Pescara, Italy, July 2019

DIGRAPHS IV
The Matrix Tree Theorem

Based on various sources.

J. J. P. Veerman,
Math/Stat, Portland State Univ., Portland, OR
97201, USA.
email: veerman@pdx.edu

Conference Website:
www.sci.unich.it/mmcs2019
SUMMARY:

* Matrix tree theorems connect different branches of mathematics (combinatorics, linear algebra, probability) in unexpected ways. For this reason, they play an important role in the graph theory literature.

* We give a detailed description of various matrix tree theorems. These theorems relate the determinant of certain submatrices of the usual Laplacian to the number of spanning trees rooted at each vertex.

* We give a simple, short, combinatorial proof loosely inspired by [1].

* We include a discussion that relates the number of spanning trees at each vertex to the stable probability measure of random walk on a strongly connected graph.
OUTLINE:
The headings of this talk are color-coded as follows:

- Boundary Operators
- Matrix Tree Theorems
- Proof of Matrix Tree Theorems
- Trees and Unicycles
BOUNDARY OPERATORS
The Boundary Matrices

Definition: Given a digraph $G$, define matrices $B$ (for Begin) and $E$ (for End)

$$E_{ij} = \begin{cases} 
1 & \text{if vertex } i \text{ ends edge } j \\
0 & \text{else} 
\end{cases}$$

$$B_{ij} = \begin{cases} 
1 & \text{if vertex } i \text{ starts edge } j \\
0 & \text{else} 
\end{cases}$$

$$E = \begin{pmatrix} 
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 
\end{pmatrix}$$

$$B = \begin{pmatrix} 
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 
\end{pmatrix}$$

Edges are columns. vertices are rows.

Consistent with definition of boundary operator in topology:

$$\partial := E - B$$
Let $V$ the set of vertices. Want an operator mapping $\mathbb{C}^V$ to itself. Thus $EE^T$, $EB^T$, $BE^T$, and $BB^T$ are natural candidates. We investigate these operators.

**FACT 1:**

$$(EE^T)_{ij} = \sum_k E_{ik}E_{jk}$$

is the $\#$ edges that end in $i$ and in $j$.
Thus it is the **diagonal in-degree matrix**.
Similarly, $BB^T$ is the **diagonal out-degree matrix**.

**FACT 2:**

$$(EB^T)_{ij} = \sum_k E_{ik}B_{jk}$$

is the $\#$ edges that start in $j$ and end in $i$.
It is the **comb. in-degree adj. matrix** $Q$ (as in DI).
And $BE^T$ is the **comb. out-degree adj. matrix** or $Q^T$.

**Lemma:** In the notation of DI, we have:

$$D = EE^T \quad \text{and} \quad Q = EB^T$$

**Exercise 1:** Check the facts as well as the ones mentioned for $BB^T$ and $BE^T$. 
The Lemma immediately implies:

**Theorem 1:** In the notation of DI, we have:

\[ L = E(E^T - B^T) \quad \text{and} \quad L_{\text{out}} = B(B^T - E^T) \]

where \( L_{\text{out}} \) is the Laplacian of the graph \( G \) with all orientations reversed.

The example in the next pages illustrate the following two remarks.

**Remark 1:** Be careful to note that \( L_{\text{out}} \neq L^T \) !!

**Remark 2:** Note that the sum of \( L \) and \( L_{\text{out}} \) is the Laplacian of the underlying graph \( G \). Thus:

**Corollary:** We have:

\[ L = L + L_{\text{out}} = (E - B)(E^T - B^T) = \partial \partial^T \]

**Remark:** This is the traditional definition of the Laplacian in topology.

**Re-Definition:** \( L \) is the standard comb. Laplacian of the previous lectures. Better notation in this context: From now on, replace \( L \) by \( L_{\text{in}} \),
Example

\[ L_{in} = \begin{pmatrix}
  0 & 0 & 0 & 0 & 0 & 0 & 0 \\
  -1 & 1 & 0 & 0 & 0 & 0 & 0 \\
  0 & 0 & 1 & 0 & -1 & 0 & 0 \\
  0 & 0 & -1 & 1 & 0 & 0 & 0 \\
  0 & 0 & 0 & -1 & 1 & 0 & 0 \\
  -1 & 0 & 0 & 0 & 0 & 2 & -1 \\
  0 & 0 & -1 & 0 & 0 & -1 & 2
\end{pmatrix} \]

\[ L_{out} = \begin{pmatrix}
  2 & -1 & 0 & 0 & 0 & -1 & 0 \\
  0 & 0 & 0 & 0 & 0 & 0 & 0 \\
  0 & 0 & 2 & -1 & 0 & 0 & -1 \\
  0 & 0 & 0 & 1 & -1 & 0 & 0 \\
  0 & 0 & 0 & -1 & 1 & 0 & 0 \\
  0 & 0 & 0 & 0 & 0 & 1 & -1 \\
  0 & 0 & 0 & 0 & 0 & -1 & 1
\end{pmatrix} \]

And \( L = L_{in} + L_{out} \) is symmetric.


