2 Eulerian digraphs and oriented trees.

A famous problem which goes back to Euler asks for what graphs G is there a closed walk which uses every edge exactly once. (There is also a version for non-closed walks.) Such a walk is called an *Eulerian tour* (also known as an *Eulerian cycle*). A graph which has an Eulerian tour is called an *Eulerian graph*. Euler's famous theorem (the first real theorem of graph theory) states that G is Eulerian if and only if it is connected and every vertex has even degree. Here we will be concerned with the analogous theorem for directed graphs. We want to know not just whether an Eulerian tour exists, but how many there are. We will prove an elegant determinantal formula for this number closely related to the Matrix-Tree Theorem. For the case of undirected graphs no analogous formula is known, explaining why we consider only the directed case.

A (finite) *directed graph* or *digraph* D consists of a *vertex set* $V = \{v_1, \dots, v_p\}$ and edge set $E = \{e_1, \dots, e_q\}$, together with a function $\varphi : E \to V \times V$ (the set of ordered pairs (u, v) of elements of V). If $\varphi(e) = (u, v)$, then we think of e as an arrow from u to v. We then call u the *initial vertex* and v the *final vertex* of e. (These concepts arose in the definition of an orientation in Definition 8.5.) A *tour* in D is a sequence $e_1, e_2, \dots, e_r$ of *distinct* edges such that the final vertex of e_i is the initial vertex of e_{i+1} for all $1 \leq i \leq r - 1$, and the final vertex of e_r is the initial vertex of e_1. A tour is *Eulerian* if every edge of D occurs at least once (and hence exactly once). A digraph which has no isolated vertices and contains an Eulerian tour is called an *Eulerian digraph*. Clearly an Eulerian digraph is connected. The *outdegree* of a vertex v, denoted $\mathrm{outdeg}(v)$, is the number of edges of G with initial vertex v. Similarly the *indegree* of v, denoted $\mathrm{indeg}(v)$, is the number of edges of D with final vertex v. A loop (edge of the form (v, v)) contributes one to both the indegree and outdegree. A digraph is *balanced* if $\mathrm{indeg}(v) = \mathrm{outdeg}(v)$ for all vertices v.

2.1 Theorem. *A digraph D is Eulerian if and only if it is connected and balanced.*

Proof. Assume D is Eulerian, and let $e_1, \dots, e_q$ be an Eulerian tour. As we move along the tour, whenever we enter a vertex v we must exit it, except at the very end we enter the final vertex v of e_q without exiting it. However, at the beginning we exited v without having entered it. Hence every vertex is entered as often as it is exited and so must have the same outdegree as indegree. Therefore D is balanced, and as noted above D is clearly connected.

Now assume that D is balanced and connected. We may assume that D has at least one edge. We first claim that for any edge e of D, D has a tour for which $e = e_1$. If e_1 is a loop we are done. Otherwise we have entered the vertex $\text{fin}(e_1)$ for the first time, so since D is balanced there is some exit edge e_2. Either $\text{fin}(e_2) = \text{init}(e_1)$ and we are done, or else we have entered the vertex $\text{fin}(e_2)$ once more than we have exited it. Since D is balanced there is new edge e_3 with $\text{fin}(e_2) = \text{init}(e_3)$. Continuing in this way, either we complete a tour or else we have entered the current vertex once more than we have exited it, in which case we can exit along a new edge. Since D has finitely many edges, eventually we must complete a tour. Thus D does have a tour which uses e_1.

Now let $e_1, \dots, e_r$ be a tour C of maximum length. We must show that $r = q$, the number of edges of D. Assume to the contrary that $r < q$. Since in moving along C every vertex is entered as often as it is exited (with $\text{init}(e_1)$ exited at the beginning and entered at the end), when we remove the edges of C from D we obtain a digraph H which is still balanced, though it need not be connected. However, since D is connected, at least one connected component H_1 of H contains at least one edge and has a vertex v in common with C [why?]. Since H_1 is balanced, there is an edge e of H_1 with initial vertex v. The argument of the previous paragraph shows that H_1 has a tour C' of positive length beginning with the edge e. But then when moving along C, when we reach v we can take the "detour" C' before continuing with C. This gives a tour of length longer than r, a contradiction. Hence $r = q$, and the theorem is proved. $\square$

