g$ attains both its maximum and minimum in A, and thus $f - g \equiv 0$, so $f = g$.

Existence can be proved with fairly simple linear algebra, but it will follow more concretely from any one of the following examples of harmonic functions. □

Example. Fix a graph $G = (V, E)$, a subset $A \subseteq V$, and a function $f_0 : A \to \mathbb{R}$.

1. For any vertex $v \in V$, run a simple random walk starting at v. Let Y be the first vertex of A that this random walk hits, and define $f(v) = \mathbb{E}[f_0(Y)]$. Then we certainly have that $f|_A = f_0$, since $Y = v$ if we start in A. For harmonicity, let the random walk be $v = X_0, X_1, \ldots$ Then we have that

$$\begin{aligned} f(v) &= \mathbb{E}[f_0(Y)] \\ &= \sum_{a \in A} f_0(a) \Pr(Y = a) \\ &= \sum_{a \in A} f_0(a) \left[\sum_{u : uv \in E} \Pr(Y = a \mid X_1 = u) \Pr(X_1 = u) \right] \\ &= \frac{1}{\deg(v)} \sum_{u : uv \in E} \sum_{a \in A} f_0(a) \Pr(Y = a \mid X_1 = u) \\ &= \frac{1}{\deg(v)} \sum_{u : uv \in E} \mathbb{E}[f_0(Y \mid X_1 = u)] \\ &= \frac{1}{\deg(v)} \sum_{u : uv \in E} f(u) \end{aligned}$$

where we use the fact that the random walk is a Markov chain, so conditioning on $X_1 = u$ is the same as just starting the random walk at u.

2. Think of every edge of the graph as a 1 Ohm resistor, and impose a voltage of $f_0(a)$ for each $a \in A$. Let $f(v)$ be the voltage at v in equilibrium. Then for any $v \notin A$ and any u adjacent to v, we have that $f(u) - f(v)$ is the voltage difference along the resistor uv, which by Ohm's law is equal to the current flowing from u to v. By Kirkhoff's node law, the total current flowing at a vertex must be zero, i.e.

$$\sum_{u : uv \in E} (f(u) - f(v)) = 0$$

Thus, f is harmonic.

3. Suppose every edge of the graph is a spring of spring constant 1. Imagine attaching each vertex of A to the real line, at position $f_0(a)$. Then let the rest of the vertices find their equilibrium positions, and let these positions be f. By Hooke's law, the force an edge uv applies to v is $f(u) - f(v)$. The fact that the system is in equilibrium just means that the forces acting on every non-fixed vertex balance, which just means that

$$\sum_{u : uv \in E} (f(u) - f(v)) = 0$$

Thus, f is harmonic.

These examples are very important, and they also prove that harmonic extensions exist. For the second and third examples, this "proof" requires us to either believe or prove that the equilibrium state always exists. However, the first example is a complete proof, since we know that a simple random walk on a finite graph will hit any set A in finite time a.s.

The main reason these examples are important is that harmonic functions are unique, and thus, in some sense, all these examples are the same. We will, in particular, exploit the connection between random walks and currents repeatedly. To do so, we will first build up a bit the theory of graphs as electrical networks, and then use this to understand random walks, and then use those to understand uniform spanning trees.

3 Electrical Networks

From now on, we will adopt the following convenient convention: $\vec{E}$ will now represent the set of *oriented* edges of the graph. This means that every edge $\{u, v\}$ will appear twice in $\vec{E}$, once as uv and once as vu. For each oriented edge $e = uv$, let $\chi_e : \vec{E} \to \mathbb{R}$ be the *unit flow along* e, namely

$$\chi_e(e') = [1_{uv} - 1_{vu}](e') = \begin{cases} 1 & e' = uv \\ -1 & e' = vu \\ 0 & \text{otherwise} \end{cases}$$

Let H denote the vector space of all antisymmetric functions on $\vec{E}$, namely all $f : \vec{E} \to \mathbb{R}$ satisfying $f(vu) = -f(uv)$ for all edges uv. We make H a Hilbert space by putting on it the natural inner product

$$\langle f, g\rangle = \frac{1}{2}\sum_{e \in \vec{E}} f(e)g(e)$$

The $1/2$ is put in this formula just to make it so that if we pick only one orientation for each edge, then we will be summing exactly once over every edge. Observe that $\{\chi_e\}_{e \in E}$ form an orthonormal basis for H, where we only include one orientation of each edge in this basis.