**Weighted Laplacians**

**Definition:** We can “weight” the edges. Let $W$ be a diagonal weight matrix.

$$L_{\text{in},W} = (EW)(E^T - B^T)$$

We drop the subscript “$W$”. In particular

$$L_{\text{in}} = (ED^{-1})(E^T - B^T)$$

where $D_{ii} = 1$ if the in-degree in 0. (see DI)

**Remark:** Note that

$$[(EW)B^T]_{ij} = \sum_k E_{ik}W_{kk}B_{jk}$$

which means the weights go to the edges (not the vertices).

**Be careful:** The symbol $L_{\text{in}}$ is reserved for the out-degree rw Laplacian. The edges have a weight different from that of $L_{\text{in}}$. See example.
Example with Weights

\[ L_{\text{in}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ -1/2 & 0 & 0 & 0 & 0 & 1 & -1/2 \\ 0 & 0 & -1/2 & 0 & 0 & -1/2 & 1 \end{pmatrix} \]

\[ L_{\text{out}} = \begin{pmatrix} 1 & -1/2 & 0 & 0 & 0 & -1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1/2 & 0 & 0 & -1/2 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{pmatrix} \]

Notice that the sum of these two is NOT symmetric. Edge 6 has received two different weights.
FORMULATION OF THE MATRIX TREE THEOREM
**Lots of Trees**

**Definition:** For the purpose of this section, we write:

\[
L_{\text{in}} = (EW)(E^T - B^T)
\]

\[
L_{\text{out}} = (-BW)(E^T - B^T)
\]

\[
L = (EW - BW)(E^T - B^T)
\]

\[
= (E - B)W(E^T - B^T)
\]

**Definition:** A spanning **out**-tree rooted at vertex \( r \) (**SOTR**) is a graph such that

- if \( i \neq r \), then **in**-degree at \( i \) equals 1.
- **in**-degree at \( r \) equals 0.
- no directed cycles.

For a **SITR**: swap “out” and “in”.

Figure: Left: **out**-tree rooted at \( r \), and right: **in**-tree.

**Definition:** A spanning **undirected** tree rooted at \( r \) (**SUTR**) is a connected graph with no cycles. (No loose vertices.)
And To Each Their Tree

\[
L_{\text{in}} = (EW)(E^T - B^T) \\
L_{\text{out}} = (-BW)(E^T - B^T) \\
L = (EW - BW)(E^T - B^T)
\]

\[
(EW)_{ij} = \sum_k E_{ik}W_{kj}
\]

So the effect of the diagonal matrix \( W \) is to multiply the \( i \)th edge (column) by the \( i \)th entry \( W_{ii} \).

**Definition:** The **weight** \( W(T) \) of a tree \( T \) is the product of the weights of all its edges.

**Definition:** For a Laplacian \( L \), let \( \mathcal{W}_r \) be the **appropriate** set of spanning trees rooted at \( r \). By this we mean:
- For \( L_{\text{in}} \), it is the SOTR’s
- For \( L_{\text{out}} \), it is the SITR’s
- For \( L \), it is the SUTR’s.
Matrix Tree Theorems

**Definition:** Assume $G$ has $n$ vertices. Let $I_r$ be the set of all vertices except $r$.

**Theorem 2 (Matrix Tree):** $L$ a Laplacian. Then

$$q_r := \det L[I_r, I_r] = \sum_{T_r \in \mathcal{W}_r} W(T_r)$$

**Observation 1:** If $G$ has $k > 1$ reaches, then no SORTs.

DII Thm 9: $L$ has eval 0 with mult. $k > 1$. Reducing $L$ by 1 column and row will give $\det L[I_r, I_r] = 0$.

**Exercise 2:** Show that for a digraph $G$ with one reach, if $r$ is not in a cabal, then $\det L[I_r, I_r] = 0$.

The proofs of the cases where $L = L_{in}$ or $L = L_{out}$ are almost identical (just swap “in” and “out”). The undirected is slightly different, but the same techniques work.

**Theorem 3:** Furthermore

$$\sum_r q_r L_{ri} = 0$$

**Observation 2:** Thus the **weight** of rooted trees at vertex $r$ has a probabilistic interpretation. (Gives stationary probability measure under rw.)
**Exercise 2:** For the graph above write out $L_{in}$, $L_{out}$, and $L$.

**Exercise 3:** Let $q_k$ the weight of out-trees rooted in vertex $k$. Show that $q_k = \prod_{k+1}^n a_i \prod_{i=1}^{k-1} b_i$.

**Exercise 4:** Show that $qL_{in} = 0$.

**Exercise 5:** Repeat exercises 2 and 3 for $L_{out}$, and $L$. 
PROOF OF
MATRIX TREE THEOREMS
First Use Cauchy-Binet

Definition (DI): $I$ ($K$) subset of the row (column) labels of matrix $A$. $A[I, K]$ consists of the entries of $A$ in $I \times K$.

