Math 7h	Professor: Padraic Bartlett

Lecture 2: Graph Theory

Week 2 — *UCSB 2014*

Mathematicians like to use graphs to describe lots of different things. Groups, electrical networks, airplane routes, and the internet itself are all objects which graphs are used to model; consequently, mathematicians have came up with several different definitions for the word "graph" itself! We state several of these definitions here:

1 The Basics

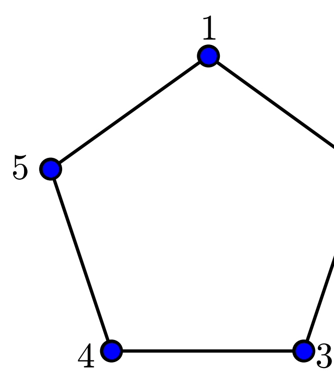
Definition. A **simple** graph G with n vertices and m edges consists of the following two objects:

1. a set $V = \{v_1, \ldots v_n\}$, the members of which we call G's **vertices**, and
2. a set $E = \{e_1, \ldots e_m\}$, the members of which we call G's **edges**, where each edge e_i is an unordered pair of distinct elements in V, and no unordered pair is repeated. For a given edge $e = \{v, w\}$, we will often refer to the two vertices v, w contained by e as its endpoints.

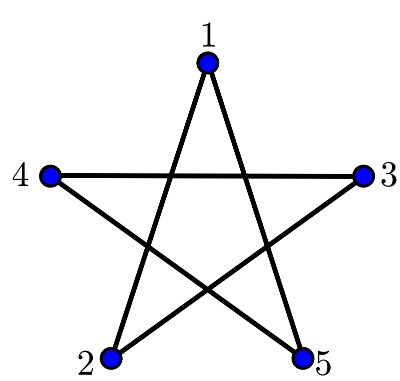
Example. The following pair (V, E) defines a simple graph G on five vertices and five edges:

- $V = \{1, 2, 3, 4, 5\}$,
- $E = \{\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 5\}, \{5, 1\}\}$.

Something mathematicians like to do to quickly represent graphs is **draw** them, which we can do by taking each vertex and assigining it a point in the plane, and taking each edge and drawing a curve between the two vertices represented by that edge. For example, one way to draw our graph G is the following:



However, this is not the only way to draw our graph! Another equally valid drawing is presented here:



As mentioned before, there are other possible definitions of a graph:

Definition. A **simple directed** graph G with n vertices and m edges consists of the following two objects:

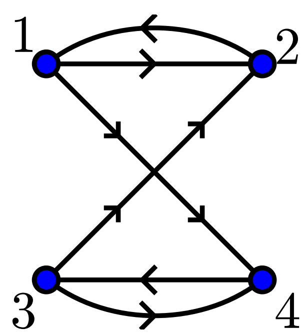
1. a set $V = \{v_1, \ldots v_n\}$ of vertices, and
2. a set $E = \{e_1, \ldots e_m\}$ of edges, where each edge e_i is an **ordered** pair of distinct elements in V, where no ordered pair is repeated.

The only difference between this definition and the definition for simple unordered graphs is that all of our edges have an ordering – i.e. the edge (a, b) is different from the edge (b, a).

Example. The following pair (V, E) defines a simple directed graph G on four vertices and six edges:

- $V = \{1, 2, 3, 4\}$,
- $E = \{(1, 2), (2, 1), (3, 4), (4, 3), (1, 3), (4, 2)\}$.

We can draw such a graph in the exact same method as before, provided that we put little arrows on our edges to indicate which direction they're traveling:



As mentioned before, there are other possible definitions of a graph:

Definition. A **multigraph** graph G with n vertices and m edges consists of the following two objects:

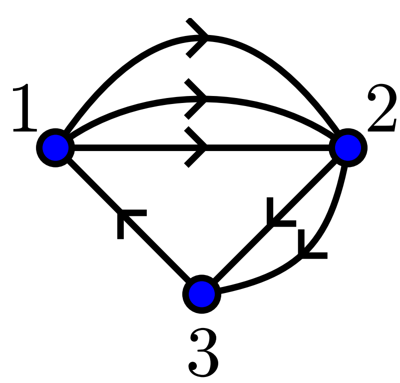
1. a set $V = \{v_1, \ldots v_n\}$ of vertices, and
2. a set $E = \{e_1, \ldots e_m\}$ of edges, where each edge e_i is an **ordered** pair of elements in V.

