Graduate Graph Theory

KNU Math 848

Classnotes

v. June 4, 2025

These notes are for a graduate level introduction to Graph Theory. They draw largely from Diestal's text [2] Graph Theory, West's text [4] Introduction to Graph Theory, and Devos' class notes [1]. The source for Section 11 is Hell and Nešetřil's text Graphs and Homomorphisms [3].

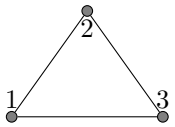
1 Introduction to Graph Theory

Definition 1.1. A graph G consists of a non-empty set $V(G)$ of *vertices* and a set $E(G) \subset 2^{\binom{V}{2}}$ of 2-element subsets of V, called *edges*. Where it is clear, we write V for $V(G)$ and E for $E(G)$.

Example 1.2. The graph K_3 with vertex and edge sets

$$V = \{1, 2, 3\} \qquad E = \{\{1, 2\}, \{2, 3\}, \{1, 3\}\}$$

is often called a triangle. We can draw it like this.



This already looks like a notational nightmare. Unless it is unclear, we will usually write uv for $\{u, v\} \in E$, and write $u \sim v$ to show that uv is in E. We read $u \sim v$ as 'u and v are *adjacent*' but we might also say that 'u and v are *neighbours*'. Sometimes we refer to uv simply as an edge e, in which case, u and v are the *endpoints* of e.

Note

One can generalise the idea of a graph to

- *digraphs* by making edges ordered sets: uv and vu are different,
- *multigraphs* by making the edgeset E a multiset,
- *hypergraphs* by allowing edges to contain more than two vertices,
- *non-simple graphs* by allowing edges of the form uu.

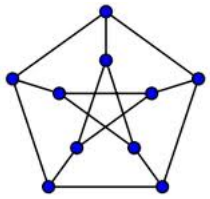
We mostly don't consider these generalisations.

Two graphs G and G' are *isomorphic* if there is a bijection $f : V(G) \to V(G')$ such that f and f^{-1} both preserve adjacency. We usually only care about graphs upto their isomorphism class. So when we draw them, we often omit vertex names.

Deciding if two graphs are isomorphic is a pretty hard problem.

1.1 Some Common Graphs

- The complete graph, or clique, K_n: $V(K_n) = [n]$ and $E(K_n) = 2^{\binom{[n]}{2}}$.
- The n-cycle C_n: $V = \mathbb{Z}_n$ and $E = \{\{i, i+1\} \mid i \in \mathbb{Z}_n\}$.
- n-path P_n: $V = \{0, 1, \ldots n\}$ and $E = \{\{i, i+1\} \mid i = 0, \ldots, n-1\}$.
- A graph is *bipartite* if there is a partition $V(G) = A \cup B$ of its vertices so that all edges are of the form ab for $a \in A$ and $b \in B$.
- The complete bipartite graph $K_{m,n}$: where $A = \{a_1, \ldots a_m\}$ and $B = \{b_1, \ldots, b_n\}$, $V = A \cup B$ and $E = A \times B$.
- The n-cube Q_n: $V = \{0,1\}^n$, E: $u \sim v$ if u and v differ in exactly one coordinate.
- The Petersen Graph:



The following is useful in proving things about the Petersen graph.

Problem 1.1. Show that for any vertex u and any other vertex v of the Petersen graph P, there is an *automorphism* f (that is, an isomorphism $f : P \to P$) such that $f(u) = v$. Graphs with this property are called *vertex transitive*.

Problem 1.2. How many edges do $K_n, K_{m,n}$ and Q_n have?

1.2 Subgraphs and Complements

Given a graph G, another graph H is a *subgraph* of G, written $H \leq G$ if $V(H) \subset V(G)$ and $E(H) \subset E(G)$. A subgraph H of G is *spanning* if $V(H) = V(G)$. A subgraph H of G is a *proper* subgraph of G if $E(H)$ is a proper subset of $E(G)$. For a subset $S \subset V(G)$, the subgraph $G[S]$ of G *induced by* S is the inclusion maximum subgraph of G with vertex set S. A subgraph H of G is *induced* if $H = G[S]$ for some subset $S = V(H)$ of G; in this case we write $H \leq_{Ind} G$.

Problem 1.3. Complete: Every graph is a spanning subgraph of a clique. A graph is bipartite if and only if it is the spanning subgraph of a _____.

Problem 1.4. Show that an alternative definition for a subgraph H of G to be induced is that $E(H) = \{uv \in E(G) \mid u, v \in V(H)\}$.

A subgraph of G isomorphic to K_n, C_n or P_n is called, respectively, an n-clique, n-cycle, or n-path of G.

A graph G is *connected* if for every two vertices u and v of G there is a $(u, v)-$path in G: a path in G with u and v as endpoints. A maximally connected subgraph of G is a *component*.

The *complement* $\overline{G}$ of a graph G is the graph defined by $V(\overline{G}) = V(G)$ and $E(\overline{G}) = \overline{E(G)}$, (where the universe used for complementation is the set $\{uv \mid u \neq v \in V(G)\}$ of all possible edges on $V(G)$).

Problem 1.5. Show that a subgraph H of G is induced if and only if $\overline{H}$ is a subgraph of $\overline{G}$.

Problem 1.6. For a graph G show that either G or $\overline{G}$ is connected.

Problem 1.7. Let G contain a cycle C and let there be a path of length at least k between vertices of C. Show that G contains a cycle of length at least $\sqrt{k}$.

(This seems trivial when you first look at it, but the path can intersect the cycle, so there is something to do.)

2 Degrees and Degree Sequences

The *degree* $\mathrm{d}(v)$ of a vertex of G is the number of edges it is in. Its *degree sequence* is the sequence of the degrees of its vertices, ordered in non-increasing order.

The minimum degree, $\delta(G)$, and the maximum degree, $\Delta(G)$, of G are the minimum and maximum degrees respectively in its degree sequence. If $\delta(G) = \Delta(G) = k$ then G is *k-regular*.

Here is a simple observation about degree sequence:

Theorem 2.1. *If $(d_1, d_2, \ldots, d_n)$ is the degree sequence of a graph, then the degree sum: $\sum_{i=1}^{n} d_i$ is equal to $2|E(G)|$.*

Corollary 2.2. *In any graph G, there is an even number of vertices of odd degree.*

A sequence $D = (d_1, d_2, \ldots, d_n)$ of non-increasing integers is *graphic* if there is a graph G on the vertices $v_1, \ldots, v_n$ such that $d(v_i) = d_i$ for all $i = 1, \ldots, n$. Such a graph G is called a *realisation* of D. We saw above that one necessary condition for a sequence to be graphic is that it sums to an even number. But this is not sufficient: $(4, 2, 0)$ is not graphic.

Problem 2.1. Show that the following condition is also necessary for a sequence $(d_1, d_2, \ldots, d_n)$ to be graphic. For all $1 \leq k \leq n$,

$$\sum_{i=1}^{k} d_i \leq k(k-1) + \sum_{i=k+1}^{n} \min(k, d_i).$$

Note

It isn't so hard to prove. Look up a 2010 paper by Tripathi, Venugopalan and West for a nice short proof.

In fact, this condition is also sufficient. But this is fair bit harder to show:

Theorem 2.3. *(Erdős-Gallai '60) A sequence $(d_1, d_2, \ldots, d_n)$ is graphic if and only if $\sum_{i=1}^{k} d_i$ is even and for all $1 \leq k \leq n$,*

$$\sum_{i=1}^{n} d_i \leq k(k-1) + \sum_{i=k+1}^{n} \min(k, d_i).$$

The Erdős-Gallai condition is what we use to decide if a sequence is graphic, but there is also a recursive method to decide it. It depends on the following idea:

Lemma 2.4. *(Havel-Hakimi) If a sequence $D = (d_1, d_2, \ldots, d_n)$ is graphic, then there is a graph with degree seqence D such that the maximum degree vertex is adjacent to the d_1 vertices of next highest degree.*

Problem 2.2. Prove this! And present it next class!

Thus to decide if a sequence D, such as $(3,3,2,2,2)$, is graphic, the *Havel-Hakimi algorithm* is as follows. While D is not all zeros,

- remove d_1, and
- remove 1 from the next d_1 degrees $d_2, \ldots, d_{d_1-1}$,

if possible.

The algorithm clearly terminates. If it terminates with an empty sequence, (or a sequence of zeros), then D is graphic. Otherwise, it is not graphic.

Starting with $(3,3,2,2,2)$, we get $(\not{3}, 3-1, 2-1, 2-1, 2) = (2,2,1,1)$, and then $(1,0,1) = (1,1)$, and then $(0,0)$. So $(3,3,2,2,2)$ is graphic.

Problem 2.3. Decide, using Havel-Hakimi, whether or not the sequence $(5,5,5,3,2,2,1,1)$ is graphic.

Problem 2.4. Show that if loops are allowed in a graph, a sequence $(d_1, d_2, \ldots, d_n)$ is graphic if and only if $\sum_{i=1}^{k} d_i$ is even. Show that if multiedges are allowed, but loops are not, then it is graphic if and only if $\sum_{i=1}^{k} d_i$ is even and $d_1 \leq \sum_{i=2}^{n} d_i$.

3 Graph Colouring

A *(proper) k-colouring* of a graph G is a function $f : V(G) \to [k]$ such that

$$u \sim v \Rightarrow f(u) \neq f(u).$$

If G has a k-colouring, it is *k-colourable*. The *chromatic number* $\chi(G)$ of G is the minimum k for which G is k-colourable. If $k = \chi(G)$, we also say that G is *k-chromatic*.

The chromatic number of a graph G is computable. Indeed, for any function $f : V(G) \to [k]$ we can check if f is a k-colouring in $O(n^2)$ time: for each vertex $v_1, \ldots, v_n$ we have to check if it gets the same colour as any of neighbour.

Note

For an algorithm on a graph, a *basic operation* is something like

- finding a vertex, or
- checking if two vertices are adjacent,

So counting the number of edges of a graph takes $2n^2$ basic operations. Big-oh notation removes constants. If an algorithm takes at most $c \cdot n^2$ basic operations for some constant c, then we say that it has running time $O(n^2)$. If the running time is polynomial in n, then we usually consider it a 'fast' algorithm. We say that the problem that it solves is 'easy'.

To decide if G has a k-colouring, we have to check at most k^n functions. To find the chromatic number of G we have to check if it has a k colouring for $k = 1$ to at most n. This full algorithm has running time $O(n^{n+2})$. Even the algorithm to decide if G has a k-colouring has running time $O(k^{n+2})$, which is exponential in n. We can do much better than this algorithm, but we have not been able to beat exponential. But some special cases are easy.

Problem 3.1. Give a characterisation of the graphs G which $\chi(G) = 1$.

Problem 3.2. Prove that for a grpah G, the following are equivalent:

i. $\chi(G) = 2$.

ii. G is bipartite.

iii. G has no odd-cycles.

Problem 3.3. Give a polynomial time algorithm to decide if a graph is 2-colourable.

Can we say something similar for 3-colouring? No. For any $k \geq 3$, it is an NP-complete problem to decide if a given graph is k-colourable. So do we give up?

Note

That a *decision problem* is NP-complete means that if one can verify a guessed solution in polynomial time, but (assuming that P $\neq$ NP) the best algorithm for solving the problem will take exponential time for infinitely many instances.
If we know a problem is NP-complete, we try to find

- restricted versions of it that are tractable (i.e., can be solved in polynomial time),
- polynomial time approximations or bounds,
- or polynomial time algorithms that solve most instances, or that solve the problem with high probability.

As deciding if a graph is 3-colourable is NP-complete, so hard. Determining the chromatic number of a graph is hard. So perhaps we try to show that for any graph the chromatic number is less than the largest degree, or show that planar graphs have low chromatic number, or give an polynomial algorithm that computes $\chi(G)$ within a factor of 2 at least 99 percent of the time.

However, there are several easy bounds on the chromatic number. The *clique number* $\omega(G)$ of a graph G is the maximum n such that K_n is a subgraph of G. The *independence number* $\alpha(G)$ is $\omega(\overline{G})$.

Problem 3.4. Show that $\chi(G) \geq \omega(G)$ and that $\chi(G) \geq n/\alpha(G)$.

Problem 3.5. Show that $\chi(G) \leq \sqrt{2m + 1/4} + 1/2$.

Note

The term *greedy* is usually used for algorithms that only try to be locally optimal, and that do not care about overall efficiency. They are often used to get loose bounds on a parameter.

As nice as this seems, determining $\omega(G)$ and $\alpha(G)$ are both NP-complete problems. So that doesn't seem to help too much. Determining the max degree $\Delta(G)$ of a graph is easy though; it takes $O(n^2)$ time.

It is easy to see that $\chi(G) \leq \Delta(G) + 1$, using the following greedy algorithm for an arbitrary ordering $v_1, \ldots, v_n$ of the vertices of G.

> **Greedy colouring:** For $i = 1 \ldots n$, colour v_i with the first colour not used on one of its already coloured neighbours.

This colouring requires at most d colours where $d-1$ is the maximum number of neighbours that any vertex has before it in the ordering. No vertex can have more than $\Delta(G)$ neighbours before it, so

$$\chi(G) \leq \Delta(G) + 1,$$

as needed.

Problem 3.6. Find a graph G and an ordering of its vertices such that the greedy colouring uses $\chi(G) + x$ colours. Show that there is an ordering of the vertices of any graph G such that the greedy colouring uses exactly $\chi(G)$ colours.

For an ordering $v_1, \ldots, v_n$ of the vertices of G, we define the *degeneracy*

$$\text{degen}_G(v_1, \ldots, v_n) = \max_{i=1}^{n} |N_G(v_i) \cap \{v_1, \ldots, v_{i-1}\}| + 1,$$

and let the *degeneracy* (or colouring number) $\text{degen}(G)$ of G be the minimum of this over all orderings of the vertices of G. The greedy colouring for an ordering $v_1, \ldots, v_n$ uses at most $\text{degen}_G(v_1, \ldots, v_n)$ colours, so are argument above actually gives

$$\chi(G) \leq \text{degen}(G) \leq \Delta(G) + 1.$$

Note

For a graph G and a subset S of the vertices of G, let the $G|_S$ be the induced subgraph of G on the vertex set S.

The degeneracy of a graph is also easy to find. Consider the following **Reverse min-degree ordering** (RMDO) of the vertices of G:

$$\text{For } i = n, \ldots, 1 \text{ let } v_i \text{ be a vertex of smallest degree in } G|_{V(G)\setminus\{v_n,\ldots,v_{i+1}\}}.$$

With this, we get the following.

Theorem 3.1. *For any graph G, we have* $\text{degen}(G) = \text{degen}_G(RMDO) = d*$ *where*

$$d^* = \max\{\delta(H) \mid H \leq_{Ind} G\} + 1.$$

Proof. We show that for any graph G the following hold:

$$d^* \leq \text{degen}(G) \leq \text{degen}_G(RMDO) \leq d^*.$$

The middle inequality is trivial as $\text{degen}(G)$ is the minimum degeneracy over all orderings of G. To see the first inequality, let H be an induced subgraph of G with $\delta(H) + 1 = d^*$, and let $v_1, \ldots, v_n$ be an ordering of $V(G)$ such that $\text{degen}_G(v_1, \ldots, v_n) = \text{degen}(G)$. Where v_i is the last vertex of H in the ordering,

$$d^* = |N_G(v_i) \cap \{v_1, \ldots, v_{i-1}\}| + 1 \leq \text{degen}_G(v_1, \ldots v_n) = \text{degen}(G).$$

For the third inequality we show that G has an induced subgraph of min degree at least $d = \text{degen}_G(RMDO) - 1$. Indeed where $v_1, \ldots, v_n$ is an $RMDO$ ordering of G and v_i is a vertex such that $|N_G(v_i) \cap \{v_1, \ldots, v_{i-1}\}| = d$ the graph $G|_{v_1,\ldots,v_i}$ has min degree d, as v_i is its min degree vertex, or would not have been chosen by the $RMDO$ algorithm from among $\{v_1, \ldots, v_i\}$. □

We can thus find the degeneracy of a graph in polynomial time by finding the degeneracy of an RMD ordering. Moreover, we get the following useful fact.

Corollary 3.2. *Every G with $\chi(G) = k$ has a subgraph H with $\delta(H) = k - 1$.*

Problem 3.7. Prove this corollary in another way by showing that any k-critcial subgraph of G must have min-degree at least $k - 1$. A subgraph of G is k-critical if it is k-chromatic, but any proper subgraph of it is $k - 1$-colourable.

Here is one of those classic theorems of graph theory.

Theorem 3.3 (Brooks '41)**.** *Let G be connected. If G is not an odd cycle or a clique, then $\chi(G) \leq \Delta(G)$.*

Proof. The theorem is trivial if $\Delta(G) \leq 2$ so assume that $\Delta = \Delta(G) \geq 3$. We proceed by induction on $n = |G|$. Let v be a vertex of G and let $H = G \setminus v$. We break the proof down into a series of claims, but are very brief on the details of the proofs of these. Please work out the details of the proofs of the claims.

Claim 3.4. *The graph $H := G \setminus v$ is Δ-colourable.*

Proof of claim. This is almost trivial. The only thing to observe is that if H contains a component that is an odd-cycle (and $\Delta = 3$) or a $(\Delta - 1)$-clique, then $\Delta(H) = \Delta - 1$. ◇

Problem 3.8. Prove the following claim. The phrase 'We are done...' means, in this case, that if this is not true then we can easily prove our result. So assume that converse and show how to extend ϕ to v. Often the phrase 'We may assume...' is used in this situation.

