ECS 20 — Lecture 17b = Discussion D8 — Fall 2013 —25 Nov 2013

Phil Rogaway

Today: *Using discussion section to finish up graph theory. Much of these notes the same as those prepared for last lecture and the lecture before.*

- Graph Theory, continued

Graph theory

Graph theory
1. Basic definitions
2. Isomorphism
3. Representation of graphs
4. Paths and cycles
5. Trees
6. Eulerian and Hamiltonian graphs
7. Longest and shortest paths
8. Colorability
9. Planarity
10. Cliques and Ramsey numbers

1. Basic Definitions

Def: A (finite, simple) **graph** G=(V, E) is an ordered pair
- V is a finite nonempty set (the *vertices* or *nodes*)
- E is a set of two-elements subsets of V (the *edges*)

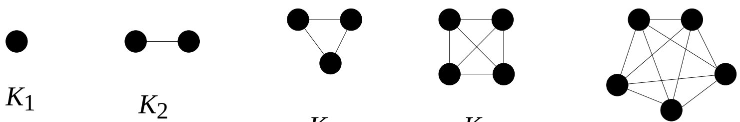
I like $\{x,y\}$ for an edge, emphasizing that $\{x,y\}$ are unordered.
Will sometimes see xy or (x,y), but both look like the order matters, which, in a simple graph, it does not.

Usually people use $n=|V|$ and $m=|E|$; alternatively, $\nu = |V|$ and $\varepsilon = |E|$ looks nice and suggestive.

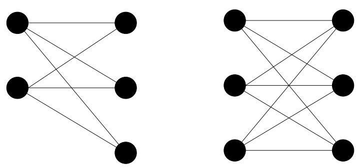
There are many other "kinds" of graphs—for example, in a **directed graph** (**digraph**), the edges (now often called **arcs**) are **ordered** pairs, instead of unordered pairs. We sometimes allow graphs with **self-loops** (and edge between a vertex and itself) or **multiple edges** (two or more different edges connecting a pair of nodes). In a network, each edge (or arcs) has a real-valued **weight**. People consider **infinite graphs**. We even have graphs where an even can be *incident* (touch) touch than two vertices (**hypergraphs**). None of these variants are not allowed in simple graphs. For this lecture, we're going to stick to them.

Conventional representation: a **picture**. (Draw some.) But be clear: the picture is **NOT** the graph, it is a **representation** of the graph. The graph is the pair (V,E).

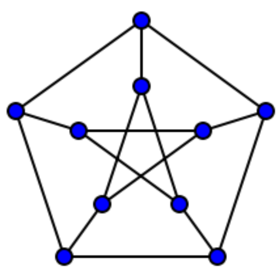
Some "special" graphs – a **clique** of size n, K_n , and complete **bipartite graphs** on n "boys" and m "girls", $K_{n,m}$.



K_1 K_2 K_3 K_4 K_5



$K_{2,3}$ $K_{3,3}$



The Peterson graph

Def: Two vertices v, w of a graph $G=(V, E)$ are **adjacent** if $\{v,w\}\in E$.
Def: The **degree** of a vertex $\deg(v) = |\{v,w\}: w\in V|$
Def: The neighbor set of v in a graph $G=(V,E)$ is $N(v) = \{w\in V: \{v,w\} \in \mathrm{E}\}$.
Note that $\deg(v) = |N(v)|$.

Some counting
Question: How many different graphs are there on $V=\{1,...,n\}$? $2^{C(n,2)} = 2^{n(n-1)/2}$
Question: what is the maximal and minimal degrees of an n-vertex graph?
Question: Count how many edges in K_n and $K_{n,m}$. .

2. Isomorphism

We don't usually care about the *names* of points in V, only how they're connected up. If two graphs are the same, up to renaming, we call them isomorphic. Formally, graphs $G=(V, E)$ and $G'=(V', E')$ are **isomorphic** if there is a permutation $\pi: V\to V'$ such that $\{v,w\}\in E$ iff $\{\pi(v),\pi(w)\}\in E'$. The properties of graphs that matter are those that are invariant under isomorphism.

Def: Graphs $G=(V,E)$ and $G'=(V',E')$ are **isomorphic** if there exists a permutation π such that $\{x,y\}\in E$ iff $\{\pi(x),\pi(y)\}\in E'$.

Proposition: Isomorphism is an equivalence relation.

Amazing fact: there is no efficient algorithm known to decide if two graphs are isomorphic. (Most computer scientists believe that no such algorithm exists.) One of the biggest open questions in computer science.

Maybe show how to prove to graphs are *non*-isomorphic using an **interactive proof**.

Prop: $\Sigma_v \deg(v) = 2m$

3. Representation of graphs

Two common ways: *adjacency matrix* and *adjacency list*.
Describe them. (This is covered in 30 or 40, which many students have had, although certainly not everyone.)