Define the following two subspaces of H, called the *star space* and the *cycle space,* respectively:

$$S = \operatorname{span}\left\{ \sum_{u : uv \in E} \chi_{uv} \;\middle|\; v \in V \right\} \qquad C = \operatorname{span}\left\{ \sum_{i=1}^{n} \chi_{e_i} \;\middle|\; e_1, \ldots, e_n \text{ an oriented cycle} \right\}$$

Lemma. *$S = C^\perp$. In other words, $H = S \oplus C$ and $S \perp C$.*

Proof. This is essentially saying that the first cohomology of the graph is generated by its cycles. For a formal proof, first observe that certainly $S \perp C$; this is because the inner product of any star with any cycle will be zero if they're edge-disjoint, and will also be zero if they're not, since in that case the cycle must include two edges incident with the star vertex, so their contributions will cancel. So it remains to prove that $H = S \oplus C$, or equivalently that $(S \oplus C)^\perp = 0$. For that, we need to define the coboundary operator: given a function $F : V \to \mathbb{R}$, we define its coboundary $dF \in H$ by

$$(dF)(uv) = F(u) - F(v)$$

Then the key property is that a function $f \in H$ is orthogonal to C if and only if $f = dF$ for some function $F : V \to \mathbb{R}$. One direction is immediate: $dF \perp C$ since for any oriented cycle $e_1, \ldots, e_n$,

$$\left\langle dF, \sum_{i=1}^{n} \chi_{e_i} \right\rangle = \sum_{i=1}^{n} (F(v_i) - F(v_{i-1})) = 0$$

where $e_i = v_{i-1}v_i$. For the converse, if $f \perp C$, then we can "integrate" it to a function $F : V \to \mathbb{R}$ by declaring $F(v) = 0$ for some starting vertex v, then $F(u) = f(vu)$ for every neighbor u of v, and continuing in this fashion: any time we get to a new vertex, we declare F to take the value that makes it so that dF agrees with f on the edge just traversed. Then the fact that f is orthogonal to all cycles precisely means that all possible ways of making these choices are consistent, so we get a well-defined function F with $f = dF$.

Now, to prove that $(S \oplus C)^\perp = 0$, suppose $f \in (S \oplus C)^\perp$. By the above, we can find F such that $f = dF$. But then for any $v \in V$,

$$0 = \left\langle f, \sum_{u:uv\in E} \chi_{uv} \right\rangle = \sum_{u:uv\in E} (F(u) - F(v))$$

Thus, F is harmonic at every vertex, so it is constant. Therefore, $dF \equiv 0$, which is what we wanted to prove. □

Let P_S be the orthogonal projection operator onto the star space S, and define $I_e = P_S \chi_e$.

Lemma. *I_e is the unit current along e, i.e. it records the current flowing along every edge when we impose a current of 1 entering the tail of e and a current of 1 exiting its head. Equivalently, $I_e = dF$, where F is the unique function harmonic everywhere but the endpoints of e, taking value 0 at the head of e and normalized so that the inner product of dF with the star at the tail is 1.*

Proof. Let F be as in the statement. Then $dF \perp C$ by our previous proof, and thus $dF \in S$. Consider $\chi_e - dF$. We have that for any $v \in V$,

$$\left\langle \chi_e - dF, \sum_{u:uv\in E} \chi_{uv} \right\rangle = \left\langle \chi_e, \sum_{u:uv\in E} \chi_{uv} \right\rangle - \left\langle dF, \sum_{u:uv\in E} \chi_{uv} \right\rangle$$

By our previous computation, the second term is zero unless v is a vertex where F is not harmonic, i.e. unless v is an endpoint of e. Similarly, the first term is zero unless v is one of the endpoints of e. Moreover, their signs will agree in both of these cases, and thus, by our normalization, this quantity will always be zero. In other words, $(\chi_e - dF) \perp S$. So $(\chi_e - dF) \in C$. This means that

$$\chi_e = dF + (\chi_e - dF)$$

is the orthogonal decomposition of χ_e as an element of S plus an element of C. Thus, $dF = P_S \chi_e = I_e$, as desired. □

Using this fact, we can compute the representation of the operator P_S as a matrix Y with respect to the orthogonal basis $\{\chi_e\}_{e\in E}$, where again we pick only one orientation for each edge. Specifically, the (e, f) entry in this matrix is exactly $\langle P_S \chi_e, \chi_f \rangle$, which we can compute as

$$Y(e, f) = \langle P_S \chi_e, \chi_f \rangle = \langle I_e, \chi_f \rangle = I_e(f)$$

Since P_S is a projection operator, it's self-adjoint, so $Y(e, f)$ is also equal to $I_f(e)$. The matrix Y is called the *transfer current matrix*, since it measures how much current is transferred between edges: when we impose a total current of 1 Ampere between the endpoints of an edge e, how much current flows in another edge f?