Claim 3.5. *We are done if v has degree less than Δ, and we are done if any under any Δ-colouring ϕ of $G \setminus v$, the neighbours of v do not get distinct colours.*

Let ϕ be a Δ-colouring of H. Label the neighbours of v as $v_1, \ldots, v_\Delta$, and assume that $\phi(v_i) = i$ for all i. Let H_{ij} be the subgraph of H induced by the vertices $\phi^{-1}(\{i, j\})$ of colours i and j.

Claim 3.6. *For $i \neq j$, we are done unless v_i and v_j lie in the same component C_{ij} of H_{ij}.*

Proof. Assume they do not, and let C be the component of H_{ij} that contains v_i. If we toggle the colours on C, that is, replace ϕ with the colouring ϕ' we get by setting

$$\phi'(x) = \begin{cases} i & \text{if } x \in C \text{ and } \phi(x) = j \\ j & \text{if } x \in C \text{ and } \phi(x) = i \\ \phi(x) & \text{if otherwise.} \end{cases}$$

Then ϕ' is a Δ-colouring and has $\phi'(v_i) = j = \phi'(v_j)$. So we are done by Claim 3.5. ◇

Problem 3.9. Show the following two claims.

Claim 3.7. *For $i \neq j$, we are done unless C_{ij} is a path from v_i to v_j.*

Claim 3.8. *For distinct i, j, k, we are done unless $C_{ij} \cap C_{jk} = v_j$.*

Now, if the vertices of $\{v_1, \ldots, v_\Delta\}$ are pairwise adjacent then G is a clique, and we are done, so wlog $v_1 \not\sim v_2$.

Let u be the neighbour of v_1 in $C_{1,2}$. Toggle the colours in C_{13} giving a new colouring ϕ' of H. We have that $\phi'(u) = \phi(u) = 2$, so defining all of the $C'_{i,j}$ with repect to ϕ', we have $u \in C'_{23}$. But u is still in a $1,2$-path from v_2, so is in C'_{12}, and so we are done by the last claim. $\square$

4 Euler's Theorem and Hamilton's Theorem

For vertices x and y of a graph G, an xy-walk is a finite sequence of vertices:

$$x = x_0, x_1, \ldots, x_\ell = y$$

such that $x_{i-1} \sim x_i$ for all $i = 1, \ldots \ell$. The length of the walk is ℓ. The walk is a *trail* if the edges are distinct, and is a *path* if the vertices are distinct, An xy-walk is *closed*, or a *circuit* if $x = y$.

Theorem 4.1. *Any xy-trail in a graph has an xy-path as a subgraph.*

Intuitively, we just start with an xy-trail and 'chop' out the repeated bits until it is a path. A lot of times it is very hard to write the intuitive proof rigorously. Try it.

Problem 4.1. Prove that any odd circuit in a graph has an odd cycle as a subgraph.

4.1 Girth and Distance

The *girth* of a graph is the length of its shortest cycle.

The *distance*, $\text{dist}(u, v)$ between two vertices u and v in a graph G is the length of the shortest uv-walk in G. Where $V = V(G)$, the *diameter* $\text{diam}(G)$ of G is

$$\max_{u,v \in V} \text{dist}(u, v),$$

and the *radius* $\text{rad}(G)$ of G is

$$\min_{u \in V} \max_{v \in V} \text{dist}(u, v).$$

A vertex u that minimises $\max_{v \in V} \text{dist}(u, v)$ is called *central*.

Problem 4.2. Show that for a graph G containing a cycle, the girth of G is at most $2\text{diam}(G) + 1$. (Hint: It might be useful to show that any closed walk that contains an edge only once, contains a cycle.)

Problem 4.3. How many vertices can a graph of radius k and maximum degree Δ have?

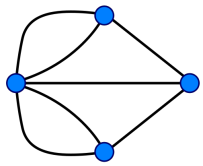
Problem 4.4. Show that for any graph G we have

$$\text{rad}(G) \leq \text{diam}(G) \leq 2\text{rad}(G).$$

4.2 Eulerian Graphs

It is often told that the first problem of graph theory was the problem of the Bridges of Königsberg. In the fictional city of Königsberg (Problem: prove or disprove my unpopular claim that Königsberg is fictional.) there were seven bridges. The dandies of the city, on a slow Sunday, would walk about the town, playing a riddlish game. Starting whereever they pleased, they were to walk around, crossing every bridge, without crossing any bridge more then once. This game went on for centuries, and several liars claimed to have done it. Fermat once wrote in the margin of a book that it was simple enough. Until eventually, Euler proved it was impossible, and stabbed Fermat in a dual. (Prove that any of this is true.)

The city of Königsburg consisted of four dots, connected by seven bridges. It looked a lot like this:



How Euler proved it is lost in the depths of historical fiction. But our story inspires the following definitions.

An *euler trail* in a graph G is a trail containing all edges and vertices of G. If it is closed, it is an *euler circuit*. A graph is called *eulerian* if it contains an euler circuit.

The following is what Euler claims to have proved.

Theorem 4.2. *A graph is eulerian if and only if it is connected, and all of its vertices have even degree.*

Proof. That an eulerian graph is connected and has all vertices of even degree is clear.

Let G be connected and have only even degrees. We show that G has an eulerian circuit. Indeed, unless G is empty, it has trails. Let

$$W = v_0, e_1, v_1, e_2, \ldots v_{n-1}, e_n, v_n$$

be the longest trail; we show that W is an Eulerian circuit.

Claim 4.3. $v_n = v_0$

Proof. If not, the vertex v_n occurs only in an odd number of edges in W. As it has even degree, there is some edge $e_{n+1} = \{v_n, v_{n+1}\}$ that is not used in W. But then W can be extended by e_{n+1}, which contradicts the choice of W. $\diamond$

Claim 4.4. *All vertices of G are in W.*

Proof. If some u is not in W, then as G is connected, there is a path P from u to some $v_i \in W$, none of whose edges are in W. But then the trail

$$u \xrightarrow{P} v_i, e_{i+1}, v_{i+1}, \dots e_n, v_0, e_1, v_1 \dots e_i, v_i$$

is a longer trail in G than W, contradicting the choice of W. ◇

Claim 4.5. *All edges of G are used in W.*

Proof. If some edge e is not used, then as W contains all vertices, $e = \{v_i, v_j\}$ for some v_i and v_j in W. We may assume that $i < j$. But then

$$v_i, e_{i+1}, \dots, v_j, e, v_i, e_i, v_{i-1}, \dots, v_0, e_n, v_n, \dots, e_{j+1}, v_j$$

is a longer trail in G than W. ◇

Thus W is an Eulerian tour of G. □

Observe that the same proof works if G is allowed to have loops or multiedges.

Problem 4.5. Show that every $4k$-regular graph has a decomposition into edge-disjoint paths of length 2.

4.3 Hamilton Paths and Cycles

This section follows West.

A path, or cycle, in G is *Hamilton* if it is spanning. A graph G is *Hamiltonian* if it contains a Hamiltonian cycle.

There were some nice conditions that characterised the eulerian graphs. We are not so lucky with Hamilton graphs. In fact, the problem of deciding if a graph on n vertices is Hamiltonian, is NP-complete. But what are some necessary or sufficient conditions for being Hamiltonian? Hamiltionian graphs clearly must be connected, and have no leaves or bridges.

We find some sufficient condtions for Hamiltonicity.

Problem 4.6. Find a Hamilton path in Q_n.

Problem 4.7. Let G be a 3-regular graph that is uniquely 3-edge-colourable (there is a unique partition of the edges into 3 parts such that no part contains edges sharing a vertex). Show that G is Hamiltonian.

Here is an early result of Dirac.

Theorem 4.6 (Dirac 1952)**.** *A graph G with n vertices is Hamiltonian if $\delta(G) \geq n/2$.*

This is implied by the slightly stronger result of Ore.

Theorem 4.7 (Ore 1960)**.** *A graph G with $n \geq 3$ vertices is Hamiltonian if every pair of non-adjacent vertices x and y satisfy the* Ore condition:

$$\mathrm{d}(x) + \mathrm{d}(y) \geq n.$$

Proof. Towards contradiction, let G be a maximal counterexample on $n \geq 3$ vertices. So G is not Hamiltonian, all pairs of non-adjacent vertices satisfy the Ore condition, and $G + xy$ is Hamiltonian for any $x \not\sim y$. (We may assume that G is not complete, as complete graphs are clearly Hamiltonian.) Let $H = G + xy$ for some $x \not\sim y$ in G. Then H is Hamiltonian, and any Hamilton cycle contains xy. So there is a Hamilton path

$$x = x_1 \sim x_2 \sim \ldots x_n = y.$$

Now if $x_1 \sim x_i$ and $x_{i-1} \sim x_n$, then

$$x_1 \sim x_2 \sim \cdots \sim x_{i-1} \sim x_n \sim x_{n-1} \sim \cdots \sim x_i \sim x_1$$

is a Hamilton cycle not using xy so in G, which is a contradiction. So for each of xs $\mathrm{d}(x)$ neighbours x_i in $\{x_2, \ldots, x_{n-1}\}$ (recall that $x \not\sim y = x_n$), x_{i-1} is not a neighbour of y. So

$$\mathrm{d}(x) + \mathrm{d}(y) = \mathrm{d}(x_1) + \mathrm{d}(x_n) \leq \mathrm{d}(x_1) + (n - 1 - \mathrm{d}(x_1)) \leq n - 1$$

This contradicts the fact that x and y satisfy the Ore condition, so the theorem is true. □

Both Dirac's and Ore's theorem give us conditions on the degree sequence of a graph that are sufficient to imply the existence of a Hamilton cycle. Are there others? There are, we look at one more generalisation, which is best possible in the sense that for any weakening of the condition (strictly in terms of the degrees) there is a non-Hamiltonian graph satisfying the weakening.

First, Bondy and Chvatal both observed that Ore's proof proves more:

Lemma 4.8. *Let G be a simple graph on n vertices, and x and y be non-adjacent vertices satisfying the Ore condition. If $G + xy$ is Hamiltonian then G is.*

This leads to the following definition.

Definition 4.9. For a graph G, the (Hamilton) closure $c(G)$ is the graph constructed by recursively adding the edge xy for non-adjacent vertices satisfying the Ore condition.

To see that this definition is well defined, assume that $G_1 = G + e_1 + \cdots + e_a$ and $G_2 = G + f_1 + \cdots + f_b$ are two closures of G under the definition. We show that all of the e_i are among the f_i. If not, there is a first e_i not in G_2. Then

$$G_1^* = (G + e_1 + \cdots + e_{i-1}) \leq G_2$$

so the degree sum of the endpoints of e_i in G_2 is greater than that in G_1^*, and so G_2 is not closed, a contradiciton.

It follows from the lemma that a graph is Hamilton if and only if its closure it. This is a nice condition, but doesn't settle the problem of deciding whether or not a graph is Hamiltonian. It is still difficult to decide if the closure is Hamiltonian. However, if we are lucky, the closure will be a clique, and then we know it is Hamiltonian. Using this idea, Chvatal showed the following.

Theorem 4.10 (Chvatal '72)**.** *Let G be a graph on $n \geq 3$ vertices, with degree sequence $D = (d_1, \ldots, d_n)$ such that for all $i < n/2$ either $d_{n-i+1} > i$ or $d_{i+1} \geq n - i$. Then G is Hamiltonian.*

Proof. Let G be a graph satisfying this condition, we show that $c(G)$ is a clique. As cliques are Hamiltonian, this is enough.

Towards contradiction assume that $c(G)$ is not a clique, so there are non adjacent vertices u and v. Where $d(v)$ denotes the degree of v in $c(G)$, let u and v be non-adjacent vertices that maximise $d(u) + d(v)$. Assume, wlog, that $m := d(u) \leq d(v)$. As $d(u) + d(v) < n$ because $u \not\sim v$, we have $m \leq n/2$ and $d(v) < n - m$. We show that $d_{n-m+1} \leq m$ and $d_{m+1} < n - m$ to contradict the assumption.

First we show that $d_{n-m+1} \leq m$. Indeed, as $d(v) < n - m$, v has at least m non-neighbours, and by choice of u and v, all have degree at most $d(u) = m$. So $d_{n-m+1} \leq d(u) = m$.

Now we show that $d_{m+1} < n - m$. Indeed, as $d(u) = m$, u has exactly $n - m - 1$ non-neighbours, which have degree at most $d(v) < n - m$. And u itself has degree at most this, so there are at least $n - m$ vertices with degree at most $n - m$. Thus $n - m > d_{n-(n-m)+1} = d_{m+1}$, as needed. □

Problem 4.8. Show that every connected graph G with $n > 2\delta(G)$ vertices has a path of length $2\delta(G)$. Deduce that every $2k$-regular graph on $4k + 1$ vertices is Hamiltonian.

5 Trees

A graph is *acyclic* or a *forest* if it has no cycles. A *tree* is a connected acyclic graph. A *leaf* of a tree is a vertex of degree one.

A *cut edge* of a graph G is any edge whose removal increases the number of components of G.

Lemma 5.1. *An edge in a graph G is a cut edge if an only if it is not in any cycles in G.*

Theorem 5.2. *A graph T is a tree if and only if it is connected and every edge is a cut edge.*

Problem 5.1.  Prove Lemma 5.1, and use it to prove Theorem 5.2.

A *spanning tree* of a graph G is a spanning subgraph which is a tree.

Theorem 5.3. *A graph G is connected if and only if it has a spanning tree.*

Proof. Let G be a graph with a spanning tree T. Let x and y be any two vertices in G. As T is spanning, they are also in T. As T is connected, there is an xy-path $P = e_1e_2 \dots e_d$ in T. But $E(T) \subset E(G)$ so P is in G as well, that is, it is an xy-path in G. This was for any $x, y \in V(G)$, so G is connected.

On the other hand assume that G is connected. Let H be a minimal connected spanning subgraph of G, that is, a connected spanning subgraph, which itself has no connected spanning proper subgraph. By minimality, every edge of H is a cut edges, so H is a tree by Theorem 5.2. □

Lemma 5.4. *Any tree has a leaf.*

Lemma 5.5. *For any tree T and any leaf v of T, the graph $T' = T \setminus \{v\}$ we get from T by removing v (and any incident edges) is also a tree.*

Theorem 5.6. *For a tree T, $|V(T)| = |E(T)| + 1$.*

Problem 5.2. Prove Lemmas 5.4 and 5.5 and use them to prove Theorem 5.6.

Theorem 5.7. *Let T be a graph with n vertices and m edges. Then the following are equivalent.*

i. T is a tree; ie. it is connected and acyclic.

ii. T has no cycles, and $m = n - 1$.

iii. T is connected, and $m = n - 1$.

iv. T is connected, but every edge is a cut edge.

v. Any two vertices of G are connected by a unique path.

Proof. From Theorem 5.2 we have the equivalence of (i) and (iv). By Theorem 5.6 we have that (i) implies (ii) and (iii). We show that (ii) $\Rightarrow$ (i) and (iii) $\Rightarrow$ (iv). This shows the equivalence of the first four statements. As an exercise, show that statement (v) is also equivalent.

(ii) $\Rightarrow$ (i): Assume that G has no cycles, so each component of G is a tree. For each component C we thus have that $|V(C)| = |E(C)| + 1$. So $n = m + c$ where c is the number of components. By $n = m + 1$, there is only one component, so it is a tree.

(iii) $\Rightarrow$ (iv): Assume that G is connected and has $n = m + 1$. If there is some edge e that is not a cut-edge, then $G' = G \setminus \{e\}$, the graph we get by removing e, is connected, and so contains a spanning tree T. Thus $n = |V(G')| = |V(T)| = |E(T)| + 1 \leq |E(G')| + 1 = m + 2$, but this is a contradiction.

□

5.1 The Minimum Spanning Tree Problem

It is not hard find a spanning tree of a connected graph G; indeed, we can find a spanning tree T with the following algorithm.

Algorithm 5.8.

i. Choose a first vertex v_0.

ii. For i from 1 to $n-1$, choose a vertex v_i from $V(G) \setminus \{v_0, \ldots, v_{i-1}\}$ with a neighbour $v \in \{v_0, \ldots, v_{i-1}\}$ and let $\{v_i, v\}$ be in T.

As G is connected there will always be some v_i in $V(G) \setminus \{v_0, \ldots, v_{i-1}\}$ with a neighbour in $\{v_0, \ldots, v_{i-1}\}$. The algorithm chooses $n-1$ edges for T, and one can check by induction that $T|_{\{v_0,\ldots,v_i\}}$ is connected for all i, so T is connected subgraph of G on n vertices having $n-1$ edges. Thus it is a spanning tree of G.

Problem 5.3. In the above algorithm, show that $T|_{\{v_0,\ldots,v_i\}}$ is connected for all i.

That was easy, but sometimes certain spanning trees are better than others.

Definition 5.9. A *weighted graph* consists of a graph $G = (V, E)$ along with a *weight function* $w : V \to \mathbb{R}$. For a subgraph G' of G the weight $w(G')$ of G' is the sum

$$w(G') = \sum_{e \in E(G')} w(e).$$

This problem can be a bit harder that simply finding any spanning tree. But it is not so much harder.

The following is Kruskal's algorithm for finding a minimum weight spanning tree T in a connected weighted graph G.

i. Order the edges of G so that $w(e_0) \leq w(e_1) \leq \cdots \leq w(e_{m-1})$.

ii. Let $e_1 \in T$.

iii. For $i = 1$ to $m - 1$ put e_i into T unless it makes a cycle.

