MATH 3120 – Kirchhoff's Matrix Tree Theorem

Ivan Zhao
ivzhao@sas.upenn.edu

June 10, 2023

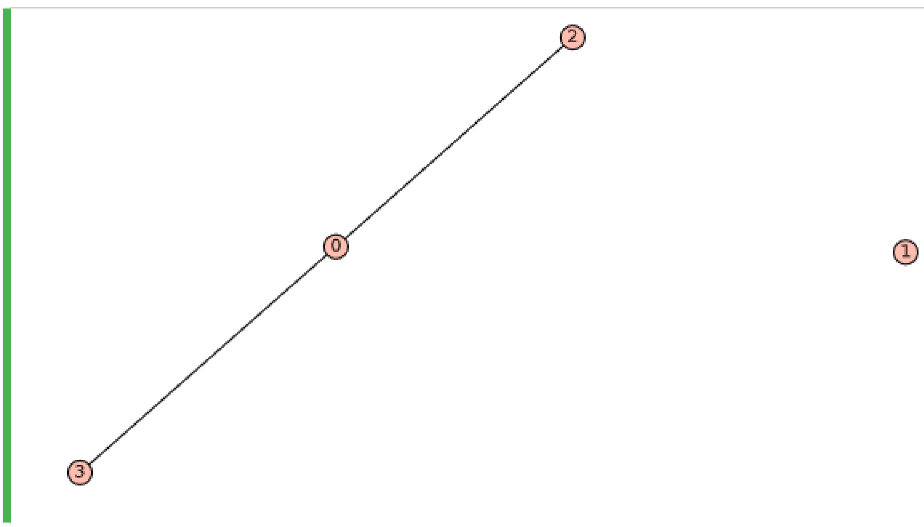
Let G = (V,E) be an unweighted and undirected graph where V is the set of vertices and E are the set of edges. We denote the number of vertices as n and the number of edges as m.

The Laplacian matrix, L_G is a n x n matrix, where row i represents a vertex i, column j represents a vertex j, and:

$$L_{i,j} = \begin{cases} -1 & \text{if } e_{i,j} \in \text{ E} \\ d_i & \text{if i = j} \\ 0 & \text{otherwise} \end{cases}$$

Note that d_i is defined as the degree of vertex i. In other words, this matrix has information about the connectivity of G. Specifically, along the diagonal is the degree of a vertex i, and $L_{i,j}$ tells us whether an edge exists between two vertices, $v_i \neq v_j$. Since both the rows and columns represent vertices, clearly the matrix for an unweighted graph is symmetric.

Let's see an example of the Laplacian matrix:
Consider the following graph:



The corresponding Laplacian matrix would be the following matrix:

$$\begin{bmatrix} 2 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix}$$

Theorem MBLT from *A First Course in Linear Algebra* by Robert Beezer, says that given a matrix we can find linear transformation. So let's try to characterize what the Laplacian matrix does to a vector v of vertices. Without loss of generality, consider v $\in \mathbb{R}^3$, where each vertex is mapped to a real number. Denote v as:

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

and let the Laplacian matrix, L_G be:

$$\begin{bmatrix} d_1 & e_{12} & e_{13} \\ e_{21} & d_2 & e_{23} \\ e_{31} & e_{32} & d_3 \end{bmatrix}$$

where d_k is the degree of vertex k, and e_{ij} is the -1 if the edge exists or 0 otherwise. Then:

$$L_G v = \begin{bmatrix} d_1 & e_{12} & e_{13} \\ e_{21} & d_2 & e_{23} \\ e_{31} & e_{32} & d_3 \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{bmatrix} d_1 x_1 + e_{12} x_2 + e_{13} x_3 \\ e_{21} x_1 + d_2 x_2 + e_{23} x_3 \\ e_{31} x_1 + e_{32} x_2 + d_3 x_3 \end{bmatrix} = \begin{bmatrix} d_1 (x_1 + \frac{e_{12} x_2 + e_{13} x_3}{d_1}) \\ d_2 (x_2 + \frac{e_{21} x_1 + e_{23} x_3}{d_2}) \\ d_3 (x_3 + \frac{e_{31} x_1 + e_{32} x_2}{d_3}) \end{bmatrix}$$