Our primary goal is to count the number of Eulerian tours of a connected balanced digraph. A key concept in doing so is that of an oriented tree. An *oriented tree* with root v is a (finite) digraph T with v as one of its vertices, such that there is a unique directed path from any vertex u to v. In other words, there is a unique sequence of edges $e_1, \dots, e_r$ such that (a) $\text{init}(e_1) = u$, (b) $\text{fin}(e_r) = v$, and (c) $\text{fin}(e_i) = \text{init}(e_{i+1})$ for $1 \le i \le r-1$.

It's easy to see that this means that the underlying undirected graph (i.e., "erase" all the arrows from the edges of T) is a tree, and that all arrows in T "point toward" v. There is a surprising connection between Eulerian tours and oriented trees, given by the next result (due to de Bruijn and van Aardenne-Ehrenfest). This result is sometimes called the BEST Theorem, after de **B**ruijn, van Aardenne-**E**hrenfest, **S**mith, and **T**utte. However, Smith and Tutte were not involved in the original discovery.

2.2 Theorem. *Let D be a connected balanced digraph with vertex set V. Fix an edge e of D, and let $v = init(e)$. Let $\tau(D, v)$ denote the number of oriented (spanning) subtrees of D with root v, and let $\epsilon(D, e)$ denote the number of Eulerian tours of D starting with the edge e. Then*

$$\epsilon(D, e) = \tau(D, v) \prod_{u \in V} (\text{outdeg}(u) - 1)!. \tag{6}$$

Proof. Let $e = e_1, e_2, \ldots, e_q$ be an Eulerian tour E in D. For each vertex $u \neq v$, let $e(u)$ be the "last exit" from u in the tour, i.e., let $e(u) = e_j$ where $\text{init}(e(u)) = u$ and $\text{init}(e_k) \neq u$ for any $k > j$.

Claim #1. The vertices of D, together with the edges $e(u)$ for all vertices $u \neq v$, form an oriented subtree of D with root v.

Proof of Claim #1. This is a straightforward verification. Let T be the spanning subgraph of D with edges $e(u)$, $u \neq v$. Thus if $|V| = p$, then T has p vertices and $p - 1$ edges [why?]. There are three items to check to insure that T is an oriented tree with root v:

(a) T does not have two edges f and f' satisfying $\text{init}(f) = \text{init}(f')$. This is clear since both f and f' can't be last exits from the same vertex.

(b) T does not have an edge f with $\text{init}(f) = v$. This is clear since by definition the edges of T consist only of last exits from vertices other than v, so no edge of T can exit from v.

(c) T does not have a (directed) cycle C. For suppose C were such a cycle. Let f be that edge of C which occurs after all the other edges of C in

> the Eulerian tour E. Let f' be the edge of C satisfying $\mathrm{fin}(f) = \mathrm{init}(f')$ $(= u$, say). We can't have $u = v$ by (b). Thus when we enter u *via* f, we must exit u. We can't exit u *via* f' since f occurs after f' in E. Hence f' is not the last exit from u, contradicting the definition of T.

It's easy to see that conditions (a)–(c) imply that T is an oriented tree with root v, proving the claim.

Claim #2. We claim that the following converse to Claim #1 is true. Given a connected balanced digraph D and a vertex v, let T be an oriented (spanning) subtree of D with root v. Then we can construct an Eulerian tour E as follows. Choose an edge e_1 with $\mathrm{init}(e_1) = v$. Then continue to choose any edge possible to continue the tour, except we never choose an edge f of E unless we have to, i.e., unless it's the only remaining edge exiting the vertex at which we stand. Then we never get stuck until all edges are used, so we have constructed an Eulerian tour E. Moreover, the set of last exits of E from vertices $u \neq v$ of D coincides with the set of edges of the oriented tree T.

