Mathematical Foundations of Computer Science

CS 499, Shanghai Jiaotong University, Dominik Scheder

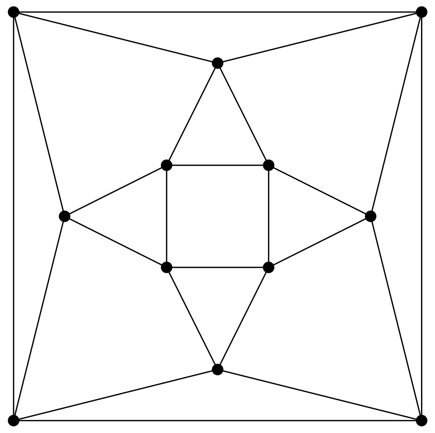
9 Hamilton Cycles, Hamilton Paths, and Non-isomorphic Trees

- Homework assignment published on Monday 2018-04-23
- Submit your first solution by Sunday, 2018-04-29, 18:00, by email
- Submit your final solution by Sunday, 2018-05-06.

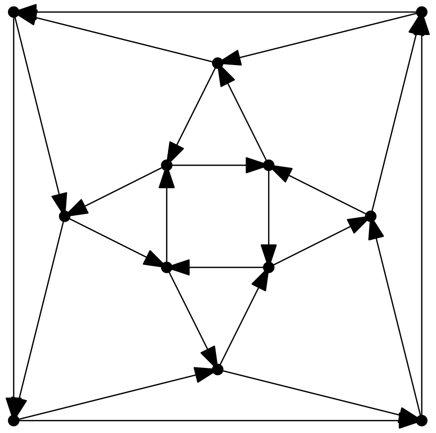
9.1 Regular Orientations of a Regular Graph

We call a graph *d-regular* if every vertex has degree d. A directed graph is (d, d)-regular if every vertex has d incoming and d outgoing edges.

Exercise 9.1. Show that in every 4-regular graph, you can orient the edges such that every vertex has two incoming and two outgoing edges, i.e., such that the resulting digraph is $(2, 2)$-regular. See the picture below for an illustration.



a 4-regular graph



a (2,2)-regular orientation

Proof. A graph G has an Eulerian Walk if and only if G is connected and evety vertex in G has even degree.
Every 4-regular graph satisfies above criterions therefore containing Eulerian Walk. We orient edges along the Eulerian walk. Since Eulerian walk can cover all the edges only once ending at start vertex, every vertex in 4-regular graph has two incoming and two outcoming edges.

□

9.2 Hamilton Cycles and Ore's Theorem

Consider K_n, the complete graph on n vertices. For $n \geq 3$, this obviously has a Hamilton cycle. How many edges do you have to delete from K_n to destroy all Hamilton cycles? That is, what is the smallest set S such that $\left(V, \binom{V}{2} \setminus S\right)$ has no Hamilton cycle? Let s_n denote the size of this set (this depends on n, thus the notation s_n). For example, $s_2 = 0$ since K_2 has no Hamilton path to begin with; $s_3 = 1$ since removing one edge from K_3 results in a graph without a Hamilton cycle.

Exercise 9.2. Find a closed formula for s_n and prove it! **Hint.** One part will be easy. For the other part, use Ore's Theorem.

Solution. The formular is $s_n = n - 2$ when $n > 2$.

First, if we remove $n-2$ edges of a vertex v in K_n and get graph G, then v has degree one. Obviously G doesn't contain Hamilton cycle. We have $s_n \leq n-2$.

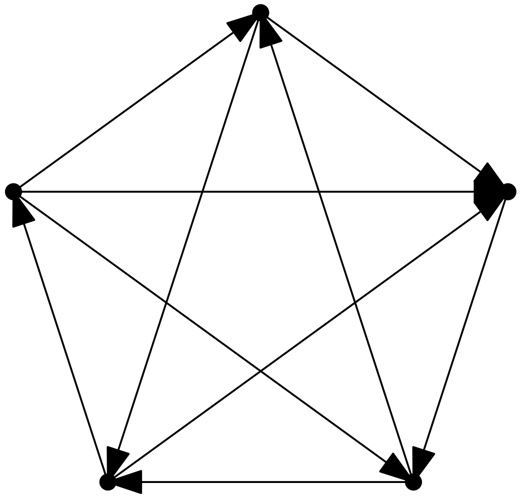
Second, if we remove $n-3$ deges of a vertex v in K_n and get graph G_1, then v has degree 2 and vertexs adjacent to v have degree $n-1$. Degree of all vertexs can be demonstrated as $(2, n-1, n-1, n-2 \ldots, n-2)$. According to Ore's theorem, G_1 contain Hamilton Cycle.