For each row i, the x_k terms in the fraction are the neighbors of x_i. This is because if x_k is not a neighbor, it will be zeroed out since $e_{ij} = 0$ if the edge does not exist. Then, there are exactly d_i number of x_k terms in the fraction. As a result, each row of $L_G v$ equals
($d_i(x_i$ - Average of the neighbors of x_i)). In other words:
$[L_G v]_i = [d_i * (x_i -$Average of neighbors of $x_i)]$ where i is the row. Notice that the average of neighbors of x_i is negative because $e_{ij} = -1$ if the edge exists.

We see that the relevant vector spaces are is V, the space of vertices, and $\mathbb{R}^n$, where n is the number of vertices.

Hermitian Square

Given graph G=(V,E), where V is the set of vertices and E is the set of edges, we can write the Laplacian matrix as the hermitian square of the incidence matrix.

Proof:
Recall that the incidence matrix is a m by n matrix, where m is the number of edges and n is the number of vertices, such that each row represents an edge and each column is a vertex. For the entries of an incidence matrix A:

$$A_{i,j} = \begin{cases} -1 & \text{if } e_{i,j} \in \text{ E} \\ 1 & \text{if } e_{j,i} \in \text{ E} \\ 0 & \text{otherwise} \end{cases}$$

We want to show that $A^T A$ is the Laplacian matrix. Each entry, $(A^T A)_{ij}$, of $A^T A$ is computed by $\sum_{l=1}^{k} A_{ki} A_{kj}$. Note that i and j represent vertices (columns), and k represents the an edge (a row). We have the following cases for $A_{ki} A_{kj}$:

Case 1: i = j
If vertex i is not involved in edge k, $A_{ki} = 0$ and $A_{kj} = 0$, so $A_{ki} A_{kj} = 0$.
If vertex i is not involved in edge k, then $A_{ki} A_{kj}$ is either (1)*(1) or (-1)*(-1). In either instance $A_{ki} A_{kj} = 1$.

Case 2: i $\neq$ j
If no edge exists between vertex i and vertex j, then $A_{ki} = 0$ and $A_{kj} = 0$, so $A_{ki} A_{kj} = 0$.
If there is an edge between vertex i and vertex j, then $A_{ki} A_{kj}$ is either (-1)*(1) or (1)*(-1). In either instance $A_{ki} A_{kj}$ = -1.

The following equation encapsulates this information:

$$A_{ki} A_{kj} = \begin{cases} 1 & \text{if } i = j \text{ and } e_{i,x} \in \text{ E for some vertex x} \\ -1 & \text{if } i \neq j \text{ and } e_{i,j} \in \text{ E} \\ 0 & \text{otherwise} \end{cases}$$

Now we can characterize $(A^T A)_{ij}$=$\sum_{l=1}^{k} A_{ki} A_{kj}$:

$$(A^T A)_{ij} = \begin{cases} \sum_{e_{i,x} \in E} 1 & i = j, e_{i,x} \in \text{ E for some vertex x} \\ -1 & i \neq j, e_{i,j} \text{ or } e_{j,i} \in \text{ E} \\ 0 & \text{otherwise} \end{cases}$$

If i = j, then $(A^T A)_{ij}$ is just the count of edges incident to i. In other words, it's the degree of vertex i. If $i \neq j$, then $(A^T A)_{ij}$ is the negative count of all edges between vertex i and j. Since there can only be one edge between vertex i and j in an undirected graph, this is just -1. If there are no edges involving i and j, then $(A^T A)_{ij}$ is just 0.