Problem 5.4. Show that the graph T returned by the algorithm is a spanning tree.

Proposition 5.10. *The spanning tree T given by Kruskal's algorithm is a minimum weight spanning tree.*

Proof. Towards contradiction, assume that there is a spanning tree S of G with $w(S) < w(T)$. Relabel the edges of T as $\{t_1, \ldots, t_n\}$ in non-decreasing order of weight; that is, so

$$w(t_1) \leq w(t_2) \leq \cdots \leq w(t_n).$$

Similarily relabel the edges of S as $\{s_1, \ldots, s_n\}$ in non-decreasing order of weight. As $w(S) < w(T)$ there must be some i so that $w(s_i) < w(t_i)$. Fix i as the minimum such index.

Let T' be the subgraph of T consisting of the edges $\{t_1, \ldots, t_{i-1}\}$ and S' be the subgraph of S consiting of the edges $\{s_1, \ldots, s_i\}$. As both are subgraphs of trees, neither contains a cycle.

Claim 5.11. *There is an edge $s \in S'$ whose vertices are in different components of T'.*

Proof. Let $T_1, \ldots, T_r$ be the components of T'. As S' is acyclic, the number of possible edges of S' between vertices of T_j is at most $|V(T_j)| - 1$, and as T_j is a tree, the number of edges are exactly $|E(T_j)| = |V(T_j)| - 1$. So S' has at most as many edges among vertices of T_j as T does. But S' has more edges than T' overall, so some edge of S' must be between components of T'. ◇

As this $s \in \{s_1, \ldots, s_i\}$ we have $w(s) \leq w(s_i) < w(t_i)$. But then our algorithm should have chosen s instead of t_i to add to T. This is a contradiction. □

Problem 5.5. Let T be a spanning tree of G. Show that for every edge $e \in E(G)$, there is an edge $e' \in E(T)$ such that $T \cup \{e\} \setminus \{e'\}$ is also a spanning tree of G.

Problem 5.6. (A little tricky) Prim's (or Jarnik's) algorithm for finding a minimum weight spanning tree is to use Algorithm 5.8. When choosing the edge in step *ii.*, one should choose one of minimum weight. Show this yields a minimum weight spanning tree.

5.2 Cayley's Formula

How many different trees are there on n vertices?

Theorem 5.12 (Cayley's formula)**.** *There are $T_n = n^{n-2}$ different labelled trees on n vertices.*

By labelled, we mean that the path $1 \sim 2 \sim 3$ on the vertex set $\{1, 2, 3\}$ is different from the path $2 \sim 1 \sim 3$. (But it is the same as the path $3 \sim 2 \sim 1$.) So as a corollary, we get that there are between $n^{n-2}/n!$ and n^{n-2} non-isomorphic trees on n vertices.

In this section we give two proofs of Cayley's formula. The first, due to Prüffer, we only sketch.

5.2.1 Prüffer's Proof

This is a proof by bijection. Observe that the number of $(n-2)$-element sequences $s \in [n]^{n-2}$ of integers from the set $[n] = \{1, 2, \ldots, n\}$ is exactly n^{n-2}. We show a correspondence between these sequences and the labelled trees on n vertices.

Prüffer Encoding

Given a tree T_1 with vertices labelled with the integers $[n]$, construct $s \in [n]^{n-2}$ as follows.

For $i = 1, \ldots, n-2$ do

- Let ℓ_i be the lowest labelled leaf in T_i.
- Let s_i be the unique neighbour of ℓ_i in T_i.
- Let $T_{i+1} = T_i \setminus \ell_i$.

Prüffer Decoding

Given an sequence $s = (s_1, \ldots, s_{n-2})$, construct a tree T on the vertex set $[n]$ as follows. Let L_1 be empty, and let $Z_i = (s_i, s_{i+1}, \ldots s_{n-2})$ for $i = 1, \ldots, n-2$.

For i from 1 to $n-2$ do

- Let ℓ_i be lowest number not in $L_i \cup Z_i$.

- Add the edge $\ell_i s_i$.
- Let $L_{i+1} = L_i \cup \{\ell_i\}$.

When the above algortihm terminates, there are two vertices not in L_{n+1}. Add the edge between them.

Problem 5.7.  Show that Prüffer encoding and Prüffer decoding are inverse operations.

5.2.2 Pitman's Proof

This beautiful little double counting proof of Cayley's Formula by Pitman, can be found in 'Proofs from the Book' by Aigner and Zeigler. It's a nice book.

We need some definitions.

An *oriented graph* $\vec{G}$ is a graph G in which we apply a orientation, or direction, to each edge. An oriented edge is often referred to as an *arc*. We write $u \to v$ to denote that there is an arc $a = (u, v)$ from u to v. We say that u is the *source* or *start* of the arc, and v is the *terminus* or *end* of the arc. We also say a is an *in-arc* of v and an *out-arc* of u. The *in-degree* of a vertex in an oriented graph is the number of in-arcs, and the *out-degree* is the number of out-arcs. A path $P = v_1 \sim v_2 \sim \cdots \sim v_d$ is a *directed* path, directed from v_1 to v_d if it is given the following orientation: $v_1 \to v_2 \to \cdots \to v_d$. An oriented cycle is directed if every vertex has in-degree one and out-degree one.

A *rooted tree* is a tree in which there is a particular vertex designated as the *root*.

We also need some easy observations. Given a tree T, choosing a root r defines a unique orientation on the tree in which for every vertex $u \neq r$ the unique path between u and r in T is a directed path from u to r. As such, we call an oriented tree a *rooted tree* if there is some vertex r with such that every arc is directed towards r.

Problem 5.8.  Show that an oriented tree is a rooted tree with root r, if and only if every vertex but r has out-degree one, if and only if it has max out-degree 1.

Pitman's proof of Cayley's theorem comes from counting the possible constructions of a rooted tree in two ways.

Let F_n be the number of sequences $(a_1, a_2, \ldots, a_{n-1})$ of arcs on the set of vertices $[n]$, such that the oriented graph $\vec{G}$ with vertices $[n]$ and arcs $a_1, a_2, \ldots, a_{n-1}$ is a rooted tree.

First we observe that $F_n = T_n \cdot n!$. Indeed, we choose a tree in T_n ways, and a root in n ways. The $n-1$ arcs are determined, but the order they come in in

the sequence can be now chosen in $(n-1)!$ ways. Together, this is $T_n \cdot n!$, as needed.

On the other hand we can construct a sequence as follows. Let a_1 be any arc. This can be chosen in $n(n-1)$ ways. At step i, let a_i be any arc such that the oriented graph with arcs $a_1, \ldots, a_i$ has no vertex of out-degree greater than one, and no cycles.

Observe that the arc we add at step i must go between components, or it makes a cycle. So it reduces the number of components by one. Thus before adding a_i there are $n-i+1$ components. Further, as each component has max out-degree 1 and is a tree, it has a unique vertex, its root, with out-degree 0. The arc a_i may end at any vertex v, and must then begin at the root of any component other than the component containing v. So can be chosen in $n(n-i)$ ways. This gives

$$F_n = \prod_{i=1}^{n-1} n \cdot (n-i) = n^{n-1} \cdot (n-1)! = n^{n-2} \cdot n!.$$

So $T_n \cdot n! = F_n = n^{n-2} \cdot n!$, which yields Cayley's formula: $T_n = n^{n-2}$.

5.3 The Matrix Tree Theorem

We have counted the number of trees on $[n]$ vertices. We can view each as a spanning subtree of K_n. This was hard enough for K_n. It should be harder for a general graph G on n vertices. And it is. But it turns out, that the number of spanning subtrees of G is counted by the determinant of a $(n-1)\times(n-1)$ matrix which we can get easily from the graph in n^2 time. Computing a determinant takes $O(n^3)$ so this is not a bad calculation. This is the subject of this subsection. We start by looking at a couple matrices associated to a graph.

Recall that $[x]_{m,n}$ is the $m \times n$ matrix with entry $x_{i,j}$ in the i^{th} row and j^{th} column.

So, for example, when

$$M = [m]_{3,4} = \begin{bmatrix} 3 & 4 & 5 & 2 \\ 1 & 0 & 3 & 3 \\ 0 & 2 & 1 & 7 \end{bmatrix},$$

we have $m_{1,3} = 5$.

If $L = [x]_{\ell,m}$ and $M = [y]_{m,n}$ then the product $LM = [z]_{\ell,n}$ is matrix defined by

$$z_{i,j} = \sum_{\alpha=1}^{m} x_{i,\alpha} y_{\alpha,j}.$$

It will be convenient to index the rows and columns of a matrix M by sets R and C rather than by the set $[m]$ and $[n]$. In this case we write $M = [x]_{R,C}$ to

mean that $x_{r,c}$ is the entry in row r and column c . For a subset $R' \subset R$ let $M^{R'}$ be the submatrix of M consisting of rows in R' and for a subset $C' \subset C$ let $M_{C'}$ be the submatrix of M consisting of columns in C'. We write M^r for the row vector $M^{\{r\}}$ and M_c for the column vector $M_{\{c\}}$. So the product $LM = [z]_{R,C}$ can be defined succincty by

$$z_{r,c} = L^r \cdot M_c.$$

5.3.1 The Adjacency Matrix $A = A(G)$

The *adjacency matrix* $A = A(G)$ of a graph G with vertex set V is the $n \times n$ matrix $A = [a]_{V,V}$ where

$$a_{i,j} = \begin{cases} 1 & \text{if } i \sim j \\ 0 & \text{if } i \not\sim j. \end{cases}$$

Observe that this is a symmetric matrix. The number of '1's in the v^{th} row A^v or column A_v is the degree of vertex r.

Problem 5.9.  Where $A = A(G)$, show that the (u, v) entry $(A^2)^u_v$ of $A^2 = AA$ is the number uv-walks of length 2 in G, and in particular that the (u, u) entry of A^2 is the degree of u in G. Show, more generally that the (u, v) entry of A^k is the number of uv-walks of length k in G.

5.3.2 The Incidence Matrix $M = M(G)$

The *incidence matrix* $M = M(G)$ is the $m \times n$ matrix $M = [m]_{E,V}$ such that

$$m_{e,v} = \begin{cases} 1 & \text{if } v \in e \\ 0 & \text{if } v \notin e. \end{cases}$$

Notice that where $\mathbf{1}$ is the appropriately sized vector of ones, $M\mathbf{1} = 2\mathbf{1}$. For all our applications, it will be more useful to assume an arbitrary orientation on G, and to define $M = M(G) = M(\vec{G})$ to reflect that orientation:

$$m_{e,v} = \begin{cases} -1 & \text{if } v \text{ is the start of } e \\ 1 & \text{if } v \text{ is the terminus of } e \\ 0 & \text{otherwise} \end{cases}$$

As we are not allowing loops, this is well defined, and is skew-symmetric. Moreover, we now have that

$$M\mathbf{1} = 0.$$

Infact, we have the following more general fact.

Problem 5.10. Show that $M_s\mathbf{1} = 0$ if and only if the subgraph G graph induced by S is a union of components of G. (That is, there are no edges between S and $V \setminus S$.)

Now consider the matrix equation $M\mathbf{x} = \mathbf{b}$.

The column vector $\mathbf{b}$ assigns a value $\mathbf{b}^e$ to each arc e of G. A solution $\mathbf{x}$ assigns a value $\mathbf{x}_v$ to each vertex v. Where $e = (u, v)$ the e^{th} row of this equation becomes $M^e\mathbf{x} = \mathbf{b}^e$ which expands to

$$-\mathbf{x}_u + \mathbf{x}_v = \mathbf{b}^e = \mathbf{b}^{(u,v)}.$$

Rearranging this we get that $\mathbf{x}_v = \mathbf{x}_u + \mathbf{b}^{(u,v)}$.

Problem 5.11. If T is a subtree of G then $M^{E(T)}_{V(T)}\mathbf{x} = \mathbf{b}$ has a solution for any $\mathbf{b}$.

Problem 5.12. If C is the ℓ-cycle $v_1e_1v_2e_2 \dots v_\ell e_\ell v_1$ in G then $M^{E(C)}_{V(C)}\mathbf{x} = \mathbf{b}$ if and only if $\sum_{i=1}^{\ell} \mathbf{b}^{e_i} p_i = 0$ where p_i is 1 or -1 depending on whether e_i is a forward arc (v_i, v_{i+1}) or a backwards arc (v_{i+1}, v_i).

Fix an arbitrary vertex v of G and let M_* be the $(n-1) \times (n-1$ matrix $M(G)_{V(G)\setminus\{v\}}$.

Lemma 5.13. *For a set S of $n-1$ edges of G, M^S_* has full rank if and only if S is a spanning tree of G.*

Proof. For a subset S of $n-1$ edges of G, M^S is an $(n-1) \times n$ matrix, so has rank at most $n-1$.

If S is a spanning tree, then we have by Problem 5.11 that the column-space of M^S is all of $\mathbb{R}^{n-1}$, so M^S has rank exactly $n-1$. As $M^S\mathbf{1} = \mathbf{0}$, we have that the column M^S_v is a linear combination of the other columns, and so M^S_* also has $n-1$ independent columns. Thus the the $(n-1) \times (n-1)$ matrix M^S_* has full rank.

On the other hand, if S it is not a spanning tree, then it has a cycle, and so by Problem 5.12, M^S has rank less than $n-1$. Thus M^S_* also has rank less than $n-1$. □

Problem 5.13. Show that for a set S of $n-1$ edges of G, $\det(M^S_*) = \pm 1$ if S is a spanning tree of G, and is 0 otherwise.

5.3.3 The Matrix Tree Theorem

Recall the Binet Cauchy formula for an $n \times M$ matrix A and an $m \times n$ matrix B when $n \leq m$:

$$\det(AB) = \sum_{S\subset[m],|S|=n} \det(A_S B^S)$$

Recall that the *transpose* M^T of an $m \times n$ matrix $M = [m]_{R,C}$ is the $n \times m$ matrix $[m]_{C,R}$ we get by interchanging rows and columns.

With the Binet Cauchy formula we can prove the following.

Theorem 5.14. *Given a connected graph G and a vertex $v \in V(G)$, let $M = M(\vec{G})$ for some orientation $\vec{G}$ of G and let $M_* = M_{V(G)\setminus\{v\}}$. Then the number of spanning trees of G is* $\det(M_*^T M_*)$.

Proof.

$$\begin{aligned}
\det(M_*^T M_*) &= \sum_{S\subset E,|S|=n-1} \det((M_*^T)_S M_*^S \\
&= \sum_{S\subset E,|S|=n-1} \det(M_*^S)^2 \\
&= \sum_{S\subset E,|S|=n-1} \begin{pmatrix} (\pm 1)^2 & \text{if } S \text{ is a tree} \\ 0^2 & \text{otherwise} \end{pmatrix} \\
&= \text{ the number of spanning trees of } G.
\end{aligned}$$

□

Problem 5.14. The *Laplacian matrix* of a graph G is the $n \times n$ matrix $L(G) = M^T M$ where $M = M(\mathbf{G})$ for some orientation $\mathbf{G}$ of G. Show that $L(G) = [x]_{V,V}$ satisfies

$$x_{u,v} = \begin{cases} \mathrm{d}(u) & \text{if } u = v \\ -1 & \text{if } u \sim v \\ 0 & \text{if otherwise,} \end{cases}$$

independent of the ordering of G.

Problem 5.15. Show how Cayley's theorem follows from the Matrix Tree Theorem.

6 Connectivity

A graph G is *k-connected*, for $k \geq 1$, if all vertex sets $X \subset V(G)$ for which $G \setminus X$ is disconnected have size at least k. The *connectivity* $\kappa(G)$ of a graph G is the maximum k for which it is k-connected.

Recall that the trivial graphs with 0 or 1 vertices were considered disconnected. With this in mind, we clearly have the following.

i. A graph is 1-connected if and only it is connected.

ii. The clique K_n has connectivity $n-1$– it is $n-1$ connected but not n connected.

iii. A graph has connectivity 1 if and only if it is connected and has a *cut-vertex*.

Can we describe all 2-connected graphs? Sure, they are the connected graphs without a cut-vertex. But there are other useful descriptions.

6.1 More descriptions of 2-connected graphs

In this section we give some useful descriptions of 2-connected graphs.

For distinct vertices a,b and c in a connected graph G, a *c avoiding ab-path* is an ab-path that does not contain c. The vertex c is an *ab-separator* if there are no c-avoiding ab-paths.

Problem 6.1. Prove the following.

i. A connected non-clique G is 2-connected if and only if there is no ab-separator for any two vertices a and b.

ii. If a and b are in a cycle in G then there is no ab-separator.

iii. If for every pair a, b of vertices iof G there is a cycle containing a and b, then G is 2-connected.

The converse of parts (ii) and (iii) of the above problem are also true, but their proofs are less trivial. See if you can prove them. We will do so after we prove Theorem 6.2. This shows how useful the theorem can be in proving things about 2-connected graphs, and graphs in general by breaking them up into their 2-connected bits.

Definition 6.1. For a graph G, an *ear* of a subgraph graph H is an ab-path $P = a \sim p_1 \sim \cdots \sim p_d \sim b$ in G such that $V(H) \cap V(P) = \{a, b\}$. A graph G is *ear-constructable* if it is a cycle, or is the union of an ear-constructable subgraph H and an ear of H.