4 Back to Spanning Trees

The astonishing and important theorem that ties this whole talk together is the following:

Theorem (Burton–Pemantle)**.** *Let T denote a uniformly random spanning tree of a graph G. Then for any collection of edges $e_1, \ldots, e_k$, we have that*

$$\Pr(e_1 \in T, \ldots, e_k \in T) = \det[Y(e_i, e_j)]_{1\le i,j\le k}$$

Remark. This says that when we consider the uniform spanning tree as a point process on E, then it is a determinantal point process, with kernel Y. Note that in particular,

$$\Pr(e \in T, f \in T) = Y(e, e)Y(f, f) - Y(e, f)^2 = \Pr(e \in T)\Pr(f \in T) - Y(e, f)^2$$

which implies that there is an anticorrelation in the appearance of points in this point process. This is an instance of the more general phenomenon that points in determinantal point process are “repelled”.

The proof of the Burton–Pemantle theorem proceeds by induction on k. The base case is an important result in its own right:

Theorem (Kirkhoff's Effective Resistance Formula)**.**

$$\Pr(e \in T) = Y(e, e) = I_e(e)$$

Proof. Let $e = uv$. To sample T, we run Wilson's Algorithm, setting our root to be v and taking u as our first vertex. We now run a loop-erased random walk starting at u, and stop once it hits v. If the final step goes across e, then e will be added to T by Wilson's Algorithm, whereas if the final step hits v via some other edge, then e will never be added to T. Thus,

$$\Pr(e \in T) = \Pr(\text{first hit of } v \text{ is along } e \text{ in a SRW starting at } u)$$

Note that since we only care about the final step, we can replace the loop-erased random walk by a simple random walk.

Now, for any vertex w, let $F(w)$ be the expected number of times a simple random walk starting at w hits u before its first hit of v. Then $F(v) = 0$, and F is harmonic on $V \setminus \{u, v\}$ by conditioning on the first step of the random walk. Additionally, again by conditioning on the first step of the random walk, we see that

$$F(u) = 1 + \frac{1}{\deg(u)} \sum_{w:uw \in E} F(w)$$

Therefore, thinking of F as a voltage, the total current flowing out of u is

$$\sum_{w:uw \in E} (F(u) - F(w)) = \deg(u)$$

Therefore, by renormalizing so as to get the definition of I_e, we see that

$$I_e(e) = \frac{F(u) - F(v)}{\deg(u)} = \frac{F(u)}{\deg(u)}$$

On the other hand, observe that the probability that we first hit v while traversing u is the same as the sum over all visits to u (before the first visit to v) of the probability that at that visit, e is the edge we next choose to follow. By linearity of expectation, this is precisely $F(u)/\deg(u)$, as desired. □

Proof of the Burton–Pemantle Theorem. First, suppose that there is some cycle among the edges $e_1, \ldots, e_k$. Then certainly $\Pr(e_1 \in T, \ldots, e_k \in T) = 0$, since no spanning tree may contain a cycle. So we first need to prove that in this case, $\det[Y(e_i, e_j)] = 0$. Write the indicator of this cycle as $\sum_i a_i \chi_{e_i}$, where $a_i \in \{-1, 0, 1\}$ (we allow 0 since some edges aren't in the cycle, and need to allow signs to deal with orientations of the edges). Then we have that for any j,

$$\sum_{i=1}^{k} a_i Y(e_i, e_j) = \sum_{i=1}^{k} a_i I_{e_j}(e_i) = \sum_{\text{an oriented cycle}} I_{e_j} = 0$$

since the sum of any current over an oriented cycle is 0. Therefore, we've found a linear combination of the columns of $[Y(e_i, e_j)]$ that sums to zero, and thus $\det[Y(e_i, e_j)] = 0$ in this case, which is what we needed. So from now on, assume that the edges $e_1, \ldots, e_k$ contain no cycles.