We see that these entries are exactly the same as the Laplacian. In both $A^T A$ and the Laplacian matrix, along the diagonal where i = j, is the degree of the vertex. Also in both matrices, when i $\neq$ j, the entry is -1 if there is an edge between i and j or 0 otherwise. Thus, we can conclude that the hermitian square of the incidence matrix is the same as the Laplacian matrix.

Inner Product

Lemma: L is a positive semi-definite matrix.

Proof:
We want to show that $x^T Lx \geq 0$. We can rewrite $x^T Lx$ as the following:

$$\begin{aligned} x^T Lx &= x^T(A^T A)x \qquad \text{where A is the incidence matrix} \\ &= (Ax)^T(Ax) \end{aligned}$$

Each row of the incidence matrix represents an edge. Thus, row k of Ax equals $(x_i - x_j)$ where the the edge (i,j) is e_k. Then we can rewrite $(Ax)^T(Ax)$ as follows:

$$(Ax)^T(Ax) = \sum_{i,j \in E}(x_i - x_j)^2$$

It must be true that $\sum_{i,j\in E}(x_i - x_j)^2 \geq 0$ since this is a summation of squares. Hence, L is positive semi-definite.

Now let's define the inner product. On page 215 of *Linear Algebra Done Wrong* by Sergei Treil, we know that given a positive semi-definite matrix, G, we can define a non-standard inner product in $\mathbb{C}^n$ as follows:

$$(x, y)_G := (Gx, y)_{\mathbb{C}^n} \qquad x, y \in \mathbb{C}^n$$

The inner product on the right hand side is the standard inner product in $\mathbb{C}^n$. Hence the inner product in our space is $(Lx, y)_{\mathbb{C}^n}$ where L is the Laplacian matrix.

Given two sets of vertices x,y $\in V$, what does the inner product (Lx, y) tell us? We can rewrite (Lx, y) as follows:

$$\begin{aligned} (Lx, y) &= y^T Lx \\ &= y^T(A^T A)x \qquad \text{where A is the incidence matrix} \\ &= (Ay)^T(Ax) \\ &= \sum_{i,j\in E}(y_i - y_j)(x_i - x_j) \end{aligned}$$

Since every row of the incidence matrix only has two non zero values, we have the following cases:
Case 1: $(y_i - y_j) = 0$ and $(x_i - x_j) = 0$
This happens when neither edges (y_i, y_j) and (x_i, x_j) exists.

Case 2: $(y_i - y_j) \neq 0$ and $(x_i - x_j) = 0$, or $(y_i - y_j) = 0$ and $(x_i - x_j) \neq 0$
This happens when only one edge, either (y_i, y_j) and (x_i, x_j), exists. In both situations, $(y_i - y_j)(x_i - x_j) = 0$.

Case 3: $(y_i - y_j) \neq 0$ and $(x_i - x_j) \neq 0$
This means that both (y_i, y_j) and (x_i, x_j) exists. For this to be true, the vertices y_i, y_j must be the same as x_i, x_j since each edge can only connect two vertices. In this case $(y_i - y_j)(x_i - x_j) \neq 0$.

Denote $G_1 = (x, E_x)$ and $G_2 = (y, E_y)$ as the sub graphs generated by taking only the vertices and incident edges of x and y from the original graph G. Notice $(y_i - y_j)(x_i - x_j) \neq 0$ only when the edges $(y_i, y_j) \in E_y$ and $(x_i, x_j) \in E_x$, while $(y_i - y_j)(x_i - x_j) = 0$ otherwise. As proven previously, $y_i, y_j = x_i, x_j$ when $(y_i - y_j)(x_i - x_j) \neq 0$. Hence $\sum_{i,j \in E}(y_i - y_j)(x_i - x_j) = \sum_{i,j \in E_x \text{ AND } i,j \in E_y}(x_i - x_j)^2$. If we think about $(x_i - x_j)$ as the weight of an edge, the summation can be interpreted as the sum of the weights squared of the edges in both G_1 and G_2. When summation evaluates to 0, then no edges are shared between G_1 and G_2. This is what the inner product between x and y tells us.