Theorem 6.2. *[Whitney] A graph G is 2-connected if and only if it is ear-constructable.*

Proof. Assume that G is ear-constructable, we show that it is 2-connected. If G is a cycle then this is trivial, so assume that G can be constructed from a 2-connected subgraph H by adding an ear $P = p_0, p_1, \ldots, p_d$.

Problem 6.2. Show that G is 2-connected

On the other hand, let G be 2-connected. Then G has a cycle, and so has some maximal subgraph H constructed as above. By its maximality, H must be an induced subgraph. So, unless we are done, there is some vertex $a \in G \setminus H$ with an edge ab to H. Since G is 2-connected there is a b-avoiding ah-path P for some vertex h of H (take a minimal one). But then $b \sim aPh$ is an ear of H, and its union with H is a larger graph constructible as above. □

Problem 6.3. Using Theorem 6.2 prove the converse of 6.1 (iii).

A *block* in a graph G is a subgraph that is maximal with respect to having no cut vertices. So blocks can be maximal two connected subgraphs of G– those that we build from a cycle by adding as many ears as possible, bridges (i.e., cut edges with their end points), and independent vertices. For a connected graph G let $B_1, B_2, \ldots, B_b$ be its blocks and $a_1, \ldots, a_a$ be its cut vertices. The *block graph* $B(G)$ graph of G is the bipartite graph on $\{a_1, \ldots, a_a, B_1, \ldots, B_b\}$ in which $a_i \sim B_j$ if $a_i \in B_j$.

Problem 6.4. Show that the block graph $B(G)$ of a connected graph G is a tree. Show that for vertices a and b of G there is an ab-separator in G if and only if a and b are in different blocks in G.

Problem 6.5. Using the last two problems, prove the converse of Problem 6.1(ii).

Another way of stating the statement that G is 2-connected if and only if every pair of vertices is in a cycle, is the following. A family of ab-paths are *internally disjoint* if for any pair of paths in the family, the only common vertices are a and b.

Fact 6.3. *A graph G is two connected if and only for every pair a, b of vertices of G there are at least two internally disjoint ab-paths in G.*

We generalise this now.

6.2 Menger's Theorem

In this section we generalise the above fact, and prove Menger's Theorem: that G is k-connected if and only if for any pair a, b of vertices, there are at least k-internally disjoint ab-paths. This, follows as an exercise from the form of Menger that we actually prove. It requires some more definitions. These are important definitions in structural graph theory.

For a graph G and an edge $e = xy$ of G, the graph G/e which we get from G by *contracting* e is the graph we get by identifying x and y. So G/e has one fewer vertices than G. The number of edges of G/e is that of G less $1+|N(x)\cap N(y)|$: we lose the edge e as well as one edge for every common neighbour of x and y. We often refer to the vertex that we get by identifying x and y as v_{xy} or as $x = y$, or even just as x or y if this doesn't cause confusion.

An AB-path is X-avoiding for a set X if it contains no vertices of X. A set X is an *AB-separator* if there are no X-avoiding AB-paths. Note that if A and B share a vertex v then the length 0 path v is an AB-path, so is in any AB-separator.

A *separation* (X, G_1, G_2) of a graph G consists of a set $X \subset V(G)$ of vertices and two induced subgraphs G_1 and G_2 such that

$$V(G_1) \cap V(G_2) = X \qquad \text{and} \qquad V(G_1) \cup V(G_2) = V(G).$$

In particular, the only edges between G_1 and G_2 are induced by X.

Clearly if two sets A and B in $V(G)$ have a AB separator X with $|X| = k$ there can be at most k disjoint AB-paths. Menger showed the following.

Theorem 6.4 (Menger 1927)**.** *For vertex sets $A, B \subset V(G)$ of a graph G if the smallest AB-separator has k vertices, then G contains k disjoint AB-paths.*

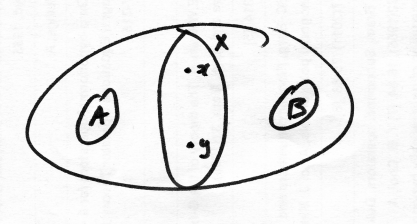
Proof. The proof is an induction on the number of edges. With no edges the result is trivial. So assume G is non-empty, and that the theorem holds for all graphs on fewer edges.

Let A and B be vertex sets in G for which the minimum AB-separator has size k. As G is non-empty it contains some edge $e = xy$. Let $G' = G/e$, and let A' and B' be the sets that A and B become after e is contracted to v_{xy}. (If x or y are in the set, then replace them with v_{xy}.)

Let X' be a minimum $A'B'$-separator in G'. By induction there are $|X'|$ disjoint $A'B'$-paths in G'. These become disjoint AB-paths in G (we may have to throw out one of x or y), so if $|X'| = k$ we are done. We may therefore assume that $|X'| < k$. Let X be the subset of $V(G)$ that we get from X' by replacing v_{xy}, if it is in X', with x and y.

The set X is an AB-separator in G so has size at least k, which means that X' has size $k-1$ and contains v_{xy}. So $x, y \in X$. We will use X to find k disjoint AB-paths in G.

The theorem also holds by induction for the graph $G'' = G - e$.



Any AX-separator in G'' is an AX-separator in G, so is an AB-separator in G, so has at least k vertices. Similarily and XB-separator in G'' has at least k vertices. By induction there are k disjoint AX-paths in G'' and k disjoint XB paths. As $|X| = k$ we can patch them together at X, and get k disjoint AB-paths. □

Problem 6.6. Show that a graph (with at least $k+1$ vertices) is k-connected if and only if for every pair (a, b) of distinct vertices the graph contains a set of k internally disjoint ab-paths.

A graph is *k-edge-connected* if for any set S of $k-1$ edges, $G \setminus S$ is connected.

Problem 6.7. Show that a graph G is k-edge-connected if and only if it contains k edge-disjoint paths between any two vertices.

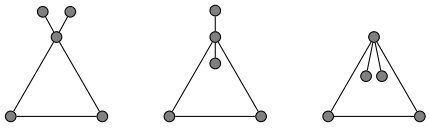
Hint Consider the *line graph* $L = L(G)$ of G. It is defined by $V(L) = E(G)$ and $ef \in E(L)$ if $|e \cap f| = 1$.

7 Planar Graphs

A graph G is *planar* if it can be drawn in the plane with no edges crossing. A drawing of the graph in the plane with no edges crossing is called a *plane graph* or an *(planar) embedding of* G.

The graph K_4 can be embedded in the plane, so is planar. But we will see that the Petersen graph cannot be. Can we decide if a graph can be embedded?

It is often useful in arguments to re-embed a plane graph– to draw it a different way. This raises the question of when two embeddings are the same, and we can answer this combinatorially or topologically.



A planar embedding of a graph G defines *faces* in the plane– maximal subsets of the planes containing no vertex or portions of an edge. Think of faces as the bits of plane you get by cutting along all the edges of the embedded graph. Topologically, faces are open sets. Their boundary is a subgraph of G. If G is 2-connected this is a cycle, and if G is connected it can be viewed as a closed walk. We often refer to this walk simply as the *face* rather than the boundary walk of the face; the size of the face is the length of the walk. In all embeddings in the above picture, there are two faces. In two of them there is one face of length 7 and one of length 3; while in the other there are two faces of length 5.

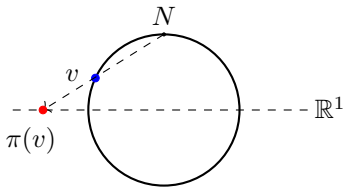
Problem 7.1. If G is a plane tree, then there is only one face. Show that its boundary walk contains each edge twice, and each vertex v exactly $\mathrm{d}(v)$ times.

A plane graph is denoted as $G = (V, E, F)$ where the face set F consists of the faces. Combinatorially, two embeddings of a graph are the same if they have the same set of faces. So the outer two embeddings in the above picture are combinatorially the same. They look a bit different though.

In an embedding of a finite graph in the plane, there is one face of infinite area, it is called the *unbounded face* or the *outer face*. The two equivalent embeddings above have different outer faces. This is often explained by viewing planar embeddings of a graph as embeddings in a sphere. Any graph embedded on the plane can be embedded on the sphere, and (certainly for finite graphs) the converse is also true. On the sphere though, the outer face is not well defined. Given a graph embedded on the sphere, we can choose any face, and embed the graph on the plane with that face as the outer face.

Recall that the n-sphere S^n is the boundary (surface) of the unit ball B^{n+1} around the origin in $\mathbb{R}^{n+1}$. The sterographic projection is the bijection π of the

punctured sphere $S^n \setminus \{N\}$ where N is the north pole $N = (0, 0, \ldots, 0, 1)$, to $\mathbb{R}^n$ that is defined by setting $\pi(v)$ to be the intersection of the line through N and v with $\mathbb{R}^n$. The following figure shows the stereographic projection of S^1 to $\mathbb{R}^1$.



Notice that ϕ maps points in the southern hemisphere of S^1 to points in the interval $(-1, 1)$ and points in the northern hemisphere to $(-\infty, -1] \cup [1, \infty)$.

Problem 7.2. Use the 2-dimensional stereographic projection to convert an embedding of a graph in $\mathbb{R}^2$ to an embedding in S^2. What happens to the unbounded face? Show that for any face of a plane graph, there is another embedding, with exactly the same faces, in which that face is the unbounded face.

Our combinatorial notion of the sameness of embeddings corresponds with them being homeomorphic embeddings on the sphere S^2. That they are homeomorphic is a concept from topology meaning that we can continuously transform one to the other always maintaining an embedding.

Using the fact that any face of an embedding can be assumed to be the outer face, you should be able to solve the following problem with ad-hoc arguments.

Problem 7.3. Show that K_5 is not planar.

Such arguments would become difficult for larger graphs, so we will look at some tools for proving this more methodically. These tools will also help answer some other obvious questions about embeddings: How different can the embeddings of a graph be? Can a graph have embeddings with different numbers of faces?

7.1 Euler's Formula and applications

Where $|V(G)| = n$ and $|E(G)| = m$ and ℓ is the number of faces. Euler showed the following fundamental identity.

Theorem 7.1 (Euler's Formula)**.** *For any connected plane graph G,*

$$n - m + \ell = 2.$$

Proof. The proof is by induction on m. As G is connected, $m \geq n-1$. If $m = n-1$ then G is a tree, so there is only one face, and the identity holds. This is our base case.

Now assume that $m \geq n$. Then there is an non-cut edge e. Remove it. Its removal joins two faces, so reduces ℓ by one, but also reduces e by one. So the identity holds from the induction hypothesis. □

Corollary 7.2. *If $G = (V, E, F)$ is a connected plane graph and $n \geq 3$, then*

$$m \leq 3n - 6.$$

Proof. Counting the number N of pairs (e, f) where f is a face in F and and e is an edge in f we get that

$$3\ell \leq N \leq 2m$$

and so

$$\ell \leq \frac{2}{3}m.$$

Rewriting $2 = n - m + \ell$ as $m - \ell = n - 2$ and then plugging in $\ell < \frac{2}{3}m$ gives

$$\frac{1}{3}m \leq n - 2$$

so $m \leq 3n - 6$ as needed.

This used that there were at least two faces, but if there is only one, then $\ell = 1 \leq \frac{2}{3}m$ because G is connected and $n \geq 3$, so $m \geq 2$. □

Problem 7.4.  Use this corollary to show that K_5 is not planar. Improve it to show that $K_{3,3}$ is not planar.

Problem 7.5.  Show that any planar graph has a vertex of degree at most 5.

Euler answers one of our questions from the introduction. For embeddings of planar graphs, the number of faces is fixed. We saw by example that the actual faces are not fixed, but Whitney showed that for 3-connected graphs they are. He showed that any two embeddings of a 3-connected graph in S^2 are homeomorphic. A lot of results given for planar graphs assume 2- or 3-connectedness. The 3-connectedness is usually so we have a unique embedding. The 2-connectedness is also useful as it means that all faces are cycles.

7.1.1 Dual Graphs

Given a plane graph G, the *dual of* G is the (general) graph G^* with vertex set

$$V(G^*) = \{v_f \mid f \text{ is a face of } G\},$$

in which the edges are defined as follows. For every edge e of G let $e^* = v_f v_g$ where f and g are the faces of G containing e. Then $E(G^*)$ is the **multiset**

$$E(G^*) = \{e^* \mid e \in E(G)\}.$$

Note that both $V(G^*)$ and $E(G^*)$ depend on the particular embedding of G. Not only is G^* a multigraph, it can have loops.

Problem 7.6. Find conditions on a graph G so that its dual G^* has no loops. Find conditions on G so that its dual, under any embedding, has no multi-edges.

The following three problems are a little tricky to do with absolute rigour, as we have not introduced the necessary topological tools. Try though, to answer them with 'schematic arguments', explaining, in the first question for example, how you draw a plane dual of each plane graph.

Problem 7.7. Show that the dual of plane graph is planar.

Problem 7.8. Show for a connected plane graph G, the dual of the dual is G. Show this isn't true for a non-connected graph.

Problem 7.9. Show that the girth of a connected plane graph is equal to the the edge-connectivity of its dual.

Theorem 7.3. *A plane graph G is bipartite if and only if its dual G^* is eulerian.*

Proof. One way is easy. Let G be a bipartite plane graph. The dual G^* is always connected, so it is enough to show that every vertex of G^* has even degree. We may assume that G has no cut-edges, as the removal of a cut-edge changes the degree of a vertex of G^* by 2, leaving its parity unchanged. For any vertex v_f of G^*, the walk around the boundary of f is a closed walk. As G is bipartite, it is even, so v_f has even degree. This was for any vertex of G^*, so G^* is euleriain.

On the other hand, assume that G^* is eulerian. We want to show that every cycle of G is even. (This is clear of the boundary of faces under a given embedding but not of every cycle.)

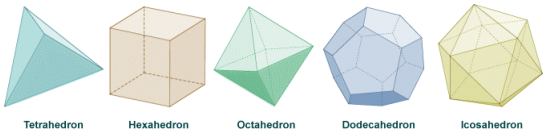
Let C be a cycle in G, and let $A \subset F(G)$ be the set of faces inside C. Any edge in any face in A is either in two faces in A or in one face and in C. Every edge of C is in one face of A. Thus C is the symmetric difference of the edge sets of faces in A. Each face has a even number of edges, so their symmetric difference does too. Thus C is even. □

Problem 7.10. Let G be a plane graph and T a spanning tree. Show that the subgraph of the dual G^* consisting of edges dual to edges in $G \setminus T$ is a spanning tree of G^*. Use this to give another proof of Theorem 7.1.

7.1.2 The Platonic Solids

Recall that a *platonic solid* is a 3-dimensional convex solid bounded by f copies of the same flat face, such that every vertex (point of intersection of more than two faces) intersects the same number of faces.

There are 5 platonic solids:



It isn't the original proof, but Euler's formula gives a nice proof that these are the only platonic solids.

The set of vertices of such a solid and the set of edges (intersections of 2 faces) make up a regular graph, and it is not to hard that this graph can be drawn in the plane so that every face has the same number of bounding edges.

Theorem 7.4. *The only connected regular graphs that can be draw in the plane with each face having the same number of edges, are the* 5 *we get from the platonic solids above, and cycles.*

Proof. Let G be a planar graph that is regular with degree d, and assume there is a planar drawing such that every face has b edges. Let f be the number of faces.

We have seen that the sum of the degrees of a graph is $2m$, so

$$2m = dn.$$

Summing the number of edges per face over all the faces, we get fb and clearly every edges was counted twice, so

$$fb = 2m.$$

By Euler's formula we then get that

$$2 = n - m + f = \frac{2m}{d} - m + \frac{2m}{b}$$

yielding

$$\frac{1}{m} = \frac{1}{d} - \frac{1}{2} + \frac{1}{b}$$

and so

$$\frac{1}{d} + \frac{1}{b} = \frac{1}{2} + \frac{1}{m}.$$

Clearly $b \geq 3$ and $d \geq 2$. If $d = 2$ then G is a cycle, so we assume that $d \geq 3$. If $b, d \geq 4$ then $\frac{1}{d} + \frac{1}{b} \leq \frac{1}{2}$ so there is no solution to above equation. This leaves five cases:

b	d	solid
3	5	dodecahedron
3	4	octahedron
3	3	tetrahedron
4	3	cube
5	3	icosahedron

□

Problem 7.11. **(This problem is hard, it won't be on the test. But if you can find a nice proof of it, I would be interested in seeing it.)** Let G be a plane graph in which every edge is coloured red or blue. For each vertex v, cyclically order the edges incident to v by how you encounter them as you travel in a circle clockwise around the vertex. (This is a cyclic ordering, so $(r, b, b, r, b) = (b, b, r, b, r)$.) Show that there is some vertex v in which all incident red edges occur consecutively. (If a vertex has 0 or 1 red edge, then they are necessarily consecutive.)

7.2 Kuratowski's Theorem

So K_5 and $K_{3,3}$ are not planar. Is the Petersen graph? That K_5 is not planar tells us that a lot of other graphs are not planar.

7.2.1 Graph Minors

For a graph G, any graph M that can be formed from G by a series of vertex deletions, edge deletions, and edge contractions, is a minor of M.

It should be pretty clear that a graph with a K_5 or $K_{3,3}$ minor is not planar. Can you think of any other graphs that are not planar? Please don't look ahead. It will ruin the surprise.

Okay. Here's the surprise.

Theorem 7.5 (Kuratowski's Theorem)**.** ***Proof not on test.*** *A graph is planar if and only if it contains no K_5 or $K_{3,3}$ minor.*

Proof. We show, by induction on $n + m$, that any $(K_5, K_{3,3})$-minor free graph is planar. The base case of $n + m \leq 4$ is trivial.