Note that in this definition we allow edges to be repeated, and furthermore that we allow an edge to contain the same element twice.

Example. The following pair (V, E) defines a multigraph graph G on three vertices and six edges:

- $V = \{1, 2, 3\}$,
- $E = \{(1, 2)_1, (1, 2)_2, (1, 2)_3, (2, 3)_4, (2, 3)_5, (3, 1)_6\}$.

We can realize this graph as the following picture:



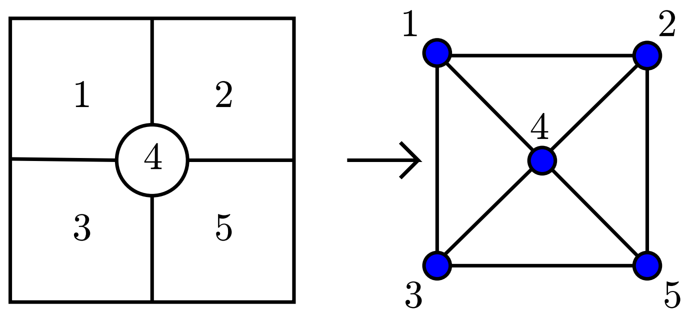
Typically, when a mathematician simply refers to a graph, they will mean a **simple** graph. Throughout this course, we will usually work with simple graphs; if we want to refer to any of the other concepts for graphs, we will explicitly say that we're doing so.

2 Graphs as Models

Before we start delving into the theory, we first show some of the ways in which graphs can model some remarkably interesting problems:

Example. (Maps.) Suppose we start with a map M consisting of several countries, and we want to find a way of assigning each country a color so that no two countries sharing a border are the same color. How many colors do we need to do this? Well, consider the following simple graph we can make out of our map, where we set

- $V =$ the collection of countries on our map, and
- $E = \{\{a, b\} : a$ and b are countries that share a border.$\}$



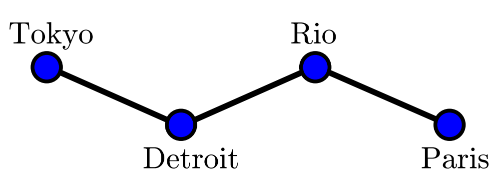
Then, our question is the following: given any graph that we can get from a map, how many colors do we need to color its vertices so that no edge connects two vertices of the same color?

The answer to this question is 4, and the problem itself is known as the **Four-Color Theorem** – proven in 1976, it is one of the first problems in mathematics to have been resolved with a computer. To this day, there are no non-computer-aided proofs known of the four-color theorem.

Example. (Travel.) Suppose you're a travelling salesman, going between cities in the country to sell your product. Specifically, suppose you have a list of C cities that you have to make it to, F a list of flights between these cities, and you're starting in some city $c \in C$. Is there some flight path you could take that would make it so you never had to visit any city twice?

Well, this clearly depends on your lists C and F, and where you're starting from. For example, if your cities were {Tokyo, Detroit, Rio, Paris} and your flights were {{ Tokyo, Detroit}, { Detroit, Rio}, {Rio, Paris }}, you could do this if you were starting in Tokyo or Paris, but not if you started in Detroit or Rio.

A natural way to visualize this problem is with graph theory! Specifically, if we think of C as the set of vertices and F as the set of edges, we can visualize such a travel map as follows:



Phrased in this fashion, our question is the following: given a graph G, is there a **path**[1] that visits every vertex exactly once?

Such a path is called a Hamiltonian path, and the question of whether such paths exist on an arbitrary graph is NP-complete[2].