To estimate all cases when $s_n = n-3$, we add k edges adjacent to v and remove k other edges in G_1 instead. We can find that the degree of all vertexs in G_1 still meet Ore's theorem. In other words, we can't destory all Hamilton Cycle when $s_n = n-3$. Thus, $s_n > n-3$.

Since $s_n > n-3$ and $s_n \leq n-2$, we have $s_n = n-2$.

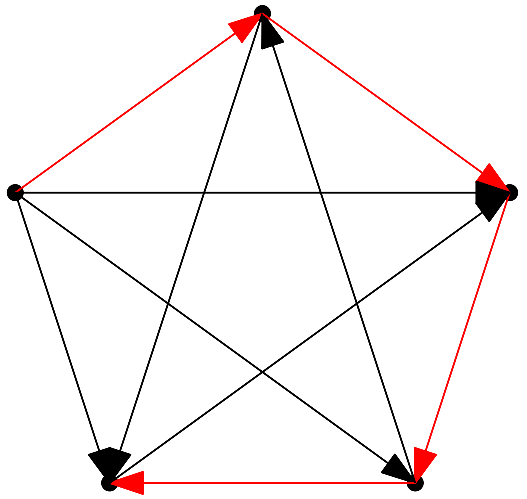
■

A *tournament* is a directed graph in which, for each pair $u, v \in V$, exactly one of the directed edges (u, v) and (v, u) is in the graph. Imagine a sports tournament in which every participant plays against every other exactly once. Draw an arc from u to v if u beat v in this tournament.



A tournament on five vertices.

Exercise 9.3. Show that every tournament has a *directed Hamilton path*, i.e., a sequence $u_1, u_2, \ldots, u_n$ such that $(u_i, u_{i+1}) \in E$ for all $i = 1, \ldots, n-1$. See the picture below.



The same tournament wiht a Hamilton path.

You probably won't be able to use the proof of Ore's Theorem directly, but you can use the proof idea.

Proof.

1. We can use an induction to prove it.
2. Base case: when there are 2 vertices, clearly there is a Hamilton path.See the picture bellow.
3. Now a graph with k vertices has a Hamilton path. Add a vertex u to it.

   Along the path, numbered the vertices as their explored order as $(v_1, v_2...v_k)$.
4. If the edge between v_k and u is (v_k, u) ,add u as the tail of the path.Then we get $(v_1, ...v_k, u)$.
5. Else turn to $v_k - 1$, if there is (v_{k-1}, u), clearly we can insert u into the path:$(v_1, v_2...v_k, u, v_{k+1}, v_k)$.
6. Else turn to the vertex before continously until there is (v_i, u), Clearly we can insert u into the path:$(v_1, v_2...v_i, u, v_{i+1}, ..., v_k)$.
7. Else all edges are from from u to v_i, add u as the head of the path.Then we get $(v_1, ...v_k, u)$ or $(u, v_1, ...v_k)$.
8. Above all, every tournament has a *directed Hamilton path.*