Theoretical Result

Let's prove the following linear algebra fact:

Property 1: Given a n x n matrix A, then

$$det(A + E_{ii}) = det(A) + det(A[i])$$

where E_{ii} is a matrix with 1 at position (i,i) and 0 elsewhere, and $A[i]$ is the matrix where the ith column and ith row is removed.

From page 89 of Treil's *Linear Algebra Done Wrong*, we can apply the permutation definition of the determinant to matrix A:

$$det(A) = \sum_{\sigma \in Perm(n)} a_{\sigma(1),1} a_{\sigma(2),2} ... a_{\sigma(n),n} sign(\sigma)$$

Now for $det(A + E_{ii})$, the (i,i)th entry increased by 1 so a_{ii} increases to $a_{ii} + 1$. This means $det(A + E_{ii})$ is equal to the original det(A), plus $\sum_{\sigma \in Perm(n)} a_{\sigma(1),1}, a_{\sigma(2),2} ..., a_{\sigma(n),n} sign(\sigma)$ excluding $a_{\sigma(i)i}$. We see that the this summation exactly $det(A[i])$. Hence we can conclude $det(A + E_{ii}) = det(A) + det(A[i])$.

Application

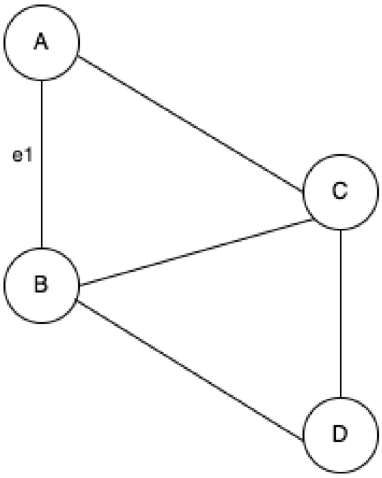
Now let's prove the following theorem: The number of spanning trees in a Graph G is given by $det(L[i])$, for any i, where L is the Laplacian matrix.

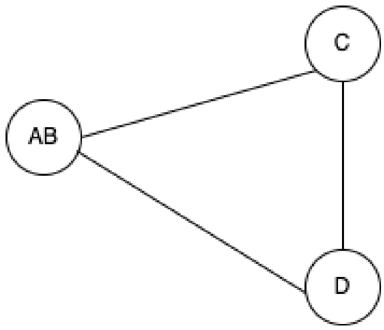
Proof by Induction:
(BASE CASE) Graph G has 2 vertices and 0 edges. We should expect the number of spanning trees to be 0 since G has 0 edges. The Laplacian matrix will look like the following:

$$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

It follows that L[i] is a 1 x 1 matrix of 0: [0]
The $det(L[i]) = 0$ meaning there are 0 spanning trees which is true.

(INDUCTION STEP) Denote t(G) as the number of spanning trees of G, $G - e$ as the graph with without edge e, and G/e as the graph where edge e is contracted. A contraction is when two vertices, A and B, combine into one vertex, AB, such that all incident edges of A and B are incident to AB. See the example below where e1 in the left graph was contracted to form the graph on the right:





Induction Hypothesis: Assume that for any graph F with at most n-1 vertices, the number of spanning trees t(F) = $det(L_F[i])$ for any vertex i. We want to show that for a graph with n vertices, the number of spanning trees is the determinant of the Laplacian with some column and row i removed.

Suppose we have a graph G with n vertices. Choose any arbitrary vertex i, and incident edge e=(i,j) in G. For any spanning tree, denoted as T, it either contains e or not. The number of spanning trees that contain e is t(G/e), because if we contract e that means that edge has to be apart of the tree. The number of spanning trees that do not contain e is t(G-e), because if we directly remove e it cannot be in a spanning tree. Then the number of spanning trees for G is t(G) = t(G/e) + t(G-e) because every T either contains e or not. Note that G/e is G with one less vertex, and G - e is G with one less edge.