4. Paths in graphs

Def: A *path* $p=(v_1, \ldots, v_n)$ in $G = (V,E)$ is a sequence of vertices s.t. $\{v_i,v_{i+1}\}\in E$ for all i in $\{1,\ldots, n-1\}$.

A path is said to *contain* the vertices and to *contain* the edges $\{v_i,v_{i+1}\}$.
The *length* of a path is the number of **edges** on it.

A *cycle* is a path of length three or more that starts and ends at the same vertex and includes no repeated edges.

A graph is *acyclic* if it contains no cycle.

A graph $G = (V,E)$ is *connected* if, for all x,y in V, there is a path from x to y.

The *components* of a graph are the maximal connected *subgraphs*.
(Graph $G' = (V',E')$ is a *subgraph* of graph $G = (V,E)$ if $V' \subseteq V$ and $E' \subseteq E$.)

Alternative definition of components: Say that $x \sim y$ (these vertices are in the same component) if there is a path from x to y. **Prop**: this is an equivalence relation. Its blocks (equivalence classes) are the components.

Alternative definition of a component: the component containing v is all vertices connected to v by paths of any lengths; and all the induced edges (the edges of the original graph that span vertices in the component).

Describe an algorithm, based on DFS, for counting the number of components of a graph and identifying them.

5. Trees

Def: A *tree* is a connected acyclic graph.

Proposition: In any tree, $m = n - 1$

Proof: By induction on n. True when n=1.
Now suppose G is an n-vertex tree, $n>1$.
Claim that there is some node of degree 1 in G.
If any node of degree 0: contradicts connected.
If all nodes of degree ≥ 2, contradict acyclicity.
Take your node of degree 1 and remove it, along with the adjoining edge.
What remains is connected and acyclic, so $(m-1) = (n-1) - 1$ for it.
Took away one edge and one vertex, so $m=n-1$.

6. Eulerian and Hamiltonian graphs

Def: A graph G is *Eulerian* if it there is a cycle C in G that goes through every **edge** exactly once.

A graph G is *Hamiltonian* if there is a cycle that goes through every **vertex** exactly once.

Theorem: (Euler) A connected graph $G = (V,E)$ on $n \geq 3$ vertices is Eulerian

iff

every vertex of G is of even degree.

Proof: ➔ Choose s. Graph is Eulerian mean there is a path that starts at s and eventually ends at s, hitting every edge. Put a label of 0 on every vertex. Now, follow the path. Every time we exit a vertex, increment the label. Every time we enter a vertex, increment the label.
At end of traversing the graph, label(v) = deg(v) and this is even.

🡐 (sketch) If every vertex is of even degree, at least three vertices. Start at s and grow a cycle C of unexplored edges until you wind up back at s. You never "get stuck" by even-degree constraint. If every edge explored: Done. Otherwise, find contact point of C and an unexplored edge (exists by connectedness) and grow out from there. Splice together the paths.

So there is a trivial algorithm to decide if G is Eulerian: just check if all its vertices are of even degree.

Amazing fact: There is no "reasonable" algorithm known to decide if a graph is Hamiltonian.
(Most computer scientists believe that no such algorithm exists.)

Theorem [Bondy- Chvátal, 1972] Let $G = (V,E)$ be a graph on $n \geq 3$ vertices and let x and y

be *non*adjacent vertices s.t. $\deg(x)+\deg(y) \geq n$. Then
G is Hamiltonian $\Leftrightarrow$ $G \cup \{x,y\}$ is Hamiltonian

Restatement: G is Hamiltonian $\Leftrightarrow$ $\mathrm{cl}(G)$ is Hamiltonian

Where $\mathrm{cl}(G)$ is the closure of G with respect to repeatedly adding vertices $\{x,y\}$ iff $\deg(x)+\deg(y) \geq n$

Proof: ➔ Obvious

🡐 Suppose
$G \cup \{x,y\}$ is Hamiltonian, $\deg(x)+\deg(y) \geq n$
but G is *not* Hamiltonian

Since $G \cup \{x,y\}$ is Hamiltonian we know at least that there's a Hamiltonian **path**

$$x = v_1\, v_2 \ldots v_n = y$$

in G. **If** x is adjacent to v_i for $i \geq 2$
then v_i is **not** adjacent to y, for if there were an $\{v_i, y\}$ edge then we have a HC in G (draw the picture).
Said differently, every time you have an **edge** coming out of x you have a **nonedge** coming out of y.
So y has a **potential** of $n-1$ edges coming out of it, but the actual number will be less than this by at least $\deg(v)$:

$$\deg(y) \leq (n-1) - \deg(x)$$

i.e, $\deg(x) + \deg(y) < n$. Contradiction.