Exercise 6: $L = AB$ where $A$ and $B$ matrices as depicted above. Show that matrix multiplication implies

$$L[I, J] = A[I, \text{all}] B[\text{all}, J]$$

Now let $|I| = |J| = k$. By Cauchy-Binet (Thm 3 of DI):

$$\det((AB)[I, J]) = \sum_{K, |K| = k} \det(A[I, K]) \det(B[K, J])$$

Since $L_{\text{in}} = (EW)(E^T - B^T)$, we have

Proposition: $I_r := V \setminus \{r\}$. Then $\det(L_{\text{in}}[I_r, I_r])$ equals

$$\sum_{K, |K| = n-1} \det((EW)[I_r, K]) \det((E^T - B^T)[K, I_r])$$
Recall: SOTR is a graph such that

1. if \( i \neq r \), then in-degree at \( i \) equals 1.
2. in-degree at \( r \) equals 0.
3. no directed cycles.

\[
\det(L_{\text{in}}[I_r, I_r]) = \sum_K \det((EW)[I_r, K]) \det((E^T - B^T)[K, I_r])
\]

In RHS, each choice of \( K \) selects \( n - 1 \) edges.

If the \( n - 1 \) edges \( K \) do not form a SOTR:
They fail 1. or 2. \( \implies \) \( E \) has column of zeroes, or
they fail 3. \( \implies \) \((E^T - B^T)\) contains a cycle-Laplacian.

Example w. 6 vertices and 5 edges: Left: column 5 of \( E[I_r, K] \) is 0. Right: \((E^T - B^T)\) \([\{2, 3, 4, 5\}, \{2, 3, 4, 5\}]\) has row sum 0.

Total contribution: zero!
Assume Tree

If the \( n-1 \) edges \( K \) do form a SOTR:

Relabel vertices twice, so that:

1. If \( j > i \), then path from \( r \leadsto i \) does not pass through \( j \).
2. And then so that edge \( i \) ends in vertex \( i \).

For each \( K \), the same permutations are done in two factors:

\[
\sum_{K, |K|=n-1} \det((EW)[I_r, K]) \det((E^T - B^T)[K, I_r])
\]

Thus no net effect!

Result: \( E[I_r, K] \) is the identity, and \( B[I_r, K] \) is upper tridiag with 0 on diag. Has det equal to 0.

Example w. 6 vertices and 5 edges: Left: Before

permutations. Right: After.

Total contribution: The weight of the tree!

Exercise 7: Repeat proof for \( L_{out} \) (trivial) and \( L \) (needs minor adaptation).
TREES, UNICYCLES, PROBABILITY
Definition: An **augmented** spanning **out**-tree rooted at vertex $r$ (**ASOTR**) is a SOTR plus 1 extra edge $k \rightarrow r$ such that $(L_{\text{in}})_{rk} > 0$. Similarly, an **ASITR** is a SITR plus 1 extra edge $r \rightarrow k$ such that $(L_{\text{out}})_{rk} > 0$.

Left: Augmented **out**-tree. Right: Augmented **in**-tree.

Definition: An augm. spanning undirected tree rooted at $r$ (**ASUTR**) is a SUTR with 1 extra edge from $r$ to a neighbor.

Remark: These graphs contain 1 cycle! They are most commonly called **cycle-rooted trees** or **unicycles**.

Definition: For a Laplacian $L$, let $\mathcal{A}_r$ be the **appropriate** set of augm. spanning trees rooted at $r$. By this we mean:
- For $L_{\text{in}}$, it is the ASOTR’s
- For $L_{\text{out}}$, it is the ASITR’s
- For $L$, it is the ASUTR’s.
Exercise 8: Show that a unicycle contains exactly 1 cycle. (Hint: contract along the spanning tree. The cycles are the remaining edges.)

Two ways to compute the weight of the $L_{in}$-appropriate $r$-rooted unicycles (ASOTR’s) for a given graph $G$ (see figure).

Left(1): To SOTR at $r$, add edge from parent $k$ of $r$ to $r$.
Right(2): To SORT at child $j$ of $r$, add edge from $r$ to $j$.

Total weight of unicycles rooted at $r$ by $u_r$.

From 1: \[ u_r = \sum_k q_r \cdot S_{rk} = q_r \cdot D_{rr} \]

From 2: \[ u_r = \sum_j q_j \cdot S_{jk} \]
Proof of Theorem 3

Equate the two expressions:

\[ q_r D_{rr} - \sum_j q_j S_{jk} = q L_{in} = 0 \]

which proves Thm 3 for \( L_{in} \).

DONE!

Remark: If \( S \) is a rw walk matrix, then \( q \) is the stationary probability measure.

Exercise 9: Prove Theorem 3 for \( L_{out} \) and \( L \).
References