Let's first examine G-e. We can relate G and G-e through their Laplacian matrices, L_G and L_{G-e}, since the difference between G and G-e is that the degrees of vertices, i and j, are different and edge e is not in G-e. If we remove column i and row i from L_G, then $L_G[i] = L_{G-e}[i] + E_{jj}$, since removing the edge e will cause the degree of vertex j to decrease by 1.

$$\begin{aligned}
det(L_G[i]) &= det(L_{G-e}[i] + E_{jj}) \\
&= det(L_{G-e}[i]) + det(L_{G-e}[ij]) \qquad \text{(by Property 1)} \\
&= det(L_{G-e}[i]) + det(L_G[ij])
\end{aligned}$$

Note that $L_{G-e}[ij]$ means we removed the ith and jth column and row. The last statement holds because once the ith and jth column is removed from $L_{G-e}[ij]$, then it is the same as $L_G[ij]$.

Now for G/e. Without loss of generality, less say we contract i onto j. Then $L_{G/e}$ has no row or column for vertex i. As a result, we can say $L_{G/e} = L_G[i]$. Subsequently, $L_{G/e}[j] = L_G[ij]$.

Finally, we have:

$$\begin{aligned}
det(L_G[i]) &= det(L_{G-e}[i]) + det(L_G[ij]) \\
&= t(L_{G-e}[i]) + t(L_G[ij])
\end{aligned}$$

By the induction hypothesis $t(L_G[ij])$ is the number of spanning trees that contain e, and $t(L_{G-e}[i])$ are the trees that do not contain e. Thus, $t(L_{G-e}[i]) + t(L_G[ij]) = t(G)$, since every spanning tree either contains e or not. We can conclude $det(L_G[i]) = t(G)$. This concludes the proof.

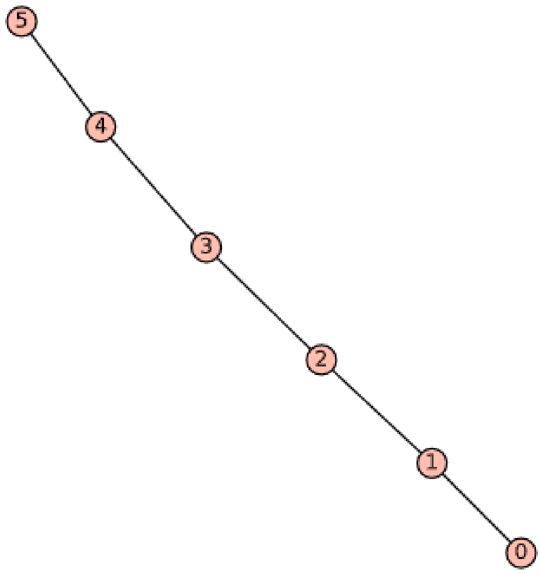
Application with Sage

Now let's apply the theorem we proved above.

Case 1

Consider a tree graph. We can generate this using sage as follows:

```
A = matrix([[-1,1,0,0,0,0],
            [0,-1,1,0,0,0],
            [0,0,-1,1,0,0],
            [0,0,0,-1,1,0],
            [0,0,0,0,-1,1]])
Graph(A.transpose()).show()
```

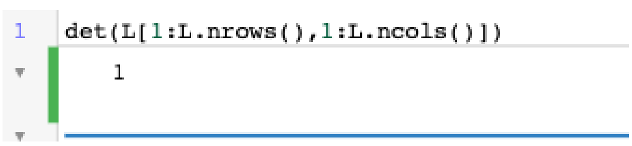


A is the incidence matrix. As discussed previously, we can calculate the Laplacian matrix from the hermitian square of the incidence matrix. The following code finds the Laplacian for the graph above

```
L = A.transpose() * A
```

Finally, the number of spanning trees will be the determinant of $L[i]$ for any i. Since the graph is already a tree we expect the result to be 1. The following code calculates this result.

```
det(L[1:L.nrows(),1:L.ncols()])
```