Proof of Claim #2. Since D is balanced, the only way to get stuck is to end up at v with no further exits available, but with an edge still unused. Suppose this is the case. At least one unused edge must be a last exit edge, i.e., an edge of T [why?]. Let u be a vertex of T closest to v in T such that the unique edge f of T with $\mathrm{init}(f) = u$ is not in the tour. Let $y = \mathrm{fin}(f)$. Suppose $y \neq v$. Since we enter y as often as we leave it, we don't use the last exit from y. Thus $y = v$. But then we can leave v, a contradiction. This proves Claim #2.

We have shown that every Eulerian tour E beginning with the edge e has associated with it a "last exit" oriented subtree $T = T(E)$ with root $v = \mathrm{init}(e)$. Conversely, given an oriented subtree T with root v, we can obtain all Eulerian tours E beginning with e and satisfying $T = T(E)$ by choosing for each vertex $u \neq v$ the order in which the edges from u, except the edge of T, appear in E; as well as choosing the order in which all the edges from v except for e appear in E. Thus for each vertex u we have $(\mathrm{outdeg}(u) - 1)!$ choices, so for each T we have $\prod_u (\mathrm{outdeg}(u) - 1)!$ choices. Since there are $\tau(G, v)$ choices for T, the proof is complete. $\square$

2.3 Corollary. *Let D be a connected balanced digraph, and let v be a vertex of D. Then the number $\tau(D, v)$ of oriented subtrees with root v is independent of v.*

Proof. Let e be an edge with initial vertex v. By equation (6), we need to show that the number $\epsilon(G, e)$ of Eulerian tours beginning with e is independent of e. But $e_1 e_2 \cdots e_q$ is an Eulerian tour if and only if $e_i e_{i+1} \cdots e_q e_1 e_2 \cdots e_{i-1}$ is also an Eulerian tour, and the proof follows [why?]. $\square$

What we obviously need to do next is find a formula for $\tau(G, v)$. Such a formula is due to W. Tutte in 1948. This result is very similar to the Matrix-Tree Theorem, and indeed we will show (Example 2.6) that the Matrix-Tree Theorem is a simple corollary to Theorem 2.4.

2.4 Theorem. *Let D be a loopless connected digraph with vertex set $V = \{v_1, \ldots, v_p\}$. Let $\mathbf{L}(D)$ be the $p \times p$ matrix defined by*

$$\mathbf{L}_{ij} = \begin{cases} -m_{ij}, & \textit{if } i \neq j \textit{ and there are } m_{ij} \textit{ edges with} \\ & \quad \textit{initial vertex } v_i \textit{ and final vertex } v_j \\ \operatorname{outdeg}(v_i), & \textit{if } i = j. \end{cases}$$

(Thus $\mathbf{L}$ is the directed analogue of the laplacian matrix of an undirected graph.) Let $\mathbf{L}_0$ denote $\mathbf{L}$ with the last row and column deleted. Then

$$\det \mathbf{L}_0 = \tau(D, v_p). \tag{7}$$

NOTE. If we remove the ith row and column from $\mathbf{L}$ instead of the last row and column, then equation (7) still holds with v_p replaced with v_i.

Proof (sketch). Induction on q, the number of edges of D. The fewest number of edges which D can have is $p-1$ (since D is connected). Suppose then that D has $p-1$ edges, so that as an undirected graph D is a tree. If D is not an oriented tree with root v_p, then some vertex $v_i \neq v_p$ of D has outdegree 0 [why?]. Then $\mathbf{L}_0$ has a zero row, so $\det \mathbf{L}_0 = 0 = \tau(D, v_p)$. If on the other hand D is an oriented tree with root v_p, then an argument like that used to prove Lemma 1.7 (in the case when S is the set of edges of a spanning tree) shows that $\det \mathbf{L}_0 = 1 = \tau(D, v_p)$.