Corollary: Assume G is a graph on ≥ 3 vertices.
If cl(G) is a clique then G is Hamiltonian

Corollary [Dirac's theorem] Assume G is a graph on \ge 3 vertices.
If $\deg(v) \geq n/2$ for all v, then G is Hamiltonian.

7. Longest and shortest paths

Def: A *shortest path* between two vertices x and y is a path from x to y such that there is no shorter (=fewer edges) path from x to y.

A *longest path* between two vertices x and y is a *simple path* (=no repeated vertices) from x to y.

Claim: There is an efficient algorithm to identify a shortest path between two designated vertices in a graph.
(You will learn one in ecs122A or ecs60)

Amazing fact: There is no efficient algorithm known to find a longest path from x to y.
(Most computer scientists believe that no such algorithm exists.)

Diameter of a graph = the length of a longest shortest path.
Explain how an *inability* to efficiently decide if a graph is Hamiltonian implies an *inability* to find a longest path between a designated pair of vertices: namely, there is a simple path (=no repeated vertices) of length $n-1$ between x and y (where $n = |V|$) iff $G \cup \{x,y\}$ has a HC.

8. Colorability

Def: A graph $G = (V,E)$ is ***k*-colorable** if we can paint the vertices using "colors" $\{1,...,k\}$ such that no adjacent vertices have the same color. Formally, …

Def: A graph is **bipartite** if it is 2-colorable. In other words, we can partition V into (V_1, V_2) such that all edges go between a vertex in V_1 and a vertex in V_2.

Proposition: There is a simple and efficient algorithm to decide if a graph G is 2-colorable / bipartite.

Proof: Modify DFS.

Initially, all vertices are uncolored: color[v]=UNCOLORED
While there are uncolored vertices *v* in *G* do DFS(*v*,0)

Algorithm DFS(*v*,*b*)
color[*v*] = *b*
for each uncolored *w* in *N*(*v*) do DFS(*w*, 1-*b*)

Amazing fact: There is no reasonable algorithm known to decide if a graph is 3-colorable.
(Most computer scientists believe that no such algorithm exists.)

Proposition [Appel, Haken 1989] Every planar graph is 4-colorable.

9. Planar graphs

Def: A graph is *planar* if you can draw it in the plane with no crossing edges.

Amazing fact: there is a simple algorithm to decide if a graph is planar
(and, when it is, to find an embedding)

Theorem [Euler, 1736]: For every planar graph, $n - m + f = 2$
Theorem [Kuratowski, 1930]: Every nonplanar graph can be contracted to a K_5 or a $K_{3,3}$
Theorem [Fary 1948] If G is planar it can be embedded in the plane with only straight lines.

The following definitions assume graph-theory terminology, to be introduced shortly.
A clique of size n in a graph is a set of n mutually connected vertices. Denote K_n.

An independent set of size m in a graph is a set of m vertices that are mutually non-adjacent.

Clique number $\kappa(G)$ = size of a *largest* clique within G (largest subgraph that's a clique)

FACT: There is no efficient algorithm known to calculate $k(G)$.

Let $R(n,m)$ = the minimum number of vertices such that a graph of
$R(n,m)$ vertices either has a clique of size n or an independent set of size m.

Alternative version:
$R(n,m)$ = the minimum number of **vertices** such that if you red/blue color the **edges** of a graph with of $R(n,m)$ vertices, there is either a **red cliq**ue of size n or a **blue clique** of size m.

Theorem: The above is well-defined:
for every $n,m \geq 1$ there exists a smallest number $R(n,m)$ such that every graph on $R(n,m)$ vertices contains either a clique of size n or an independent set of size m.

Claim: $R(3,3)=6$

Interpretation: In any room of 6 people, there are 3 mutual friends or
3 mutual strangers.

$R(n,m)$: First, $R(3,3) \geq 6$. Draw a five pointed start surrounded by a circle.
There is no triangle and no independent set of size 3.

Second, $R(3,3) \leq 6$

Remove person 1 5 people left.
Put into two pots: friends with 1, non-friends with 1.
One has at least three people.
If three friends: Case 1: some two know each other: DONE
Case 2: no two know each other: DONE
If three non-friends: ...symmetric

Difficult Puzzle: What is the minimum number of people that must assemble in a room such that there will be at least n friends or n non-friends: $R(n,n)$

R(4,4) = 18 (Greenwood and Gleason, 1955)
R(5,5) in [43, 49] Exoo, 1989; Radziszowski 1995
R(6,6) in [102,165]

"Erdős asks us to imagine an alien force, vastly more powerful than us, landing on Earth and demanding the value of $R(5, 5)$ or they will destroy our planet. In that case, he claims, we should marshal all our computers [not computer scientists?!] and all our mathematicians and attempt to find the value.

"But suppose, instead, that they ask for $R(6, 6)$. In that case, he believes, we should attempt to destroy the aliens." Quoted by Joel Spencer