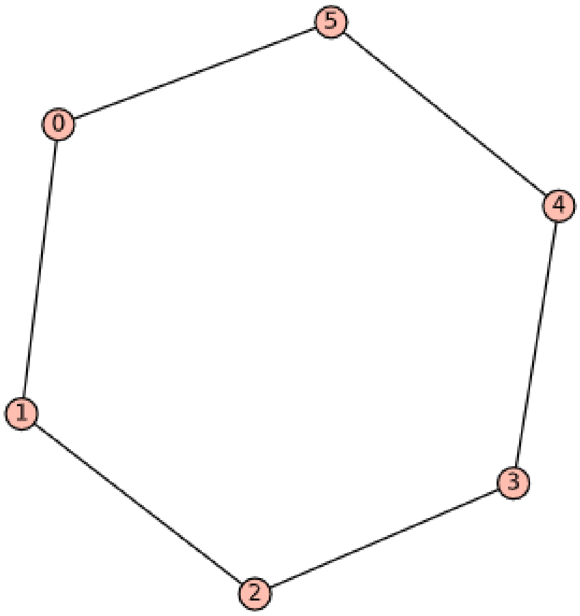


Indeed the result is 1 as expected.

Case 2

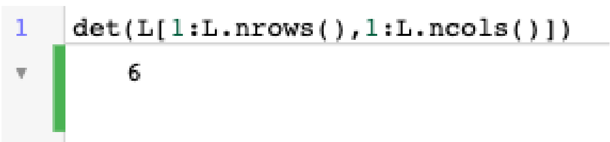
Now let's consider a cycle. We can generate this using sage as follows:

```
A = matrix([[-1,1,0,0,0,0],
            [0,-1,1,0,0,0],
            [0,0,-1,1,0,0],
            [0,0,0,-1,1,0],
            [0,0,0,0,-1,1],
            [-1,0,0,0,0,1]])
Graph(A.transpose()).show()
```



Similar to the process in case 1, we can find the Laplacian and calculate the number of spanning trees as follows. Note that we expect the number of spanning trees to be 6 since the graph is a cycle and we can remove any of the 6 edges to get a spanning tree.

```
L = A.transpose() * A
det(L[1:L.nrows(),1:L.ncols()])
```



Indeed the result is 6 as expected.

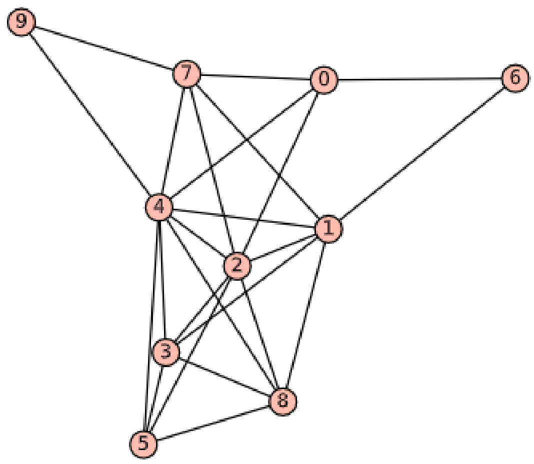
Case 3

Now let's try a random graph. We can generate this using the function:

```
def randomIncidenceMatrix(n, p):
    random = graphs.RandomGNP(n,p)
    random.show()
    G = random.incidence_matrix().transpose()
    for i in range(0, G.nrows()):
        t = false
        while (t == false):
            for j in range (0, G.ncols()):
                if ((G[i,j] == 1) & (t == false)):
                    G[i,j]=-1
                    t = true
    return G
```

Sage's built in graph generator returns an incidence matrix that doesn't have direction. Thus, to adhere to the definition of an incidence matrix defined earlier in this write-up, we arbitrarily assign direction to the edges of the incidence matrix. Let's generate a graph with 10 vertices:

```
Z = randomIncidenceMatrix(10, .5)
```