Claim 7.6. *We may assume G is connected.*

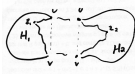
Proof. Clearly G is planar if and only if each component is, and is $(K_5, K_{3,3})$-minor free if and only if each component is. ◇

Claim 7.7. *We may assume that G is 2-connected*

Proof. Assume that G has a 1-separation H_1, H_2 with separator $\{x\}$. As G was $(K_5, K_{3,3})$-minor free so are these subgraphs, so by induction they can be embedded in the plane. Further, we can embed them with x on the outer face. Moving the copies of x together, this gives us an embedding of G. ◇

Claim 7.8. *We may assume that G is 3-connected.*

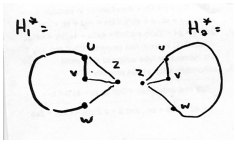
Proof. If not there is a seperator $\{u, v\}$ seperating H_1 and H_2 with $H_1 \cup H_2 = G$ and $H_1 \cap H_2 = \{u, v\}$. If $u \sim v$ then we can embed H_1 and H_2 and glue the embeddings together on $u \sim v$. So assume that $u \not\sim v$. Let $z_i \in H_1 \setminus \{u, v\}$. By Menger, there are two internally disjoint paths from z_1 to z_2; one must go through u and the other through v. These paths in H_2 imply that $H_1 + \{u, v\}$ is a minor of G and in H_1 imply that $H_2 + \{u, v\}$ is a minor. They are therefore both free of K_5 and $K_{3,3}$ minors, so can be embedded and then joined on $u \sim v$.



◇

Claim 7.9. *For all edges uv, $G' := G \setminus uv$ is 3-connected.*

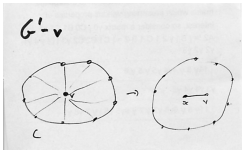
Proof. If not, there is a separator $\{v_{uv}, w\}$ of G', so a separator $\{u, v, w\} = H_1 \cap H_2$ of G. Where $z_1 \in H_1$ and $z_2 \in H_2$ there are 3 internally disjoint z_1z_2-paths. Let



be the minors they yield. They are embeddable. The vertex z has degree 3 so is in at most 3 faces, one of which contains the path uzw. The graphs can be embedded with this face as the outer face, and so can be embedded as shown in the figure. These embeddings can be patched together.

◇

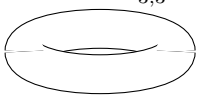
Now let xy be an edge of G and let $G' = G/xy$ where $v = v_{xy}$. Then G' has no K_5 or $K_{3,3}$ minor, so can be embedded. Further, by the claim, $G' - v = G - xy$ is 2-connected, so the boundary of the face on which v lived was cycle C.



We expand v back to xy. Let X be the neighbours of x on C and Y those of Y. If $|X| \leq 1$ or $|Y| \leq 1$ then the expansion is trivial, so assume that both contain at least 2 vertices. If $|X \cap Y| \geq 3$ we have a K_5 minor, so we may assume that $|X \cap Y| \leq 2$. If $|X \cap Y| \leq 2$ and $X = Y$ then the expansion is trivial, so we may assume there is some $x_1 \in X - Y$. Where y_r is the first element of Y we get to from x_1 in the clockwise direction and y_ℓ is the first we get in the counterclockwise direction, let P be the path of C between them that contains x_1 and P' the path that does not contain x_1. By choice of y_r and y_ℓ, P contains no elements of Y except y_r and y_ℓ; and P' can contain no element of X other than possibly y_r and y_ℓ or we have a $K_{3,3}$ minor. So we can extend our embedding.

So there is an embedding.

□

Problem 7.12. Show that $K_{3,3}$ can be drawn without crossing edges on the surface of a torus: 

8 Colouring Planar Graphs

We saw that every graph G with $\chi(G) = k$ has a subgraph H with $\delta(H) = k-1$. So we get immediately that every planar graph is 6-colourable.

Problem 8.1. Show that any planar graph is 6-colourable.

In fact, more is true:

Theorem 8.1. *(The Four-Colour Theorem [Haken-Appel 1976]) Any planar graph is* 4*-colourable.*

The proof of the Four-Colour Theorem is very long and difficult. This is slightly easier, but still interesting:

Theorem 8.2 (Heawood 1890 (based on Kempe))**.** *Any planar graph is* 5*-colourable.*

Proof. (Sketch)

- Induction on $n = |V(G)|$, so we may assume that $n \geq 6$.
- By Euler's theorem there is a vertex v of degree at most 5, and by induction $G \setminus \{v\}$ has a 5-colouring, giving its neighbours $v_1, \ldots, v_5$, embedded cyclically, distinct colours.
- There is a $\{1,3\}$-path between v_1 and v_3 or recolour á la Brooke's.
- With v this is a cycle separating v_2 and v_4.
- v_2 and v_4 cannot be in the same component of $\{2,4\}$-subgraph of G.

□

Problem 8.2. **Not on test.** This proof is based on an incorrect proof of the 4-colour Theorem by Kempe. Find the incorrect proof of the 4-colour Theorem on the internet.

Conjecture 8.3 (Hadwiger '43)**.** *Any k-chromatic graph has a K_k-minor.*

Hadwiger proved the case $k = 4$. The proof is difficult.

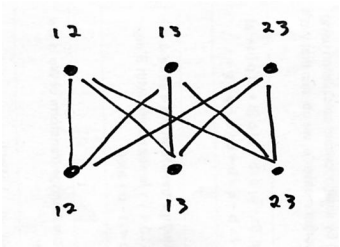
8.1 List Colourings

For a graph G a *set of lists* is a function $L : V(G) \to 2^{\mathbb{Z}}$. A function $\phi : V(G) \to \mathbb{Z}$ is a *list colouring* of (G, L) if it is a proper colouring of G and $\phi(v) \in L(v)$ for all $v \in V(G)$.

A graph G is k-list-colourable if there is a list colouring of (G, L) for any list L such that $|L(v)| = k$ for all v. The *list chromatic number* $\chi_L(G)$ of G is the minimum k for which G is k-list-colourable.

Clearly $\chi(G) \leq \chi_L(G)$ as one can take $L(v) = [k]$ for all v. And I know you think we have equality, because the lists are most restrictive when they are the same. But that isn't actually true.

Try to find a colouring of this set of lists for $K_{3,3}$



Problem 8.3. Find a bipartite graph B with $\chi_L(B) = k$ for any $k > 2$.

Thomassen proved the following.

Theorem 8.4. *Every planar graph is* 5*-list colourable.*

Proof. We actually prove the following stronger statement. Let G be a plane graph in which every face, but possibly the outer face, is a triangle. Let $v_1 \sim v_2$ be an edge on the boundary B of the outer face. Then (G, L) has a list colouring for any set of lists L satisfying

- $|L(v)| \geq 5$ for $v \notin B$
- $|L(v)| \geq 3$ for $v \in B \setminus \{v_1, v_2\}$
- $|L(v_1)| = |L(v_2)| = 1$, but $L(v_1) \neq L(v_2)$.

The proof is by induction on $n = |V(G)|$, and is clear if $n = 2$.

Case: B isn't a cycle. There is a cut-vertex $u \in B$ separating G into G_1 and G_2. We may assume that v_1 and v_2 are in G_1. By induction G_1 has an L-colouring ϕ_1. On G_2, reduce the list $L(u)$ to $\phi_1(u)$ and the list of some neighbour in B to size 1. Then the lists on G_2 meet the conditions above, so there is a colouring. This yields an L colouring of G.

Case: B has a chord uw. The set $\{u, w\}$ is a 2-separator yielding a 2-separation $G_1 \cup G_2$ of G. Again we may assume that v_1 and v_2 are in G_1, and

get that G_1 has an L colouring ϕ_1 by induction. Reducing the lists of u and w in G_2 we still get a good set of lists, so can colour it.

Case: B is an induced cycle: Let v_3 be the other neighbour of v_2 in B, and let $v_2, u_1, \ldots, u_k, v_4$ be a cyclic ordering of the neighbours of v_3. $L(v_3)$ contains two colours a and b not in $L(v_2)$. Removing v_3 from the graph and the colours a and b from the lists of $u_1, \ldots, u_k$, we can list colour $G \setminus \{v_3\}$ by induction. Replacing v_3, possibly v_4 has been coloured a or b, but at least one of these colour is available to use on v_3. □

8.2 Discharging

Discharging is a beautiful technique that has yielded many difficult results. We see how a simple example of it is used to give the following technical Lemma which is used in the in the proof of the 4-colour theorem.

Lemma 8.5. *Let G be a triangulation of the plane. Then one of the following is true*

i. G has a vertex of degree ≤ 4.

ii. G has two adjacent vertices of degree 5.

iii. G has a triangle with vertices of degrees 5 and 6.

Proof. Assuming i) is false we show that ii) or iii) hold.

Start by assigning an initial charge $f_0(v) = 3(6 - \mathrm{d}(v))$ to every vertex v. By Euler's formula we have that

$$12 = 6n - 2m = \sum_{v \in V} (6 - \mathrm{d}(v))$$

so $\sum_V f_0(v) = 36$ is the total initial charge.

Now, for every vertex v of degree 5, and every neighbour v' of v of degree at least 7, move a charge of 1 from v to v', and let $f_1 : V \to \mathbb{Z}$ be the final charge function. Clearly f_1 satisfies $\sum_V f_1(v) = 36$, so there is some vertex u with $f_1(u) > 0$. If u has degree 5, then since it started with charge 3, it has at most 2 neighbours of degree greater than 6. So either it has a neighbour of degree 5 or adjacent neighbours of degree 6.

Also u cannot have degree 6 as such vertices have intial and final charge 0.

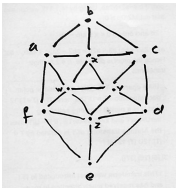
If u has degree 7, then it started with charge -3, so has at least 4 neighbours of degree 5, two of which must be adjacent.

If u has degree 8, then it stated with charge -6 so has at least 7 neighbours of degree 5, two of which must be adjacent.

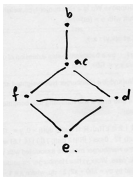
Finally, u cannot have degree $d \geq 9$ or more, because then it had initial charge $\leq -d$ or less, and this could not be made positive with d neighbours. $\square$

Using a similar Lemma, one can show that a planar graph must contain one of about 600 special subgraphs. One can than replace these with smaller subgraphs and use induction. One such examples is the following.

Problem 8.4. Let G be a planar graph containing the subgraph



show that it can be 4-coloured if the graph G' we get from it by replacing the above subgraph with the following one can be 4-coloured.



(We contract edges ax and xb, w, y and z, and add edge fd.)

9 Flows, tensions, colourings, and Duality

There are natural relations between vertex colourings of a graph, edge colours of a graph, edge colours of the dual of a graph, and vertex colourings of the dual. They can give us some pretty cool things. We follow Chapter 6 of [2].

9.1 Network flows

For a graph $G = (V, E)$ we will now view E as the symmetric binary relation $\cup_{\{u,v\}\in E}\{uv, vu\}$ on V. For an additive group H, a function $f : E \to H$ is an *edge H-map* if it is *anti-symmetric*: $f(uv) = -f(vu)$. For sets $X, Y \subset G$ we let

$$f(X, Y) = \sum_{xy\in E, x\in X, y\in Y} f(xy).$$

Notice that to define, or draw, an edge H-map f on a graph $G = (V, E)$ we usually define an orientation $\vec{E}$ of G and define f on $\vec{E}$. When we do this, we will often flip the orientation of an edge $e = uv$ to vu simplify an argument. In doing so, we switch f on that edge from $f(e) = f(uv)$ to $f(e) = f(vu) = -f(uv)$. This is fine, it defines the same edge map.

An edge-map f is Kirchoff at v if

$$f(v, V) = 0 \qquad \text{or equivalently, if} \qquad f(N^-(v), v) = f(v, N^+(v)).$$

It is a *flow* if it is Kirchoff at every vertex.

You should verify that you understand the notation by showing that for any flow f and any $X \subset V$ we have:

i. $f(X, X) = 0$,

ii. $f(X, V) = 0$, and so where $\overline{X} = V - X$,

iii. $f(X, \overline{X}) = f(X, V) - f(X, X) = 0 - 0 = 0$.

A *network* (G, s, t, c) consists of a graph $G = (V, E)$, a *source* $s \in V$, a *sink* $t \in V$, and a capacity function

$$c : E \to \mathbb{R}^+.$$

(Common restrictions are to assume that c maps to $\mathbb{Z}^+$ (is integral), or that at least one of $c(uv)$ and $c(vu)$ are 0 for every edge uv (c is directed), or that $c(uv) = c(vu)$ (c is symmetric).) We slightly change the definition of flow for networks. Given a network (G, s, t, c), a *flow* is an edge-map $f : E \to \mathbb{R}^+$ that is Kirchoff at all vertices but s and t, and that *respects the capacity*: $f(e) \leq c(e)$ for all $e \in E$. Alternately one can define a *flow f on* (G, s, t, c) as the restriction

to G of a capacity-respecting flow f^+ on the augmented network (G^+, s, t, c^+) where we add the edge ts with capacity $c^+(ts) = \infty$. A *cut* of a network is a partition $(S, \overline{S})$ of its vertices such that $s \in S$ and $t \in \overline{S}$.

Proposition 9.1. *For a flow f of a network N, every cut $(S, \overline{S})$ satisfies $f(S, \overline{S}) = f(s, V)$.*

Proof. Indeed,

$$f(S, \overline{S}) = f^+(S, \overline{S}) - f^+(s, t) = 0 + f^+(s, \bar{t}) = f(s, V)$$

where the last equality is because $f(st) = 0$. □

The value $|f| := f(s, V) = f^+(ts)$ is called the *value* of f. For each cut $(S, \overline{S})$ we have $0 = f^+(S, \overline{S}) = f^+(st) + f(S, \overline{S})$, and so $|f| = f^+(ts) = f(S, \overline{S}) \leq c(S, \overline{S})$. So the maximam value of a flow of a network is at most the minimum capacity of a cut of the network.

Theorem 9.2. *(Ford-Fulkerson's Max-flow min-cut Theorem) A network always has a flow whose value is the value of the minimum cut of the network.*

Proof. A network alway has a flow of value 0. We show for a flow f that if there is no cut $(S, \overline{S})$ with $|f| < c(S, \overline{S})$, then there is a flow with greater value.

Given a flow f, a path $s = v_0, v_1, \ldots, v_d$ is *augmenting* if $a(v_i, v_{i+1}) := c(v_i, v_{i+1}) - f(v_i, v_{i+1}) > 0$ for all i. If there is an augmenting path to t then adding $a(v_i, v_{i+1})$ to f for all i gives a new flow with great value.

So we may assume there are no augmenting path to t. Let S be the set of all vertices to which there is an augmenting path. Then $f(x, y) = c(x, y)$ for all $x \in S$ and $y \in \overline{S}$, and so

$$|f| = f(S, \overline{S}) = c(S, \overline{S}).$$

□

9.2 Tensions and Vertex H-maps

We start with another edge H-map that is different from an H-flow.

We view a cycle $C = (v_0, v_1, v_2, \ldots, v_{\ell-1}, v_0)$ in a graph H as the set of oriented edges $C = \{v_i v_{i+1} \mid i \in \{0, \ldots, \ell - 1\}\}$ where indices are modulo ℓ. For an edge H-map $\beta : E \to H$ to an additive group H, we set

$$\beta(C) = \sum_{i=0}^{\ell-1} \beta(v_i v_{i+1}).$$

An edge H-map β is a *tension* if $\beta(C) = 0$ for all cycles C of H.

For an additive group H, a *vertex H-map* of a graph G is simply a function $\alpha : V(G) \to H$. It is a *(proper) H-colouring* if adjacent vertices get different colours (images in H). An *edge H-map* of $V(G)$ is an *edge H-colouring* if edges sharing a vertex get different colours.

Problem 9.1. Find a $\mathbb{Z}$-tension of the four wheel W_4. Find one that is not 0 on every edge. Find one that is not 0 on any edge.

Hint: there is an easy recipe for finding H-tensions of G.

A vertex map $\alpha : V(G) \to H$ defines an edge map $\beta = \delta(\alpha)$ by setting

$$\beta(uv) = \alpha(v) - \alpha(u)$$

for $uv \in E$. An edge H-map β is *nowhere zero* if $\beta(e) \in H^* = H \setminus \{0\}$ for all $e \in E$. (As edge H-maps that are not nowhere zero always trivially exist, one often finds that the designation of 'nowhere zero' is often dropped in the literature. But if you find a result about the existences of an edge map– a flow or a tension– it usually means a nowhere zero edge map.)

Problem 9.2. Show the following.

Fact 9.3. *For any vertex H-map $\alpha : V(G) \to H$ of G, the map $\beta = \delta(\alpha)$ is a tension. Further, β is nowhere zero if and only if α is a proper colouring.*

Conversely, fixing the value of $\alpha(v_0) = h \in H$ for a vertex v_0 of a (connected) graph G, a H-tension β defines a vertex H-map $\alpha = \alpha_{v_1,h}$ as follows. For each vertex v other than v_1 let u be a neighbour of v on which α is defined and let

$$\alpha(v) = \alpha(u) + \beta(uv).$$

Problem 9.3. Show that because β is a tension, $\alpha = \alpha_{v_1,h}$ is well defined, and that $\alpha(v) = \alpha(u) + \beta(uv)$ for all edges uv of G.