□

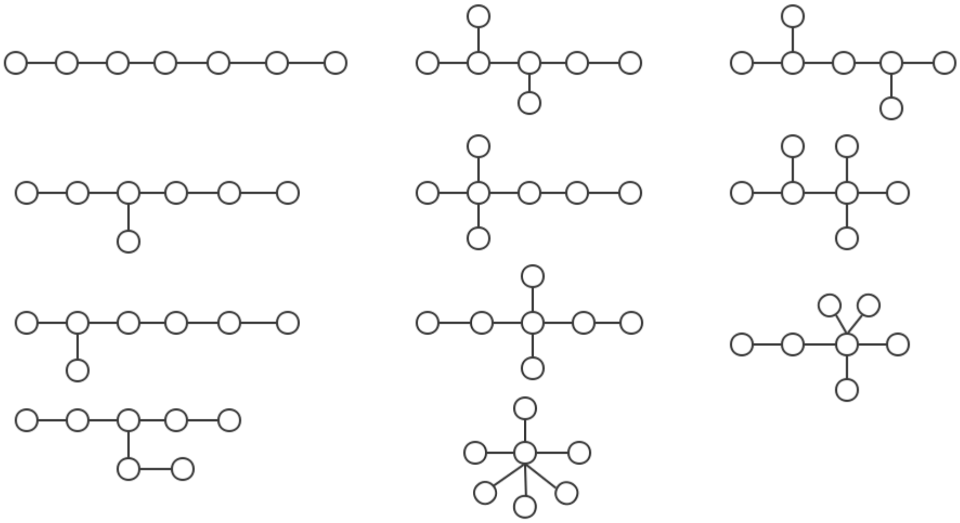
9.3 Isomorphism Classes of Trees

In the lecture (and in the videos) we have seen that the number of trees on vertex set $V = \{1, 2, \ldots, n\}$ is n^{n-2}. This however ignores isomorphisms. For example, there are $3^{3-2} = 3$ trees on vertex set $\{1, 2, 3\}$, but all those trees look alike (are isomorphic). On $\{1, 2, 3, 4\}$, there are 16 trees, but there are only two isomorphism classes: the path and the star. For five vertices, there are 125 trees but only three isomorphism classes: the path, the star, and the "T-shape" (see video on counting the number of trees). For $n = 6$ we get the path, the Y-shape, the Euro symbol, the Star Wars fighter, the Scandinavian cross, and the star, so six isomorphism classes (but a total of 1296 trees).

Exercise 9.4. List of isomorphism classes on seven vertices. That is, draw trees $T_1, \ldots, T_m$ on seven vertices such that no two of them are isomorphic but every tree on seven vertices is isomorphic to one of them. How many do you get?

n	1	2	3	4	5	6	7
number of isomorphism classes	1	1	1	2	3	6	?

Solution. We get 11 different trees.



$T_1, T_2 ... T_{11}$

■

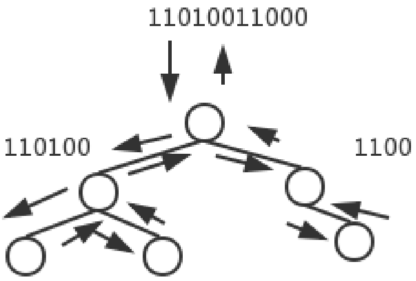
Alright, so let's denote by t_n the number of isomorphism classes of trees on n vertices. That is, t_n is the largest number m such that we can find trees $T_1, \ldots, T_m$ on n vertices such that no two of them are isomorphic. We would like to have an exact and explicit formula for t_n, but that is probably too much to ask for. Instead, let us try to understand t_n approximately and asymptotically.

Exercise 9.5. Show that $t_n \leq 4^n$. Hint: Consider the video on the isomorphism problem on trees. It defines a way to encode a tree as a 0/1-sequence.

Proof. Encode a tree as 0/1-sequence in the way below.

If there is a vertex ,we write 1 to explore it and then write 0 to return.

Take the following tree as an example.



Then if we have n vertices, we have a sequence of 2n bits.

There are $2^{2n} = 4^n$ trees.

However, in a tree if the leftchild tree of a node is the same as the rightchild tree of the same node, they are isomorphisms.BUt they have different 0/1-sequences.

Thus $t_n \leq 4^n$ □

Exercise 9.6. Show that $t_n \geq \frac{e^n}{\text{poly}(n)}$, where poly($n$) is some polynomial in n. Hint: There are n^{n-2} trees on $V = [n]$. We group them together in "buckets" of isomorphic trees. How large can a bucket be? Answer this and then use Stirling's approximation for $n!$.

Proof. For a tree with n vertices, it is less than $n!$ naming methods for the same structure.

$$t_n \geq \frac{n^{n-2}}{n!} = \frac{n^n}{n^2 \times n!} \tag{1}$$

From Stirling's approximation,

$$n! \approx \sqrt{2\pi n}(\frac{n}{e})^n \tag{2}$$

Then

$$t_n \geq \frac{e^n}{n^2\sqrt{2\pi n}}. \tag{3}$$

Let

$$poly(n) = n^2\sqrt{2\pi n}. \tag{4}$$

We have $t_n \geq \frac{e^n}{\text{poly}(n)}$, where poly($n$) is some polynomial in n. □

****Exercise 9.7.** *Try to improve those bounds. That is, find some $a < 4$ such that $t_n \in O(a^n)$ or some $b > e$ such that $t_n \in \Omega(b^n)$. Any improvement will be kind of interesting. Aim for simple proofs!*

Remark. The "true" rate of growth is known by a result of George Pólya but apparently it is quite difficult (I write "apparently" because I have never studied this work).