The proof now proceeds by induction on k, with the base case being Kirkhoff's theorem. For the inductive case, we write

$$\Pr(e_1 \in T, \ldots, e_k \in T) = \Pr(e_k \in T \mid e_1, \ldots, e_{k-1} \in T) \Pr(e_1, \ldots, e_{k-1} \in T)$$

Let Y_k be the matrix $[Y(e_i, e_j)]_{1\leq i,j\leq k}$ and Y_{k-1} the submatrix $[Y(e_i, e_j)]_{1\leq i,j\leq k-1}$. Then by the inductive hypothesis, we want to prove that

$$\det Y_k \overset{?}{=} \Pr(e_k \in T \mid e_1 \in T, \dots, e_{k-1} \in T) \det Y_{k-1}$$

Consider the graph $G/\{e_1, \dots, e_{k-1}\}$ gotten by contracting all the edges $e_1, \dots, e_{k-1}$ (i.e. identifying their two endpoints as a single vertex). Then the key observation is that since there is no cycle among the edges $e_1, \dots, e_{k-1}$, there is a bijection between spanning trees in $G/\{e_1, \dots, e_{k-1}\}$ and spanning trees in G that contain $e_1, \dots, e_{k-1}$. Therefore, conditioning on $e_1, \dots, e_{k-1} \in T$ is the same as moving to the uniform spanning tree in $G/\{e_1, \dots, e_{k-1}\}$, i.e.

$$\Pr(e_k \in T \mid e_1, \dots, e_{k-1} \in T) = \Pr(e_k \in T_{G/\{e_1,\dots,e_{k-1}\}})$$

where $T_{G/\{e_1,\dots,e_{k-1}\}}$ is a uniform spanning tree in $G/\{e_1, \dots, e_{k-1}\}$.

Finally, we need the following lemma:

Lemma. *Let*

$$Z = (\mathrm{span}\{I_{e_1}, \dots, I_{e_{k-1}}\})^\perp \subseteq H$$

and let P_Z be the orthogonal projection onto Z. Let $\widehat{I_{e_1}}$ denote the unit current across e_1 in the graph $G/\{e_1, \dots, e_{k-1}\}$. Then

$$\widehat{I_{e_k}} = P_Z I_{e_k}$$

Proof. The key property is that since there are no cycles among $e_1, \dots, e_k$, e_k will not be contracted to a loop in $G/\{e_1, \dots, e_{k-1}\}$. However, the actual proof is a bit technical and not particularly interesting (it's little more than linear algebra), so we'll skip it. □

With this lemma, we know that

$$\Pr(e_k \in T \mid e_1, \dots, e_{k-1} \in T) = \Pr(e_k \in T_{G/\{e_1,\dots,e_{k-1}\}}) = \widehat{I_{e_k}}(e_k)$$

by Kirkhoff's Theorem. Additionally,

$$\widehat{I_{e_k}} = P_Z I_{e_k} = I_{e_k} - \sum_{i=1}^{k-1} a_i I_{e_i}$$

for some constants a_i, by the definition of Z. Let $\widehat{Y}$ be the matrix gotten from Y_k by subtracting these same multiples of each row from the final row. In other words $\widehat{Y}$ has the same entries as Y_k in the first $k-1$ rows, whereas its (k, j) entry is

$$\widehat{Y}(k, j) = Y(k, j) - \sum_{i=1}^{k-1} a_i Y(i, j) = I_{e_k}(e_j) - \sum_{i=1}^{k-1} a_i I_{e_i}(e_j) = \widehat{I_{e_k}}(e_j)$$

However, for all $1 \leq j \leq k-1$, the edge e_j was contracted to a loop in $G/\{e_1, \dots, e_{k-1}\}$, so no current flows through that edge, i.e. $\widehat{I_{e_k}}(e_j) = 0$. So the final row of $\widehat{Y}$ has zeros everywhere other than in the final position, where it has $\widehat{I_{e_k}}(e_k)$. So expanding $\det Y_k = \det \widehat{Y}$ along the final row, we find that

$$\det Y_k = \det \widehat{Y} = \widehat{I_{e_k}}(e_k) \det Y_{k-1}$$

as desired. □