Graphs, Walks, and Matrix-Tree Theorem

Def: A walk in a directed graph (V, E) is a sequence $v_1, e_1, v_2, e_2, \ldots, v_r$ where $e_i = (v_i, v_{i+1})$.

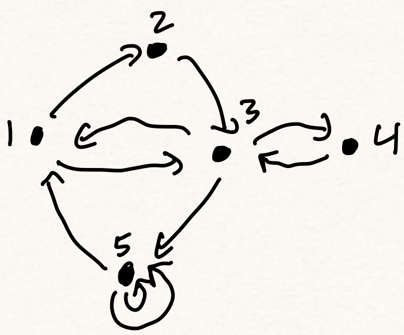
Length = # edges.

Def: A walk is a path if no edges are repeated.

A walk is a circuit if it ends at the vertex it started on. (also: a tour).

Def: An Eulerian tour is a tour that uses every edge exactly once.

Counting walks



Adjacency Matrix

$$A = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 2 \end{pmatrix} \end{array}$$

$$A^2 = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 2 & 0 & 2 \\ 1 & 0 & 0 & 1 & 1 \\ 2 & 1 & 1 & 0 & 4 \end{pmatrix}$$

Note: A^2_{ij} = # walks length 2 from i to j.

In general: If A = adjacency matrix,

$$\boxed{A^n_{ij} = \#\text{ walks length } n \text{ from } i \text{ to } j.}$$

Eulerian tours

Thm: $D = (V, E)$ has an Eulerian tour iff it is

① Weakly connected - underlying undirected graph is connected

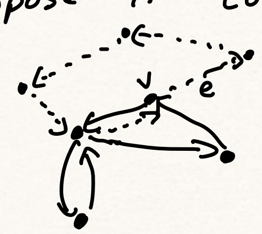
② Balanced: $\text{indeg}(v) = \text{outdeg}(v)$ for all v.

Pf: ($\Rightarrow$) Clear.

($\Leftarrow$) Assume it is weakly connected and balanced.

Notice that tours of various sizes do exist; like v (with no edges).

Consider a tour w/ a maximal # of edges that does not repeat edges. If it does not contain all edges, consider some edge that is connected to the tour but not in the tour, and suppose it connects to vertex v in the tour.



Since $\text{indeg}(v) = \text{outdeg}(v)$ we can choose an edge

$v \xrightarrow{e} \bullet$ not in the tour and follow more edges not in the tour until we return to v. This extends the tour.

$\square$

<u>Application</u>: Binary deBruijn sequences.

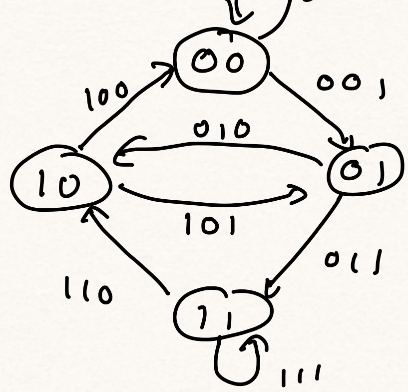
Does there exist a necklace of length 2^n w/ every length-n binary string as a (cyclic) consecutive subsequence?

Ex: (00010111) has all 8 strings of length 3.

<u>DeBruijn graph</u>: Vertices are binary strings of length $n-1$, directed edge from

$$a w_1 \cdots w_{n-2} \longrightarrow w_1 \cdots w_{n-2} b$$

labeled by $a w_1 \cdots w_{n-2} b$.



DeBruijn sequence = Eulerian tour!

00111010

Q: How many binary de Bruijn sequences are there?

In general: How many Eulerian tours?

Spanning trees

Def: An oriented tree is a rooted tree w/ every edge directed towards the root.

Def: An oriented spanning tree in $D=(V,E)$ is a pair (V,E'), $E' \subseteq E$, that is an oriented tree. (Uses all vertices)

Relation to Eulerian tours:

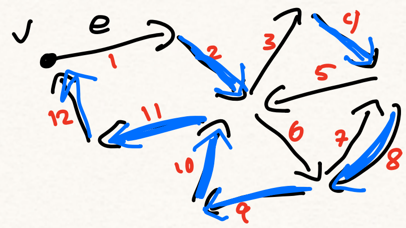
Thm: Let D be weakly connected and balanced, $e \in E$, $e=(v,w)$. Let $\tau(D,v)$ = # oriented spanning trees of D w/ root v, $\varepsilon(D,e)$ = # Eulerian tours starting at e.

Then $$\varepsilon(D,e) = \tau(D,v) \cdot \prod_{u \in V} (\text{outdeg}(u)-1)!$$

Pf: Let $d = \prod_{u \in V} (\text{outdeg}(u)-1)!$. Want a d-to-1 map

$$\varphi: \underset{\substack{\uparrow \\ \text{Eulerian tours} \\ \text{start @ } e}}{E(D,e)} \longrightarrow \underset{\substack{\uparrow \\ \text{oriented sp. trees, root } v.}}{T(D,v)}$$

Define φ by the "last exit" on all but vertex v:



Ex: Could have also done tour
1 2 6 7 5 3 4 8 9 10 11 12

d-to-1: Given a tree $T \in \mathcal{T}(D, v)$,

construct Eulerian tour:

- Start w/ edge e: $v \xrightarrow{e} v_1$
- At v_1, $\text{outdeg}(v) - 1$ possible edges before final edge (in T)
- Follow one, continue, next time we are at v_1 there are $\text{outdeg}(v) - 2$ choices left, and so on
- Similar for each vertex.

Note: $\varepsilon(D, e)$ indep of $e \Rightarrow \tau(D, v)$ indep. of v!

So to count Eulerian tours, suff. to count spanning trees.

Will do so w/ Matrix-Tree theorem.