Simplifying notation, if β is an H-tension, we let $\alpha = \partial(\beta)$ be any of these $\alpha_{v_1,x}$, where we understand that it is only defined once a colour is chosen for some vertex: $\delta(\partial(\beta)) = \beta$ and $\partial(\delta(\alpha)) = \alpha + h$ for some constant $h \in H$.

The operations ∂ and δ define a $|H|$-to-one correspondence between vertex H-maps of G and H-tensions of G, and under this correspondence, vertex H-colourings correspond to nowhere zero H-tensions.

Lemma 9.4. *A graph G has a vertex H-colouring if and only if it has a nowhere zero H-tension.*

The group structure on H doesn't play much of a role for vertex H-colourings, only the size is important.

An edge $\mathbb{Z}$-map β of G is an edge k-map if $\beta(G) \subset \{0, \pm 1, \cdots \pm (k-1)\}$. Notice that by picking the right orientation on each edge of G, a k-map can be

viewed as a map from E to the set $\{0, 1, \ldots, k-1\}$ of k elements; as such an edge $\mathbb{Z}_k$-map is exactly an edge k-map. However, summing to 0 in $\mathbb{Z}_k$ corresponds to summing to any multiple of k in $\mathbb{Z}$, so while a k-tension is a $\mathbb{Z}_k$-tension, the converse is not true. We get the following though.

Theorem 9.5. *For any group H, a graph G has a nowhere zero $|H|$-tension if and only if it has a nowhere zero H-tension.*

Proof. As a vertex H-colouring of G is exactly a vertex k-colouring of G, this is immediate from Lemma 9.4. □

9.3 Planar duality: tensions to flows

Tutte's duality, which we will see in the next section, connects $|H|$-colourings to nowhere zero H-flows. We have seen the connection of $|H|$-colourings to nowhere zero H-tensions, we look now at the connection, for planar graphs, of nowhere zero H-tensions to nowhere zero H-flows.

Recall that the dual G^* of a (two-connected) plane graph $G = (V, E)$ has a vertex v_f for every face f of G, and for every edge $e \in E$ of has an edge $e^* = \{f, f'\}$ where f and f' are the faces with e in their boundary. When G is oriented, and $e = uv$ is in $\vec{E}$, the two faces containing e become designated as the left face L or the right face R, and G^* is oriented by setting $e^* = LR$. **Dual edges go left to right.**

For an edge H-map β of G, consider the map

$$\beta^* : E^* \to H : e^* \mapsto \beta(e).$$

Clearly $(\beta^*)^* = \beta$, so we call this the dual map of β. The dual map β^* is clearly anti-symmetric, and so is an edge H-map of G^*; and is clearly nowhere zero if and only if β is. We can say more.

Theorem 9.6. *An edge H-map β of a 2-connected plane graph G is an (nowhere zero) H-tension if and only if β^* is an (nowhere zero) H-flow.*

Problem 9.4. Prove this theorem using the following lemma.

Recall that a *facial cycle* of a 2-connected plane graph G is a cycle that is the boundary of a face. In the proof of Theorem 7.3 we saw that any cycle of G is the symmetric difference of the edgesets of facial cycles. If we had time, we might have defined the *edge-space* of a graph as a free $\mathbb{Z}_2$-module generated by the edges of G. The cycles would form a subspace of this space called the *cycle-space* and observed that for a plane graph, the cycle-space is generated by the facial cycles. We could do this again in the following proof, only using orientations, our *edge-space* would now be a free $\mathbb{Z}$-module. The proof would be trivial with this setup.

Lemma 9.7. *For a 2-connected plane graph G an edge H-map β is a tension if and only if it $\beta(C) = 0$ for all face cycles of G.*

Proof. The only if part is clear. We have to show that if $\beta(C) = 0$ holds for all face cycles C, then it holds for all cycles C. For a cycle C let $C_1, \ldots, C_r$ be the boundaries of the faces on the region inside C; consider all cycles to be clockwise. This 'clockwise' is independent of the orientation of G that we might use to represent β; but given such an orientation the essential observation is that if some edge of G is in two cycles C_i and C_j then it agrees with the clockwise orientation in one, and is opposite to it in the other. Thus when we sum $\beta(C_i)$ over all cycles $C_1, \ldots, C_r$, any edge that appears in two cycles cancels itself. So we are left only with the edges that appear in only one of these cycles. These are exactly the edges of C; and so

$$\beta(C) = \sum_{i=1}^{r} \beta(C_i) = \sum_{i=1}^{r} 0 = 0$$

as needed. □

From Theorem 9.5 and Theorem 9.6, we get the following statement.

Theorem 9.8. *For a 2-connected plane graph G and a group H, the following are equivalent:*

i. G has a nowhere zero H-flow.

ii. G has a nowhere zero $|H|$-flow.

iii. The dual G^* is $|H|$-colourable.

The following shows how H-flows for groups H different than $Z_{|H|}$ can be useful. Recall that a *k-edge colouring* of a graph is an edge k-map in which no vertex has more than one edge of any colour.

Theorem 9.9. *(Tait) A plane triangulation G^* is 4-colourable if and only if its dual G is 3-edge colourable.*

Proof. With the above theorem it is enough to show that G, which is cubic, has nowhere zero $\mathbb{Z}_2 \times \mathbb{Z}_2$-flow if and only if it is 3-edge colourable. We show that a nowhere zero $\mathbb{Z}_2 \times \mathbb{Z}_2$-flow of cubic G is exactly a 3-edge colouring. Indeed as $\mathbb{Z}_2 \times \mathbb{Z}_2 \setminus \{(0,0)\}$ has exactly 3 colours it is enough to observe that in a nowhere zero $\mathbb{Z}_2 \times \mathbb{Z}_2$-flow the three edges of a particular vertex v must get distinct colours: there is no other way that three elements from the set $\{(0,1),(1,0),(1,1)\}$ can add up to 0 in $\mathbb{Z}_2 \times \mathbb{Z}_2$. □

Note that we needed that G is 2-connected in our results about nowhere zero flows so that G^* is loopless. But we loose nothing by this.

Problem 9.5. Show that under any flow $\beta : G \to H$ of a graph G, we have $\beta(e) = 0$ for any bridge (cut-edge) e of G.

In the proof of the above theorem we showed that a cubic graph G has a nowhere zero $\mathbb{Z}_2 \times \mathbb{Z}_2$-flow if and only if it has a edge 3-colouring. This part didn't used the planarity of G.

Corollary 9.10. *A cubic 3-edge colourable graph is bridgeless.*

9.4 The Tutte Flow Theorem and some Corollaries

Theorem 9.8gives the following for planar graphs; the proof used the dual. Tutte, however, showed it for all graphs.

Theorem 9.11. *For any group H, a graph G has a nowhere zero H-flow if and only if it has a nowhere zero $|H|$-flow.*

Proof. The proof for H other than $\mathbb{Z}_k$ is more difficult, and can be found in [2]. Here we only sketch the proof for $H = \mathbb{Z}_k$.

The non-trivial part is to show that a nowhere zero $\mathbb{Z}_k$-flow α gives a nowhere zero k-flow. Assume that we have a nowhere zero $\mathbb{Z}_k$-flow α, and view it as an edge k-map. It is oriented, but unless it is a k-flow, it will have vertices v such that $\alpha(v, V)$ that are non-zero multiples of k.

Let t be one with $\alpha(t, V) > 0$. Let T be the set of vertices we can get to from augmenting paths $t = v_0 \dots v_r$ from t– paths such that $\alpha(v_i v_{i+1}) > 0$. If there is $s \in T$ with $\alpha(s, V) < 0$, then take k from each edge in the augmenting path from t. As $\alpha(v_i, v_{i+1}) > 0$, this is still a k-edge map; moreover, it reduces $\alpha(t, V)$ by k, and increase $\alpha(s, V)$ by k. The proof will then follow by induction. So we show there is $s \in T - t$ with $\alpha(s, V) < 0$. Indeed, we show that

$$\begin{aligned} \alpha(T - t, V) &= \alpha(T - t, T - t) + \alpha(T - t, t) + \alpha(T - t, \overline{T}) \\ &= \alpha(T - t, t) + \alpha(T - t, \overline{T}) \end{aligned}$$

is less than 0. That $\alpha(T - t, \overline{T}) \leq 0$ is by definition of T and

$$\alpha(t, T - t) = \alpha(t, V) - \alpha(t, t) - \alpha(t, \overline{T}) > 0$$

so $\alpha(T - t, t) < 0$. □

The *flow number* $\phi(G)$ of a graph is the minimum number k such that G has a nowhere zero k-flow.

Fact 9.12. *A graph G has $\phi(G) = 1$ if and only if it has no edge.*

A graph is *even* if each component is Eulerian, iff each degree is even.

Fact 9.13. *A graph G has $\phi(G) = 2$ if and only if it is even (and non-empty).*

Proof. A graph G has $\phi(G) \leq 2$ if and only it has an nowhere zero $\mathbb{Z}_2$ flow, and a nowhere zero $\mathbb{Z}_2$-flow just assigns a 1 to every edge and says that the sum of these on each vertex is 0 modulo 2. □

Problem 9.6. Prove this for planar graphs using duality instead of using Theorem 9.5.

A cubic graph has $\phi(G) \geq 3$ as it has vertices of odd degree.

Corollary 9.14. *A cubic graph G has $\phi(G) = 3$ if and only if it is biparite.*

Proof. Given a bipartite cubic graph orient all edges from one partite set to the other, and then give each edge a 1. This is a nowhere 0 $\mathbb{Z}_3$-flow.

For the other direction, a nowhere zero $\mathbb{Z}_3$-flow puts ± 1 on each edge. Reorient the edges so all of these are 1. As the edges (with orientation) must add to 0 mod 3 at eash vertex, each vertex has all edges in or all edges out; this is a bipartition. □

As K_4 is cubic and non-bipartite it has $\phi(G) \geq 4$. It is planar though, and so its planar dual (which is also K_4) is 4-colourable; so it has a nowhere zero 4-flow. Thus $\phi(G) = 4$. However, the following is true.

Corollary 9.15. *For any even $n \geq 6$, we have $\phi(K_n) = 3$.*

Proof. To see that $\phi(K_6) = 3$ we observe that it is the disjoint union of two K_3 and a $K_{3,3}$. The two K_3 have obvious nowhere zero $\mathbb{Z}_3$-flow: direct them and put a 1 on each edge. The cubic graph $K_{s,s}$ has a nowhere zero $\mathbb{Z}_3$-flow, as shown above. Summing flows gives a flow, so K_6 has a nowhere zero $\mathbb{Z}_3$-flow.

For $n \geq 8$ we use induction, decomposing K_n into K_{n-2} and $K_2 * \overline{K_{n-2}}$. It is enough to show that the latter set of $n-2$ triangles sharing an edge e has a nowhere zero $\mathbb{Z}_3$-flow. Well each triangle has the obvious one we saw above. Adding these together might assign a value of 0 to e, but if it does, we can change the flow on owe of the triangles to -1, and this becomes non-zero.

□

A nowhere zero 4 flow colours the edges of oriented G with the colours $1, 2$ and 3. Using $\mathbb{Z}_2 \times \mathbb{Z}_2$ again allows us to get rid of orientation.

Theorem 9.16. *A graph has a nowhere zero 4-flow if and only if it is the union of two even subgraphs.*

Proof. Given a nowhere zero $\mathbb{Z}_2 \times \mathbb{Z}_2$-flow γ of a graph G we get a nowhere $\mathbb{Z}_2$-flow $\pi_1 \circ \gamma$ on the subgraph G_1 of edges e with $\gamma(e) \in \{(1,0),(1,1)\}$ and another on the subgraph G_2 of edges e with $\gamma(e) \in \{(0,1),(1,1)\}$. These graphs are both even Corollary 9.13. Their intersection has a nowhere zero $\mathbb{Z}_2$-colouring so is even, and so G is also even.

On the other hand, assume that G is the intersection of even graphs G_1 and G_2. They have nowhere zero $\mathbb{Z}_2$-colourings γ_1 and γ_2 respectivley. Then G has a nowhere zero $\mathbb{Z}_2 \times \mathbb{Z}_2$ colouring $\gamma(e) = (\gamma_1(e), \gamma_2(e))$ where $\gamma_i(e) = 0$ if $e \notin G_i$.

□

Tutte made several conjectures about flows.

Conjecture 9.17. *(Tutte)*

i. Every bridgless graph has a 5*-flow.*

ii. Every bridgless graph without without a Petersen graph minor has a 4*-flow.*

iii. Every graph without a 1 *or* 3 *edge cutset has a* 3*-flow.*

See [2] for a nice short proof of Seymour's result that every bridgless graph has a 6-flow.

10 Chordal Graphs and Lexicographic Breadth First Search

Colouring graphs– finding their chromatic number– is hard. But colouring trees is easy. Can we find other classes of graphs for which colouring is easy?

Note 10.1

Planar graphs are easy to 4-colour, and we can decide if a graph is planar, but Garey, Johnson and Stockmeyer showed in 1976 that it is hard to decide if a planar graph is 3-colourable. So finding chromatic number is hard even for planar graphs.

Perfect graphs are a large class of graphs that are 'easy' to colour. A graph G is perfect if for every induced subgraph $H \leq G$ we have that $\chi(H) = \omega(H)$.

Recall that $\omega(H)$ is the clique number– order of the maximum clique.

Problem 10.1. Give an example of a graph that is not perfect. For $k \geq 3$ find a graph G_k with $\chi(G_k) = k$ but $\omega(G_k) < k$.

Now, the fact that $\chi(G) = \omega(G)$ doesn't, apriori, make colouring perfect graphs easy. It is a hard problem to find the largest clique of a graph. But in 1988 Grötschel, Lovász, and Schrijver gave a polynomial time algorithm for finding the chromatic number of perfect graphs. This algorithm is a bit advanced though. Let's do something easier.

We have a good algorithm for colouring graphs– the greedy algorithm gives the chromatic number *if we start with the right ordering*. So maybe there are graphs for which we can easily find this ordering.

10.1 Interval graphs

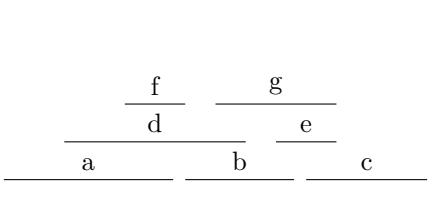
'Intersection graph' is a catch-all term for a graph whose vertices are mathematical (often geometric or set) objects such that two are adjacent if they intersect. Depending on the construction these graphs can be quite natural – occuring in various applications; and they can have a lot nice properties.

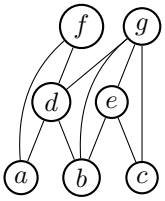
Let $\mathbb{R}_I$ be the family of all intervals in $\mathbb{R}$. An *interval representation* of a graph G is a function $I : V(G) \to \mathbb{R}_I : v \mapsto I_v$ such that

$$u \sim v \text{ if and only if } I_u \cap I_v \neq \emptyset.$$

A graph G is an *interval graph* if it has an interval representation.

Example 10.1. A graph and its interval representation. We draw the intervals not on the real line, but above them, so that we can see them all.





Problem 10.2. Show that (for finite graphs) we can replace the family $\mathbb{R}_I$ of intervals with 'open intervals' or with 'closed intervals' without changing the class of interval graphs. Show that we may assume no two intervals share an endpoint, again without changing the class of inteval graphs.

There are two common restrictions on the intervals that do change the class of graphs. An interval graph is a ...

- *unit* interval graph it has an interval representation in which all intervals have unit length.
- *proper* interval graph it has an interval representation in which no interval properly contains another interval.

Problem 10.3. Find an interval graph that is not a unit interval graph. Show that a graph is a proper interval graph if and only if it is a unit interval graph.

A *k-hole* in a graph G is an induced copy of the cycle C_k in G. A *k-antihole* in G is an induced copy of C_k in $\overline{G}$.

Problem 10.4. Show that any graph with a 5-hole also contains a 5-antihole.

The following gives a very useful propery of interval graphs, we will come back to it.

Problem 10.5. Show that an interval graph has no k-holes for $k \geq 4$. Show that interval graphs may have 4-antiholes, but have no k-antihole for $k \geq 5$. (Hint: The proof of the last statement for $k = 5$ is basically done above, and for $k \geq 6$ can be done at once in just a line or two.)

Having the interval representation of an interval graph makes several difficult problems easy.

Problem 10.6. Show that a set of vertices in an interval graph induces a clique if and only if the associated set of intervals has non-empty intersection. Explain how you would therefore find the maximum clique of an interval graph, if you had its interval representation.

Problem 10.7. Order the vertices of an interval graph by the ordering of the left endpoints of their intervals, (we showed above these can be assumed to be distinct). Show that a greedy colouring uses $\omega(G)$ colours. Conclude that $\chi(G) = \omega(G)$ for interval graphs.

A graph class is *hereditary* if G being in the class and H being an **induced** subgraph of G implies that H is also in the class.

Problem 10.8. Show that the class of interval graphs is hereditary, conclude, using Problem 10.7, that interval graphs are perfect.

Okay. So having this interval represenation of a graph is nice. How do we find it. One can do so in linear time, but we look at chordal graphs first.

10.2 Chordal Graphs

Interval graphs have this nice property that they have no *holes*. This property arises everywhere. A graph G is *chordal* if it has no holes. So interval graphs are chordal.

Here is an example of a class of chordal graphs.