Now assume that D has $q > p-1$ edges, and assume the theorem for digraphs with at most $q-1$ edges. We may assume that no edge f of D has initial vertex v, since such an edge belongs to no oriented tree with root v and also makes no contribution to $\mathbf{L}_0$. It then follows, since D has at least p edges, that there exists a vertex $u \neq v$ of D of outdegree at least two. Let e be an edge with $\text{init}(e) = u$. Let D_1 be D with the edge e removed. Let D_2 be D with all edges e' removed such that $\text{init}(e) = \text{init}(e')$ and $e' \neq e$. (Note that D_2 is strictly smaller than D since $\text{outdeg}(u) \geq 2$.) By induction, we have $\det \mathbf{L}_0(D_1) = \tau(D_1, v_p)$ and $\det \mathbf{L}_0(D_2) = \tau(D_2, v_p)$. Clearly $\tau(D, v_p) = \tau(D_1, v_p) + \tau(D_2, v_p)$, since in an oriented tree T with root v_p, there is exactly one edge whose initial vertex coincides with that of e. On the other hand, it follows immediately from the multilinearity of the determinant [why?] that

$$\det \mathbf{L}_0(D) = \det \mathbf{L}_0(D_1) + \det \mathbf{L}_0(D_2).$$

From this the proof follows by induction. □

2.5 Corollary. *Let D be a connected balanced digraph with vertex set $V = \{v_1, \ldots, v_p\}$. Let e be an edge of D. Then the number $\epsilon(D, e)$ of Eulerian tours of D with first edge e is given by*

$$\epsilon(D, e) = (\det \mathbf{L}_0(D)) \prod_{u \in V} (\text{outdeg}(u) - 1)!.$$

Equivalently (using Lemma 1.9), if $\mathbf{L}(D)$ has eigenvalues $\mu_1, \ldots, \mu_p$ with $\mu_p = 0$, then

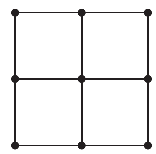
$$\epsilon(D, e) = \frac{1}{p} \mu_1 \cdots \mu_{p-1} \prod_{u \in V} (\text{outdeg}(u) - 1)!.$$

Proof. Combine Theorems 2.2 and 2.4. □

2.6 Example. (the Matrix-Tree Theorem revisited) Let G be a connected loopless undirected graph. Let $\hat{G}$ be the digraph obtained from G by replacing each edge $e = uv$ of G with a pair of directed edges $u \to v$ and $v \to u$. Clearly $\hat{G}$ is balanced and connected. Choose a vertex v of G. There is an obvious one-to-one correspondence between spanning trees T of G and oriented spanning trees $\hat{T}$ of $\hat{G}$ with root v, namely, direct each edge of T

toward v. Moreover, $\mathbf{L}(G) = \mathbf{L}(\hat{G})$ [why?]. Hence the Matrix-Tree Theorem is an immediate consequence of the Theorem 2.4.

2.7 Example. (the efficient mail carrier) A mail carrier[2] has an itinerary of city blocks to which he (or she) must deliver mail. He wants to accomplish this by walking along each block twice, once in each direction, thus passing along houses on each side of the street. The blocks form the edges of a graph G, whose vertices are the intersections. The mail carrier wants simply to walk along an Eulerian tour in the digraph $\hat{G}$ of the previous example. Making the plausible assumption that the graph is connected, not only does an Eulerian tour always exist, but we can tell the mail carrier how many there are. Thus he will know how many different routes he can take to avoid boredom. For instance, suppose G is the 3×3 grid illustrated below.



This graph has 128 spanning trees. Hence the number of mail carrier routes beginning with a fixed edge (in a given direction) is $128 \cdot 1!^4 2!^4 3! = 12288$. The total number of routes is thus 12288 times twice the number of edges [why?], viz., $12288 \times 24 = 294912$. Assuming the mail carrier delivered mail 250 days a year, it would be 1179 years before he would have to repeat a route!

2.8 Example. (binary de Bruijn sequences) A *binary sequence* is just a sequence of 0's and 1's. A *binary de Bruijn sequence* of degree n is a binary sequence $A = a_1 a_2 \cdots a_{2^n}$ such that every binary sequence $b_1 \cdots b_n$ of length n occurs exactly once as a "circular factor" of A, i.e., as a sequence $a_i a_{i+1} \cdots a_{i+n-1}$, where the subscripts are taken modulo n if necessary. For instance, some circular factors of the sequence *abcdefg* are *a*, *bcde*, *fgab*, and *defga*. Note that there are exactly 2^n binary sequences of length n, so the only possible length of a binary de Bruijn sequence of degree n is 2^n [why?]. Clearly any cyclic shift $a_i a_{i+1} \cdots a_{2^n} a_1 a_2 \cdots a_{i-1}$ of a binary de Bruijn sequence $a_1 a_2 \cdots a_{2^n}$ is also a binary de Bruijn sequence, and we call two such