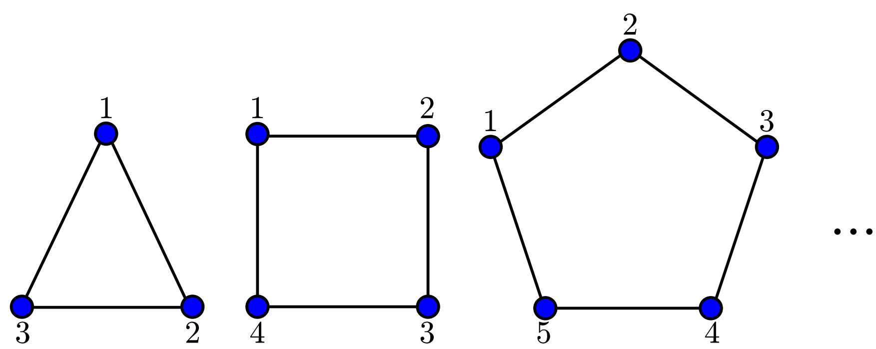
[1]A **path** of length n is a sequence of alternating vertices and edges $v_0, e_{01}, v_1, e_{12}, \ldots v_n$ from our graph G, so that each edge $e_{k,k+1}$ connects the vertex v_k to the vertex v_{k+1}. Intuitively, a path is just a way of "walking around" on our graph for n steps.

[2]This means, roughly, that there is no "fast" way to find out whether such a path exists; basically, there isn't any way to find out if a graph has a Hamiltonian path that's much faster than just trying out *every* path and seeing if any of them work. Furthermore, if you *could* find a faster algorithm in general, the fact that this problem is NP-complete would allow you to solve a ton of other really difficult problems (like how to factor numbers into primes) quickly as well. See Wikipedia for a better description of what NP-complete means.

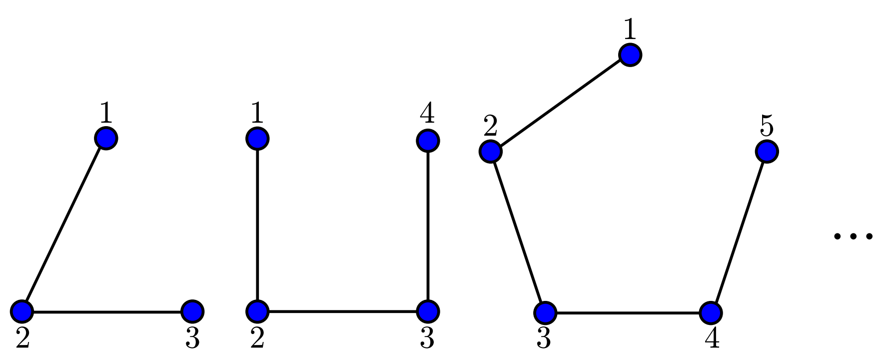
3 Several Key Graphs

In the field of graph theory, there are several graphs which come up enough to where we've given them names. We list them here, along with some of their properties:

- **The cycle graph C_n.** The cycle graph on n vertices, C_n, is the simple graph on the vertex set $\{v_1, v_2, \ldots v_n\}$ with edge set $E(C_n) = \{\{v_1, v_2\}, \{v_2, v_3\}, \ldots \{v_{n-1}, v_n\}, \{v_n, v_1\}\}$. The cycle graphs C_n can be drawn as n-gons, as depicted below:

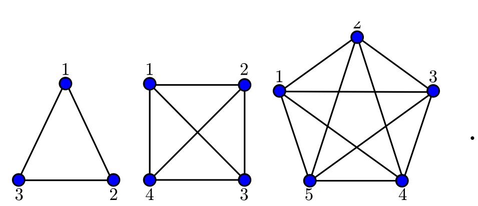


- **The path graph P_n.** The path graph on n vertices, P_n, is the simple graph on the vertex set $\{v_1, v_2, \ldots v_n\}$ with edge set $E(C_n) = \{\{v_1, v_2\}, \{v_2, v_3\}, \ldots \{v_{n-1}, v_n\}\}$. The path graphs P_n can be drawn as paths of length n, as depicted below:



  Every vertex in a P_n has degree 2, except for the two endpoints v_1, v_n, which have degree 1. P_n contains $n-1$ edges.