For a tree T let $\mathcal{T}_T$ be the collection of subtrees of T. A *sub-tree representation* of a graph G consists of a tree T and a function $T^* : V(G) \to \mathcal{T} : v \mapsto T_v$ such that $u \sim v$ if and only if $T_u \cap T_v$ is non-empty (contains at least one vertex). We usually write $\mathcal{T}_T$ simply as $\mathcal{T}$ and T^* simply as T; so T is both a tree and a function.

Problem 10.9. Show that any graph with a sub-tree representation is chordal.

In fact the following is true. You will show it later after a couple of useful ideas.

Theorem 10.2. *A graph is chordal if and only if it is the intersection graph of a family of subtrees of a tree.*

Problem 10.10. Show that an interval graph G has a sub-tree representation T where the tree T is a path.

With this in mind:

Problem 10.11. Find a chordal graph that is not an interval graph.

There are several useful characterisations of chordal graphs. We look at one more that will help us decide if a graph is chordal.

An vertex v of a graph G is *simplicial* if its neighbours induce a clique. An ordering $v_1, \ldots, v_n$ of the vertices of a graph G is a *simplicial ordering* or a *perfect*

elimination ordering if for each i the vertex v_i is simplicial in the subgraph G_i of G induced by the vertex set $\{v_1, \ldots, v_i\}$.

We have seen graphs with simplicial orderings.

Problem 10.12. Show that the left endpoint ordering of an interval graph is simplicial.

Problem 10.13. Show that if v is simplicial in G and $G' = G \setminus \{v\}$ is chordal, then G is chordal. Conclude that any graph with a simplicial ordering is chordal.

The following should not be so much harder than the similar results for interval graphs.

Problem 10.14. Show that for a graph G with a simplicial ordering the chomatic number is equal to the clique number.

Dirac, in 1962, showed the following.

Theorem 10.3. *A graph is chordal if and only if it has a simplicial ordering.*

The difficult part of this is showing that any chordal graph has a simplicial vertex. We do this with the following stronger statement. (This proof is from [4] and is attributed to Voloshin and Farber-Jamison.)

The i^{th} *distance neighbourhood* $N^i(v) = N^i_G(v)$ of a vertex v in a graph G is the set of vertices that are distance exactly i from r in G. So $N^0(v) = \{v\}$ and $N^1(v) = N(v)$. The largest d such that $N^d(v)$ is non-empty is the *eccentricity* of v in G; the sets $N^i(v)$ for $i = 0, 1, \ldots, d$ are a partition of $V(G)$.

Lemma 10.4. *Let x be a vertex in a chordal graph G. Among the vertices of G that are furthest from x, there is a simplicial vertex of G.*

Proof. The proof is by induction on n and is trivial if $n = 1$, so assume that $n \geq 2$. Let x have eccentricity d. If $d = 1$ then by induction there is a simplicial vertex in $N^1(x) = G \setminus \{x\}$, and this vertex is simplicial in G. So we may assume that $d \geq 2$.

Let H be a component of the subgraph $G[N^d(x)]$ of G, and let S be the set of vertices in $N^{d-1}(x)$ with neighbours in H.

We claim that S is a clique. Indeed, if there were non-adjacent vertices s and s' in S then there is an induced path P between them in H, and an induced path P' between them in $G(V) \setminus (N^d(x) \cup N^{d-1}(x))$. There can be no edges between P and P' so with s and s' they make an induced cycle of girth at least 4. But this is impossible as G is chordal.

If every vertex of H is adjacent to every vertex of S, then any simplicial vertex h of H is simplicial in $S \cup H$, and so in G, and so we are done. Thus we

may assume that there is s in S that is not adjacent to all of H. Applying the induction hypothesis to $G[S \cup H]$ with s playing the role of x, we get a simplicial vertex in $G[S \cup H]$ that is not adjacent to s, so is in H. Again, this is simplicial in G and so we are done. □

Problem 10.15. Use Lemma 10.4 to prove Theorem 10.3.

Use the fact that every chordal graph G has a simplicial vertex to prove the following by induction. This finishes the proof of Theorem 10.2.

Problem 10.16. Every chordal graph has a subtree representation T such that

i. the vertices of the tree T correspond to maximal cliques of G,

ii. if two vertices of T are adjacent then corresponding maximal cliques of G intersect, and

iii. the subtree T_v for a vertex v of G is induced by the set of vertices of T corresponding to maximal cliques of G that contain v.

10.3 Breadth First Search

Many algorithms on a graph G start with an ordering $v_1, \dots v_n$ of its vertices. There are many ways to order the vertices of G, and one of the most common is the BFS ordering. The 'BFS' comes from a 'breadth first search', and refers to a spanning tree T of G that can be used to find the ordering.

To define the BFS algorithm, we us the notion of a queue. The queue Q is an ordered set, so has a minimum element $\min(Q)$. We will update it in our algorithm by removing (or *popping*) the minimum element of Q, and adding (or *pushing*) new maximum elements to Q.

Any ordering of $V(G)$ output by the following algorithm is a BFS-*ordering from* v_1.

Algorithm 10.5 (BFS(G, v_1))**.**

Input: *A connected graph G and a vertex $v_1 \in V(G)$.*
Ouput: *An ordering $v_1, \dots, v_n$ of $V(G)$.*
Initialisation: *Let $i = 1$, $Q = N(v_1)$, and R be the set $V(G) \setminus N[v_1]$*
Algorithm: *While Q is not empty, increase i by 1 and:*

i. Let v_i be the vertex $\min(Q)$ and remove it from Q.

ii. Move the vertices of $N(v_i) \cap R$ from R to the top of Q.

Writing an ordering of $V(G)$ as a labelling $\{v_1, \ldots, v_n\}$ of the vertices is useful, but thinking of it as a relation $\leq$, where where we write $v_i \leq v_j$ if $i < j$, also has its uses. Doing this, we can throw away the indices, and write such things as $a \leq b$ and the variations $a < b$ and $b \geq a$.

Problem 10.17. Show that a BFS-ordering $\leq$ of G with minimum element r satisfies the following:

$$u \leq v \Rightarrow d(r, u) \leq d(r, v).$$

(The closer a vertex is to the minimum element r the earlier it comes in the ordering.)

Problem 10.18. Draw a reasonably random looking graph on 10 vertices with around 25 edges, label its vertices, and find a BFS ordering of the vertices.

Problem 10.19. Write a program in Python or Sage to find a BFS ordering of an inputted graph and starting vertex. Prove that its running time is $O(|n| + |m|)$.

Problem 10.20. Show by example that BFS(G, v) can produce different orderings of $V(G)$.

Problem 10.21. Show that finding the distance between two vertices u and v of G can be done with one appliction of BFS. Show that the diameter of a graph can be determined in polynomial time.

BFS orderings of a graph can be quite useful, but if we are more discerning about the order we add vertices to Q, they can be even more useful.

10.4 Lexicographic Breadth First Search

We will now give a more deterministic algorithm for finding a BFS-ordering of $V(G)$. Before we give our algorithm, we need another definition. Recall that the elements of Q are ordered. The *lexicographic ordering* $\leq_Q$ of the family 2^Q of subsets of Q is defined by by setting $A \leq_Q B$ if $A \setminus B$ contains a smaller element of Q than $B \setminus A$ does.

The following is the *lexicographic breadth first search*.

Algorithm 10.6 (Lex-BFS(G, v_1))**.**

Input: *A connected graph G and a vertex $v_1 \in V(G)$.*
Ouput: *An ordering $v_1, \ldots, v_n$ of $V(G)$.*
Initialisation: *Let $i = 1$, Q be a queue containing $N(v_1)$ and $R = V(G) \setminus N[v_1]$*
Algorithm: *While Q is not empty, increase i by 1 and:*

i. Let v_i be the vertex $\min(Q)$ *and remove it from Q.*

ii. While $N(v_i) \cap R$ is non-empty move, from R to the top of Q, the vertex $x \in N(v_i) \cap R$ for which $N(a) \cap Q$ is minimum with respect to $\leq_Q$.

Clearly an ordering produced by Lex-BFS(G, v_1) is a BFS-ordering. It has the following nice property that a BFS-ordering might not have.

Fact 10.7. *Let $\leq$ be an ordering of $V(G)$ returned by* Lex-BFS(G, v_1). *If $c < b < a$ and $b \not\sim c \sim a$ then there is some vertex $f = f(b, a) < c$ such that*

i. $b \sim f \not\sim a$, and

ii. for all vertices g with $g < f$ we have $b \sim g \iff a \sim g$.

Proof. When b was put into the queue in the Lex-BFS it was put in before a, and all their neighbours that are below b were in Q. As b was chosen before a, we had that $N(b) \cap Q \leq_Q N(a) \cap Q$ and so the smallest vertex f that was a neighbour of one but not the other, was adjacent to b and not a. All vertices g below f thus have $b \sim g \iff a \sim g$. □

Problem 10.22. Show (again) that Lex-BFS(G, v) can produce different orderings of $V(G)$.

Theorem 10.8. *A graph G the following are equivalent.*

i. Lex-BFS(G, v_1) *produces a simplicial ordering for every choice of v_1.*

ii. Lex-BFS(G, v_1) *produces a simplicial ordering for some choice of v_1.*

iii. G is chordal.

Proof. Cleary *i.* implies *ii.*, and we have seen that *ii.* implies item *iii.*, so we only have to show that *iii.* implies *i.* Let $\sigma = (v_1, \ldots, v_n)$ be output by Lex-BFS(G, v_1) for arbitrary vertex v_1; we show that σ is simplicial. To do so, we actually show just that v_n is simplicial. Arguing that $(v_1, \ldots, v_{n-1})$ is also a Lex-BFS ordering of $G \setminus \{v_n\}$ from v_1, the result follows by induction.

That v_n is simplicial is immediate from the following.

Claim 10.9. *For vertices $a_1 < b_1 < w$ of G with $a_1 \sim w \sim b_1$, we have $a_1 \sim b_1$.*

Proof. Towards contradiction, we assume that $a_1 \not\sim b_1$. We will show by induction on i that G contains an arbitrarily long induced path

$$a_i, \ldots, a_2, a_1, a_0, b_1, b_2, \ldots, b_i \text{ with } a_i < b_i < a_{i-1} < b_{i-1} < \cdots < a_1 < b_1 < w.$$

This will contradict the fact that G is finite, and give our result. We induce with the stronger statement that the path also satisfies the property (*) 'for all $g < a_i$ we have that $g \sim a_{i-1}$ if and only if $g \sim x$ for all x in the path with $a_{i-1} < x$'.

This property (*) is vacuously true for $i = 1$ (treating w as a_0) and so we have the base of our induction. Assume we have a path $a_i, \ldots, a_1, w, b_1, \ldots, b_i$, satisfying the necessary properties.

As $a_i < b_i < a_{i-1}$, and $b_i \not\sim a_i \sim a_{i-1}$ because the path is induced, Fact 10.7 gives us a vertex $b_{i+1} = f(b_i, a_{i-1}) < a_i$ that is adjacent to b_i but not to a_i. Moreover, for any $g < b_{i+1}$ we have that $g \sim b_i \iff g \sim a_{i-1}$, and by induction we have that $g \sim a_{i-1}$ if and only if $g \sim x$ for all $x > a_{i-1}$ in the path, so $g \sim b_i$ if and only if $g \sim x$ for all $x > b_i$ in the path; this is statement (*), but for b_{i+1} in place of a_i. Observing that b_{i+1} is not adjacent to a_i, or we would have an induced cycle of girth at least 4, we have that the path $a_i, \ldots, a_2, a_1, a_0, b_1, b_2, \ldots, b_i, b_{i+1}$ is induced. This is our induction step. Switching the roles of a and b we can extend the path to a_{i+1} in the same way, and the argument follows by induction. $\diamond$

$\square$

With this we can test if G is chordal by picking a vertex v_1, running BFS(G, v_1) and then checking if the resulting ordering is simplicial.

Problem 10.23. Use this test to show that the Petersen graph is not chordal. (Or pick another reasonably complicated graph on 10 vertices and test its chordality by this test.)

Problem 10.24. Write a computer program that takes a graph as input and returns a T or F depending on whether or not it is chordal.

Problem 10.25. How long does the test for chordality take?

Notice that the ordering produced by Lex-BFS(G, v_1) is still not unique– two vertices in U might have the same minimum label set. To make it unique, we have to decide how ties are broken. We can do this with the following variation that uses an initial ordering $\sigma = (u_1, \ldots, u_n)$ of $V(G)$.

Algorithm 10.10 (Lex-BFS(v_1, σ))**.**

Input: *A connected graph G and a vertex $v_1 \in V(G)$.*
Ouput: *An ordering $v_1, \ldots, v_n$ of $V(G)$.*
Initialisation: *Let $i = 1$, Q be a queue containing $N(v_1)$ and $R = V(G) \setminus N[v_1]$*
Algorithm: *While Q is not empty, increase i by 1 and:*

i. Remove a vertex from Q and call it v_i.

ii. Add the neighbours of v_i in R to Q, in the following order: add a before b if

$$N(a) \cap Q \leq_Q N(b) \cap Q,$$

or if $N(a) \cap Q = N(b) \cap Q$ and $a \leq_\sigma b$.

iii. Remove neighbours of v_i from R.

We will not go into the details, but to check if a graph is an interval graph one can run Lex-BFS(v_1, σ) six times, but after the first time, you must use the results of earlier runs, some times with the ordering reversed. It seems like magic.

11 Graph Homomorphsims

See the book **Graphs and Homomorphisms** by Hell and Nešetřil [3] for this material.

Given two graphs G and H, a homomorphsim $\phi : G \to H$ is a vertex mapping $\phi : V(G) \to V(H)$ that preserves edges:

$$u \sim_G v \Rightarrow \phi(u) \sim_H \phi(v).$$

The language of homomorphsims is useful for much graph terminology:

- A walk of length n in a graph H is a homomorphism of a P_n to H.
- It is a trail if the homomorphism is edge-injective and a path if the homomorphism is vertex-injective.
- A k-colouring of a graph G is a homorphism of G to K_k.
- A graph is bipartite if and only if it admits a homomorphism to K_2.
- A graph is bipartite if and only if it admits no homomorphsim from an odd cycle.

11.1 The Homomorphism Order

The family of graphs with homomorphisms as morphisms is a category, and as such one can consider many categorical constructions.

The first such construction is to use us homomorphisms to define a partial order on the family of graphs. Recall that a *partial order* is a transitive reflexive, asymmetric relation on a set.

Problem 11.1. Show that the relation $G \to H$ is a transitive reflexive relation on the class of all graphs, but not asymmetric.

To make a partial order, we have make '$\to$' into an asymmetric relation. We have to remove directed cycles. The standard approach is the following.

Two graphs G and H are *homomorphically equivalent* if $G \to H$ and $H \to G$. Let $[G]$ denote the set of all graphs that are homomorphically equivalent to $[G]$ and write $[G] \leq [H]$ if $G \to H$.

Problem 11.2. Show that the set of *homomorphism classes* $[G]$ is partially ordered under $\leq$.

This is called the *homomorphism order of graphs.*

Problem 11.3. This partial order is well defined. Show that if $G \to H$, then $G' \to H'$ for any $G' \in [G]$ and $H' \in [H]$.

It can become bothersome to always talk of 'homomorphism classes of graphs' and so one often takes the route of restricting to a canonical representative of each class.

Problem 11.4. Show that for any graph G, there is, upto isomorphism, a unique smallest (wrt number of vertices) graph $C(G)$ in the homorphism class $[H]$.

This graph $C(G)$ is the *core* of H, and if a graph C is the core or some graph G, then it is a *(graph) core*. The homomorphism order of graphs can be viewed as a partial order of the set of graph cores. This is the point of view we will take. For questions of the homomorphism order, one should assume that all graphs are cores.

There are many questions that arise from the fact that we have a partial order.

Problem 11.5. Does the homomorphism order have a maximum or minimum element. (We consider only finite graphs.)

Problem 11.6. Does the homomorphism order have any infinite acsending or infinite descending chains?

Clearly

$$K_1 \to K_2 \to K_3 \to K_4 \to \cdots$$

is an infinite ascending chain in this order and

$$\cdots \to C_{11} \to C_9 \to C_7 \to C_5 \to C_3$$

is an infinite descending chain.

An *antichain* in the homomorphism order is a set of graphs amongst which there are no homomorphisms.

Problem 11.7. Does the homomorphism order have any infinite antichains?

This problem is a little harder, but we will see a useful tool for answering it in a second.

A pair (A, B) is a *gap* in the homomorphism order if $A < B$ and there is no graph C such that $A < C < B$.

Problem 11.8. Show that (K_1, K_2) is a gap, but that for all $n \geq 4$, the pair (K_{n-1}, K_n) is not.

Problems 11.7 and 11.8 are a bit tricky, but both can be answered with the following result.

Theorem 11.1 (Erdős 1959)**.** *For all $g, k \geq 3$ there exist graphs of girth g and chromatic number k.*

We will prove this in a later chapter using the probabilistic method. For now, we consider a slightly weaker constructive results that can also be used in answering the above questions.

Theorem 11.2 (Mycielski 1955)**.** *For $k \geq 3$ there is a k-chromatic graph M_k containing no K_3.*

Proof. The graph $M_3 = C_5$ satisfies the theorem for $k = 3$. Consider the following construction.