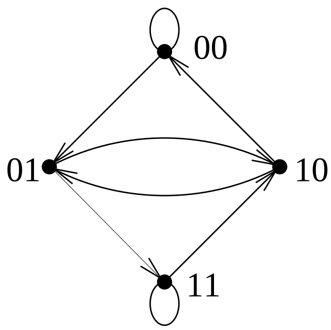
[2] postperson?

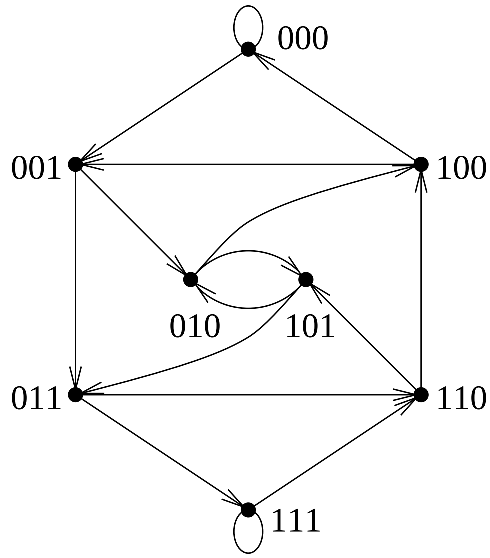
sequences *equivalent*. This relation of equivalence is obviously an equivalence relation, and every equivalence class contains exactly one sequence beginning with n 0's [why?]. Up to equivalence, there is one binary de Bruijn sequence of degree two, namely, 0011. It's easy to check that there are two inequivalent binary de Bruijn sequences of degree three, namely, 00010111 and 00011101. However, it's not clear at this point whether binary de Bruijn sequences exist for all n. By a clever application of Theorems 2.2 and 2.4, we will not only show that such sequences exist for all positive integers n, but we will also count the number of them. It turns out that there are *lots* of them. For instance, the number of inequivalent binary de Bruijn sequences of degree eight is equal to

$$1329227995784915872903807060280344576.$$

The reader with some extra time on his or her hands is invited to write down these sequences. De Bruijn sequences are named after Nicolaas Govert de Bruijn, who published his work on this subject in 1946. However, it was discovered in 1975 that de Bruijn sequences had been earlier created and enumerated by C. Flye Sainte-Marie in 1894. De Bruijn sequences have a number of interesting applications to the design of switching networks and related topics.

Our method of enumerating binary de Bruijn sequence will be to set up a correspondence between them and Eulerian tours in a certain directed graph D_n, the *de Bruijn graph* of degree n. The graph D_n has 2^{n-1} vertices, which we will take to consist of the 2^{n-1} binary sequences of length $n-1$. A pair $(a_1a_2\cdots a_{n-1}, b_1b_2\cdots b_{n-1})$ of vertices forms an edge of D_n if and only if $a_2a_3\cdots a_{n-1} = b_1b_2\cdots b_{n-2}$, i.e., e is an edge if the last $n-2$ terms of $\mathrm{init}(e)$ agree with the first $n-2$ terms of $\mathrm{fin}(e)$. Thus every vertex has indegree two and outdegree two [why?], so D_n is balanced. The number of edges of D_n is 2^n. Moreover, it's easy to see that D_n is connected (see Lemma 2.9). The graphs D_3 and D_4 look as follows:





Suppose that $E = e_1 e_2 \cdots e_{2^n}$ is an Eulerian tour in D_n. If $\mathrm{fin}(e_i)$ is the binary sequence $a_{i1}a_{i2}\cdots a_{i,n-1}$, then replace e_i in E by the last bit $a_{i,n-1}$. It is easy to see that the resulting sequence $\beta(E) = a_{1,n-1}a_{2,n-1}\cdots a_{2^n,n-1}$ is a binary de Bruijn sequence, and conversely every binary de Bruijn sequence arises in this way. In particular, since D_n is balanced and connected there exists at least one binary de Bruijn sequence. In order to count the total number of such sequences, we need to compute $\det \mathbf{L}(D_n)$. One way to do this is by a clever but messy sequence of elementary row and column operations which transforms the determinant into triangular form. We will give instead an elegant computation of the eigenvalues of $\mathbf{L}(D_n)$ based on the following simple lemma.