- **The complete graph K_n.** The complete graph on n vertices, K_n, is the simple graph on the vertex set $\{v_1, v_2, \ldots v_n\}$ that has every possible edge: in other words, $E(K_n) = \{\{v_i, v_j\} : i \neq j\}$. We draw several of these graphs below:



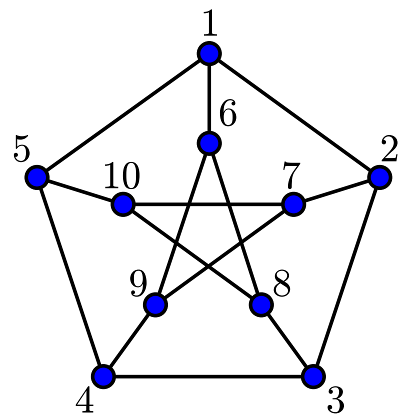
  Every vertex in a K_n has degree $n-1$, as it has an edge connecting it to each of the other $n-1$ vertices; as well, a K_n has $n(n-1)/2$ edges in total in it, by the degree-sum formula. (Explicitly: every vertex has degree $n-1$ and there are n vertices, therefore the sum of the degrees of K_n's vertices is $n(n-1)$. We've shown that this quantity is twice the number of edges in the graph; dividing by 2 then tells us that the number of edges in K_n is $n(n-1)/2$, as claimed.)

- **The complete bipartite graph $K_{n,m}$.** The complete bipartite graph on $n+m$ vertices with part sizes n and m, $K_{n,m}$, is the following graph:

  - $V(K_{n,m}) = \{v_1, v_2, \ldots v_n, w_1, w_2, \ldots w_m\}$.
  - $E(K_{n,m})$ consists of all of the edges between the n-part and the m-part; in other words, $E(K_{n,m}) = \{(v_i, w_j) : 1 \leq i \leq n, 1 \leq j \leq m\}$.

  The vertices v_i all have degree m, as they have precisely m edges leaving them (one to every vertex w_j); similarly, the vertices w_j all have degree n. By either the degree-sum formula or just counting, we can see that there are nm edges in $K_{n,m}$.

- **The Petersen graph** P The Petersen graph P is a graph on ten vertices, drawn below:



The vertices in P all have degree three; by counting or the degree-sum formula, P has 15 edges.

4 Colorings of Graphs

So: one things mathematicians like to do with graphs is **color them**! We define what we mean by "coloring" here:

Definition. We say that a graph G is k**-colorable** if we can assign the colors[3] $\{1, \ldots k\}$ to the vertices in $V(G)$, in such a way that every vertex gets exactly one color and no edge in $E(G)$ has both of its endpoints colored the same color. We call such a coloring a **proper coloring**, though sometimes where it's clear what we mean we'll just call it a coloring.

For a fixed graph G, if k is the smallest number such that G admits a k-coloring, we say that the **chromatic number** of G is k, and write $\chi(G) = k$.

To illustrate how this definitions goes, we work a few examples:

1. K_n: The complete graph on n vertices has chromatic number n. To see that it is at least n, simply paint each of the vertices $\{v_1, \ldots v_n\}$ of $V(K_n)$ a different color (say, v_i is painted i;) then every edge trivially has two endpoints of different colors. To see that this is necessary, take any proper coloring of K_n, and look at any vertex v_i: because it's connected to every other vertex, it cannot be the same color as any other vertex (and therefore must have a different color than every other vertex, which forces n colors.)

2. Edgeless graphs: If a graph G has no edges, its chromatic number is 1; just color every vertex the same color. These are also the **only** graphs with chromatic number 1; any graph with an edge needs at least two colors to properly color it, as both endpoints of that edge cannot be the same color.

3. The pentagon: you can prove (do so!) that the pentagon cannot be colored with only two colors. In fact, the chromatic number of the pentagon is 3: simply color its vertices R, G, R, G, B in order by walking around the perimeter of the pentagon. (In fact, this same idea can be used to show that any cycle of length $2k+1$ is 3-colorable: we know that these are not bipartite, and that they do admit 3-colorings via the $R, G, R, G \ldots R, G, B$-coloring described above.)

Ok! So: at this point in time, we've finally covered enough material to talk about what I want this class to focus on: the **unit distance graph**!

5 The Unit Distance Graph Problem

Definition. Consider the following method for turning $\mathbb{R}^2$ into a graph:

- Vertices: all points in $\mathbb{R}^2$.

[3] By "color," we just mean a collection of distinct labels, like (say) natural numbers. Actual colors have the disadvantage of being finite in number, which is rather pesky.

- Edges: connect any two points (a, b) and (c, d) iff the distance between them is exactly 1.