With the incidence matrix Z, we can calculate the Laplacian matrix and the number of spanning trees:

```
L = Z.transpose() * Z
det(L[1:L.nrows(),1:L.ncols()])
```

```
L = Z.transpose() * Z
det(L[1:L.nrows(),1:L.ncols()])
```

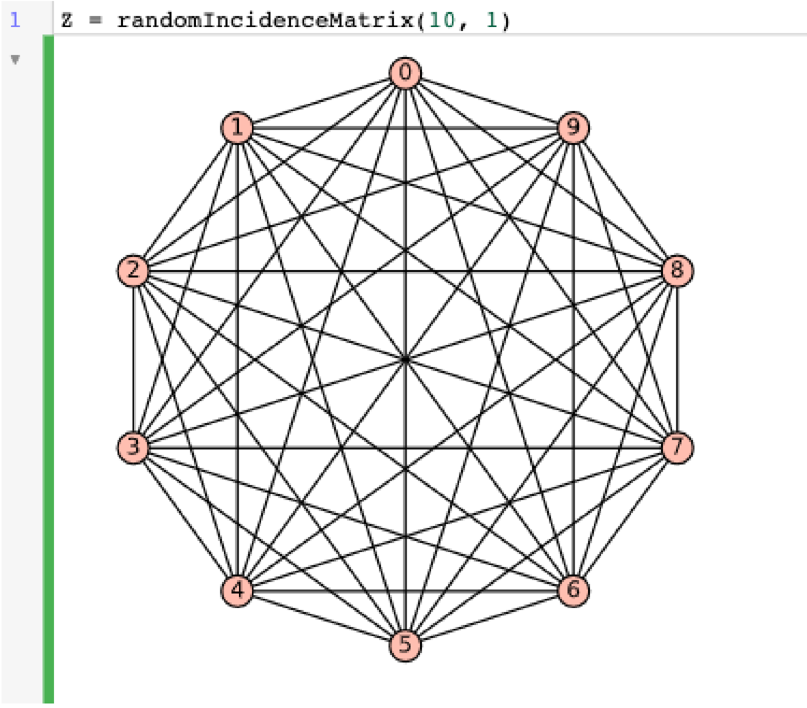
```
92604
```

Our code tells us that this random graph above can form 92604 spanning trees.

Case 4

Now let's consider a complete graph. Using the *randomIncidenceMatrix*() function above we can generate the following graph:

```
Z = randomIncidenceMatrix(10, 1)
```



With the incidence matrix Z, we can calculate the Laplacian matrix and the number of spanning trees:

```
L = Z.transpose() * Z
det(L[1:L.nrows(),1:L.ncols()])
```

```
L = Z.transpose() * Z
det(L[1:L.nrows(),1:L.ncols()])

    100000000
```

Our result tells us that there are 100000000 spanning trees. This is expected because for a complete graph there should be 2^{n-2} spanning trees where n is the number of nodes (proof omitted). Since there are 10 nodes, our graph should have $2^8 = 100000000$ spanning trees, which is exaclty what our code tells us.

Runtime

Assume we're given a incidence matrix. The process we used above consists of the the following steps:
1. Finding Laplacian
2. Calculating determinant of $L[i]$

Denote m as the number of edges and n as the number of vertices.
To find the Laplacian we multiply the incidence matrix transpose with the incidence matrix. The incidence matrix is m by n, so there are a total of $m * n^2$ multiplications.

The Laplacian L is a n by n matrix. Then $L[i]$ is a n-1 by n-1 matrix. So the number of multiplications will be $O(n!)$.

The dominating term is the calculation of the determinant, so the overall number of multiplications will be $O(n!)$.

Sources

A First Course in Linear Algebra, Robert Beezer

Linear Algebra Done Wrong, Sergei Treil (2017 version)

18.409 An Algorithmist's Toolkit Lecture 2, Jonathan Kelner

ORIE 6334 Bridging Continuous and Discrete Optimization Lecture 8, David P. Williamson