2.9 Lemma. *Let u and v be any two vertices of D_n. Then there is a unique (directed) walk from u to v of length $n-1$.*

Proof. Suppose $u = a_1 a_2 \cdots a_{n-1}$ and $v = b_1 b_2 \cdots b_{n-1}$. Then the unique path of length $n-1$ from u to v has vertices

$$a_1a_2\cdots a_{n-1}, a_2a_3\cdots a_{n-1}b_1, a_3a_4\cdots a_{n-1}b_1b_2, \ldots, a_{n-1}b_1\cdots b_{n-2}, b_1b_2\cdots b_{n-1}. \ \square$$

2.10 Theorem. *The eigenvalues of $\mathbf{L}(D_n)$ are 0 (with multiplicity one) and 2 (with multiplicity $2^{n-1}-1$).*

Proof. Let $\mathbf{A}(D_n)$ denote the directed adjacency matrix of D_n, i.e., the rows and columns are indexed by the vertices, with

$$\mathbf{A}_{uv} = \begin{cases} 1, & \text{if } (u,v) \text{ is an edge} \\ 0, & \text{otherwise.} \end{cases}$$

Now Lemma 2.9 is equivalent to the assertion that $\mathbf{A}^{n-1} = J$, the $2^{n-1} \times 2^{n-1}$ matrix of all 1's [why?]. If the eigenvalues of $\mathbf{A}$ are $\lambda_1, \ldots \lambda_{2^{n-1}}$, then the eigenvalues of $J = A^{n-1}$ are $\lambda_1^{n-1}, \ldots, \lambda_{2^{n-1}}^{n-1}$. By Lemma 1.4, the eigenvalues of J are 2^{n-1} (once) and 0 ($2^{n-1} - 1$ times). Hence the eigenvalues of $\mathbf{A}$ are 2ζ (once, where ζ is an $(n-1)$-st root of unity to be determined), and 0 ($2^{n-1} - 1$ times). Since the trace of $\mathbf{A}$ is 2, it follows that $\zeta = 1$, and we have found all the eigenvalues of A.

Now $\mathbf{L}(D_n) = 2I - \mathbf{A}(D_n)$ [why?]. Hence the eigenvalues of $\mathbf{L}$ are $2 - \lambda_1, \ldots, 2 - \lambda_{2^{n-1}}$, and the proof follows from the above determination of $\lambda_1, \ldots, \lambda_{2^{n-1}}$. $\square$

2.11 Corollary. *The number $B_0(n)$ of binary de Bruijn sequences of degree n beginning with n 0's is equal to $2^{2^{n-1}-n}$. The total number $B(n)$ of binary de Bruijn sequences of degree n is equal to $2^{2^{n-1}}$.*

Proof. By the above discussion, $B_0(n)$ is the number of Eulerian tours in D_n whose first edge the loop at vertex $00\cdots0$. Moreover, the outdegree of every vertex of D_n is two. Hence by Corollary 2.5 and Theorem 2.10 we have

$$B_0(n) = \frac{1}{2^{n-1}} 2^{2^{n-1}-1} = 2^{2^{n-1}-n}.$$

Finally, $B(n)$ is obtained from $B_0(n)$ by multiplying by the number 2^n of edges, and the proof follows. $\square$

Note that the total number of binary sequences of length 2^n is $N = 2^{2^n}$. By the previous corollary, the number of these which are de Bruijn sequences is just $\sqrt{N}$. This suggests the following unsolved problem. Let $\mathcal{A}_n$ be the set of all binary sequences of length 2^n. Let $\mathcal{B}_n$ be the set of binary de Bruijn sequences of degree n. Find an explicit bijection $\varphi : \mathcal{B}_n \times \mathcal{B}_n \to \mathcal{A}_n$, thereby giving a combinatorial proof of Corollary 2.11.