This graph is called the **unit distance graph.**

Visualizing this is kinda tricky — it's got an absolutely insane number of vertices and edges. However, we can ask a question about it:

Question. How many colors do we need in order to create a proper coloring of the unit distance graph?

So: the answer isn't immediately obvious (right?) Instead, what we're going to try to do is just **bound** the possible answers, to get an idea of what the answers might be.

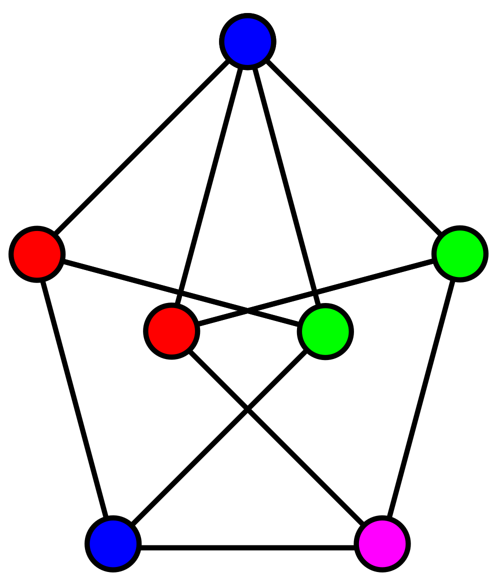
How can we even bound such a thing? Well: to get a lower bound, it suffices to consider finite graphs G that we can draw in the plane using only straight edges of length 1. Because our graph on $\mathbb{R}^2$ must contain any such graph "inside" of itself, examining these graphs will give us some easy lower bounds!

So, by examining a equilateral triangle T, which has $\chi(T) = 3$, we can see that

$$\chi(\mathbb{R}^2) \geq 3.$$

This is because it takes three colors to color an equilateral triangle's vertices in such a way that no edge has two endpoints of the same color.

Similarly, by examining the following pentagonal construction (called a Moser spindle,)



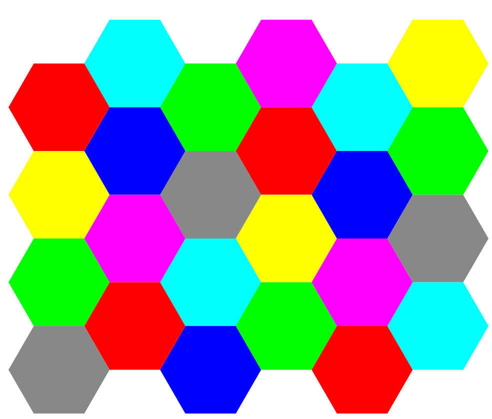
we can actually do one better and say that

$$\chi(\mathbb{R}^2) \geq 4.$$

Verify for yourself that you can't color this graph with three colors!

Conversely: to exhibit an upper bound on $\chi(\mathbb{R}^2)$ of k, it suffices to create a way of "painting" the plane with k-colors in such a way that no two points distance 1 apart get the same color.

So: consider the following way to color the plane!



To be specific: start by tiling the plane with hexagons of diameter slightly less than 1. Then, color the hexagons with seven colors as described above; i.e. repeat the color pattern gray, red, teal, yellow, blue, green, magenta

on each strip of hexagons, shifted two colors over for each strip. This gives you a mesh of hexagons, so that any two hexagons of the same color are at least more than distance 1 apart. Therefore, any line segment of length 1 cannot bridge two different hexagons of the same color! As well, because the hexagons have diameter slightly less than one, no line segment of length 1 can lie entirely within a hexagon of the same color. Therefore, there are no line segments of length 1 with both endpoints of the same color!

In other words, we have just proven that this is a proper coloring of the plane! So we can color the plane with seven colors: i.e. we just showed that

$$\chi(\mathbb{R}^2) \leq 7.$$

These bounds on $\chi(\mathbb{R}^2)$ were not too crazy to find: it took us no more than seven pages to get here, starting from the basic definition of a graph! As a result, we might hope that completely resolving this question is something we could easily finish within a few more pages.

Surprisingly: the answer is no! This problem – often called the Hadwiger-Nelson problem in graph theory literature – has withstood attacks from the best minds in combinatorics since the 1950's, and is still open to this day. We know no better bounds than the ones we've just presented.

Find some?