For a graph G let the Mycielski graph $M = M(G)$ be the graph on the vertex set $V(M) = V \cup U \cup \{w\}$ where $V = V(G) = \{v_1, \ldots, v_n\}$ and $U = \{u_1, \ldots, u_n\}$ and

$$E(M) = E(V) \cup \bigcup \{u_i v_j, v_i v_j \mid v_i v_j \in E(G)\} \cup \{w u_i \mid u_i \in U\}.$$

We claim that if G is K_3-free then so is $M(G)$ and if $\chi(G) = k$ then $\chi(M(G)) = k + 1$. One this is proved, setting $M_k = M(M_{k-1})$ finishes the proof of the theorem by induction. Part of is is pretty straight-forward.

Problem 11.9. Show that if G is K_3-free then $M(G)$ is. Show that if $\chi(G) = k$ then $\chi(M(G)) \leq k + 1$.

To finish the proof, we show that if $\chi(G) = k$ then $M(G)$ does not have a k-colouring, so $\chi(M(G)) = k + 1$. Indeed, towards contradiction, assume that $M(G)$ has a k colouring f, and wlog that $f(w) = k$. Let R be the vertices of V that get colour k, and redefine f on R by setting $f(v_i) = f(u_i)$; then restrict f to V. We claim that f is now a $(k-1)$-colouring of G, which would be impossible. Indeed, it is enough to observe that f properly colours edges within R, and those between R and $\overline{R}$. But there are no edges within R as the vertices all had colour k. For edges from $v_i \in R$ to $v_j \in \overline{R}$, we have that $v_j \sim u_i$ so v_j it doesn't have colour $f(u_i)$.

□

Problem 11.10. This problem is pretty hard. It is not on the test. Alter Mycielski's construction to show that for every odd $g \geq 3$ there is a g-chromatic graph with odd girth $2g + 1$.

We will show that (K_1, K_2) is the only gap in the homomorphism order. In fact, this follows from Problem 11.10, but it uses a couple more nice little ideas that we should look at first.

11.2 Products and the Exponential Graph

The *(categorical) product* $G \times H$ of two graphs G and H is defined by $V(G \times H) = V(G) \times V(H)$ and

$$(g, h) \sim (g', h') \iff g \sim g' \text{ and } h \sim h'.$$

Problem 11.11. Show that the projections

$$\pi_1 : G \times H \to G \qquad \text{and} \qquad \pi_2 : G \times H \to H$$

defined $\pi_1((g, h)) = g$ and $\pi_2((g, h)) = h$ are homomorphisms. Conclude that if $G \to K$ or $H \to K$ then $G \times H \to K$.

Problem 11.12. Show that $G \times H$ has the following property, (which means that it is the *product* in the category of graphs).

For any graph R with homomorphisms $g : R \to G$ and $h : R \to H$, there is a unique homomorphism $f : R \to G \times H$ such that $g = \pi_1 \circ f$ and $h = \pi_2 \circ f$.

For graphs G and H, let the *exponential graph* H^G has as its vertices the set of mappings (not homomorphsims) of $V(G)$ to $V(H)$. Two maps ϕ and ϕ' are adjacent if $\phi(g) \sim \phi'(g')$ for all edges $g \sim g'$ of G.

Problem 11.13. Show that a vertex ϕ of H^G has a loop if and only if it is a homomorphism.

Proposition 11.3. *For graphs F, G and H we have*

i. $H^{G+F} \cong H^G \times H^F$

ii. $H^{G \times F} \cong (H^G)^F$

iii. $G \times F \to H$ *if and only if* $F \to H^G$.

Proof. Part *i.*: We claim that $\Phi : H^{G+F} \to H^G \times H^F$ defined for a map $\phi \in H^{G+F}$ by $\Phi(\phi) = (\phi|_G, \phi|_F)$ is an isomorphism. It is clearly a bijection as $\phi \neq \phi'$ if and only if ϕ and ϕ' differ on some vertex g of G or f of F, if and only if $\phi|_G$ differs from $\phi'|_G$ or $\phi|_F$ differs from $\phi'|_F$.

To see that it is an isomorphism observe that $\phi \sim \psi$ in H^{G+F} if and only if $\phi_G \sim \psi_G$ in H^G and $\phi_F \sim \psi_F$ in H^F. Both are true if and only if $\phi_G(g) = \phi(g) \sim \psi(g') = \psi_G(g')$ for every $g \sim g'$ in G and blah blah for F.

Problem 11.14. Prove part *ii*. (The proof is a similar to that of part *i*. The hardest part of it is keeping the notation and definitions straight. If you have trouble with it, it is in the Hell Nešetřil book.)

Problem 11.15. Prove part *iii*. (Using part *ii* and Problem 11.13, this should only take a line or two.)

□

Theorem 11.4. *The only gap in the homomorphism order is* (K_1, K_2).

Proof. We must show that if $(G, H) \neq (K_1, K_2)$ and $G \to H$ (but $G \not\leftarrow H$), then there is a graph M such that $G \to M \to H$ (but $G \not\leftarrow M \not\leftarrow H$).

The graph $M = G + (Z \times H)$ will do, where Z is a graph, from Problem 11.10, of odd girth greater than the odd girth of G having chromatic number greater than the chromatic number of G^H.

Indeed, G maps to the copy of G in M, and as $G \to H$ and $Z \times H \xrightarrow{\pi_2} H$, we have that $M \to H$. On the other hand as Z has large odd girth, so does $Z \times H$, and so $H \not\to Z \times H$, we also have $H \not\to G$, so $H \not\to M$. Finally, if $M \to G$, then in particular $Z \times H \to G$ and so $Z \to G^H$; but Z has larger chromatic number than G^H, so this is impossible.

□

11.3 H-Colouring Dichotomy

For a graph H the problem of *H-colouring* denoted $\mathrm{Hom}(H)$ is to decide for a given instance G whether or not there is a homomorphism $G \to H$.

As 3-colouring is difficult, or NP-complete, we have that $\mathrm{Hom}(K_3)$ is NP-complete.

Problem 11.16. Show that for any bipartite graph H, $\mathrm{Hom}(H)$ can be solved in polynomial time.

What about for other graphs? Hell and Nesětřil showed the following in 1990.

Theorem 11.5. *For any H containing an odd cycle,* $\mathrm{Hom}(H)$ *is NP-complete.*

The proof of this uses a series of polynomial reductions.

Example 11.6 (Proof that C_5 is in NPC)**.** Let C_5 have vertices $1, \ldots, 5$ with vertices being adjacent if they are consecutive mod 5. Let P be a path on the four vertices $v_1, \ldots, v_4$ with vertices being adjacent if their indices are consecutive.

Given an instance G of $\mathrm{Hom}(K_5)$, let the graph G^* be constructed from G by the following construction. For every original edge uv of G, remove uv and add a new copy of P by identifying the vertices v_1 and v_4 in the copy of P with the vertices u and v of G respectively.

Notice that there for $i \neq j \in \{1, \ldots, 5\}$ there are C_5-colourings ϕ_{ij} of P such that $\phi_{ij}(v_1) = i$, and $\phi_{ij}(v_4) = j$; however, there is no C_5-colouring of P that takes both v_1 and v_4 to the same image. One can thus verify that

$$G^* \to C_5 \iff G \to K_5.$$

Thus $K_5 <_{NP} C_5$, and so because K_5 is in NPC, so is C_5.

Problem 11.17. Using a similar indicator construction as in the example above, prove that if $\mathrm{Hom}(H)$ is NP-complete for all graphs containing a triangle, then it is true for all graphs containing an odd cycle.

The rest of the proof of Theorem 11.5 takes some work. But it follows easily from a (much stronger) result of Barto and Kozik which holds for all relational structures H.

A homomorphism $\phi : H^d \to H$ is a *cycle term* if

$$\phi(v_1, \ldots, v_n) = \phi(v_2, \ldots, v_n, v_1) = \cdots = \phi(v_n, v_1, \ldots, v_{n-1}).$$

Theorem 11.7. *If H has no cycle term of arity p, for any given prime $p > |V(H)|$, then* $\mathrm{Hom}(H)$ *is NP-complete.*

Now let H be a simple graph containing an odd cycle. Let p be an odd prime greater than n. By the theorem, if we can show that H has no cycle term or arity p, then $\mathrm{Hom}(H)$ is NP-complete. Assume it has a p-cycle term.

Problem 11.18. Show that if H has an odd cycle of length $g \leq p$ for p odd, then it has a closed circuit $v_1 \sim v_2 \sim \cdots \sim v_p \sim v_1$ of length p.

But then

$$\phi(v_1, \ldots v_p) = \phi(v_2, \ldots, v_p, v_1)$$

and

$$\phi(v_1, \ldots v_p) \sim \phi(v_2, \ldots, v_p, v_1)$$

which means that H contains a loop. But this is a contradiction.

The converse of the Theorem 11.7 was proved independently by Bulatov, and Zhuk, in 2017.

Theorem 11.8 (CSP-dichotomy conjecture)**.** *If a digraph (or any relational structure) H has a cycle term, then* $\mathrm{Hom}(H)$ *is polynomial time solvable.*

12 Extremal Graph Theory

The basic idea of Extremal Graph Theory is to find the largest (smallest) graph with (without) a given property. Largest (smallest) can refer to the number n of vertices or m of edges, but often n is fixed, and largest refers to m.

12.1 Turan's Theorem

For a graph H, the *extremal number* $\mathrm{ex}(n, H)$ for H is the maximum number of edges a graph on n vertices can have without containing a copy of H as a subgraph.

The Turan graph $T_{k-1}(n)$ is the complete $(k-1)$-partite graph with vertex partition

$$V = V_1 \cup V_2 \cup \ldots V_{k-1}$$

where

$$|V_1| \leq |V_2 \leq \cdots \leq |V_{k-1}| \leq |V_1| + 1.$$

When $n = q(k-1)$ for integer q, this has $\binom{k-1}{2}q^2 = \frac{k-2}{2(k-1)}n^2$ edges.

Problem 12.1. Find a general formula for the size $t_{k-1}(n)$ of $T_{k-1}(n)$.

Theorem 12.1. *[Turan 1941] For all $n \geq k > 1$, $\mathrm{ex}(n, K_k) = t_{k-1}(n)$.*

Proof. It is enough to show that the maximum number of edges in a graph omitting K_k is a complete multipartite– by convexity the turan graph has the most edges for a fixed number of parts, and the edges count increases in the number of parts, which is bounded by $k-1$, so this will do.

Let G be an extremal K_k-free graph on n vertices. If it is not complete multipartite, then there are adjacent vertices y_1 and y_2 with common non-neighbour x. If $d(y_1) > d(x)$, then deleting x and replacing it with a clone of y_1 gives a K_k-free graph with more edges; a contradiction. On the other hand, if $d(y_1), d(y_2) \leq d(x)$, then removeing y_1 and y_2 and cloning x twice gives a K_k-free graph with more edges. □

Problem 12.2. Show that when we add an edge to $T_{k-1}(n)$ we get no only one copy of K_k, but at least $\lfloor \frac{n}{k-1} \rfloor^{k-2}$ of them.

This is not only true of the Turan graphs.

Theorem 12.2. *[Erdős-Stone '46] For all integers $k \geq 2$ and $s \geq 1$, and every $\varepsilon > 0$ there exists an integer n_0 such that every graph with $n \geq n_0$ vertices and at least $t_{k-1}(n) + \varepsilon n^2$ edges, contains a $K_{s,s,\ldots,s}$ (with k 's's).*

We will prove this later with the regularity lemma.

12.2 Ramsey Theory

Ramsey showed the following.

Theorem 12.3. *For all $s, t \geq 3$ there exists a number n such that for any red-blue colouring of the edges of K_n there is either a red K_s or a blue K_t.*

The *ramsey number* $r(s,t)$ is the minimum such n. The ramsey numbers that we know are:

s,t	3	4	5
3	6		
4	9	18	
5	14	25	$43-49$
6	18	$35-41$	$58-87$
7	23	$49-61$	$80-143$

Problem 12.3. Show that for all $t \geq 3$, $r(t,t) < 2^{2t}$.

The proof of the following theorem is one of the early uses of the probabilistic method, which we will see more of, in the next section.

Theorem 12.4 (Erdős)**.** *For all $t \geq 3$ we have $r(t,t) > 2^{t/2}$*

Proof. Red-blue colour the edges of K_n as by letting each edge (independently) be blue with probability $1/2$. For a random set S of vertices, the probability that S induces a mono-chromatic K_t is

$$2 \cdot \left(\frac{1}{2}\right)^{\binom{t}{2}} = 2^{1-\binom{t}{2}}.$$

There are $\binom{n}{b}$ such sets, so the probability that at least one is monochromatic is less than

$$\binom{n}{t}\frac{2}{2^{\binom{t}{2}}} < \frac{n^t}{t!}\frac{2}{2^{\binom{t}{2}}} = \frac{n^t}{2^{t^2/2}}\frac{2^{1+t/2}}{t!} < \frac{n^t}{2^{t^2/2}}.$$

Setting $n = 2^{t/2}$, this is less than 1, so some random colouring of K_n yields no mono-chromatic K_t. □

More generally we can define $R(H_r, H_b)$ for two graphs H_r and H_b as the minimum n such that for any red-blue colouring of the edges of K_n there is a red copy of H_r or a blue copy of H_b.

This is clearly equivalent to finding the minimum n such that for any graph G on n vertices, either G contains a copy of H_r or $\overline{G}$ contains a copy of H_b.

Problem 12.4. For a tree T on t vertices, find $R(T, K_s)$.

We will show the following, once we have seen the regularity lemma.

Theorem 12.5 (Chvatal, Rödl, Szemeredi, Trotter '83)**.** *For all Δ there exists $c_\Delta > 0$ such that for all H with $\Delta(H) \leq \Delta$ we have $R(H,H) < c_\Delta |V(H)|$.*

12.3 VanDerWaerden

The *Van Der Waerden number* $W(2,\ell)$ is the smallest n such that an 2-colouring of $[n]$ yields a monochromatic arithemetic progression of length ℓ. Van Der Waerden proved such numbers exist, in 1927. (They still exist.)

Problem 12.5. Show that $W(2,5) < 325$.

The best known bound for $W(k,\ell)$ is

$$W(k,\ell) < 2^{2^{k^{2^{\ell+9}}}}$$

by Gowers. Graham offers $1000 for showing

$$W(2,\ell) < 2^{k^2}.$$

13 The Probabilistic Method

We saw one proof using the probabilistic Method in Section 12.2. Proofs like this are simply counting proofs.

Consider the following two proofs of a simple result. They are essentially the same proof.

Proposition 13.1. *Every graph G has a bipartite subgraph H with $m_H \geq m_G/2$.*

Counting proof. Count the number P of pairs (S, e) where S is a subset of $V(G)$ and $e = \{u, v\}$ is an edge of G with one endpoint in S. We choose the edge, then endpoint, and then the other elements of S; so

$$P = m_G \cdot 2 \cdot 2^{n-2} = m_G 2^{n-1}.$$

There are 2^n different subsets S, so one is counted in at least $m_G/2$ pairs. H is the bipartite subgraph of G induced by the partite sets S and $V(G) \setminus S$. □

Probabilistic Version. Randomly choose a subset S ov $V(G)$ by putting each vertex v of G in S with probability $1/2$. Let H be the bipartite subgraph of G induced aby partite sets S and $V(G) \setminus S$. We show that the expected number of edges in H is at least $m_G/2$. Indeed, each edge of G is in H with probability $1/2$. So the expected number of edges in H is $\sum_{e \in G} 1/2 = m_G/2$.

It follows that there is some choice of S so that the induced subgraph H has at least $m_G/2$ edges. □

13.1 Random Graphs

The *Erdős Rényi Random Graph* $G_{n,p}$ on n vertices $[n]$ is constructed as follows. For each pair of vertices u, v the edge (u, v) is in $G_{n,p}$ with probability p, independent of the existence of other edges.

Formally, the random graph $G_{n,p}$ as a probability space, containing the $2^{\binom{n}{2}}$ (labelled) graphs on n vertices. The probability that $G_{n,p}$ is a given graph H on the vertices $[n]$ depends on the number of edges of H. If H has $|E(H)| = m$ edges, then the probability that $G_{n,p} = H$ is $p^m(1-p)^{\binom{n}{2}-m}$.

Computing the probability that $G_{n,p}$ contains an isomorphic copy of a certain subgraph is a little harder. But a starting point is to look at the expected number of copies of a subgraph.

Technically, we define a random variable $X_H = X_H(G_{n,p})$ that counts the copies of a subgraph H, and use the linearity of expectation to find its expected value.

The expected number of edges, or copies of K_2 in $G_{n,p}$ is $\sum_{u \neq v \in [n]} p = p \cdot \binom{n}{2}$. To find the expected number of k-cycles we observe that there are $n!/2k(n-k)!$

possible k-cycles on $[n]$ and that each occurs in $G_{n,p}$ with probability p^k. So the expected number of k-cycles is

$$n!/2k(n-k)\cdot p^k < (np)^k/k2$$

Note

f(x) = o(g(x)):

$\lim_{x\to\infty} \frac{f(x)}{g(x)} = 0$.

More generally, the expected number of cycles of length less than or equal to k is

$$\sum_{i=3}^{k} \frac{n^i p^i}{2i} < (k-2)(np)^k.$$

Now think of p as a function of n. If $p < n$, that is, $\lim_{n\to infty} p/n < 1$ then the expected number C of cycles of length k or less is $o(1)$.

Markov's inequality allows use to use expected value to bound the probability of an event:

For any positive random variable X

$$Prob(X \geq a) \leq E(X)/a,$$

where $E(X)$ is the expected value of X.

Apply this to the RV X that counts the cycles in $G_{n,p}$. If $p < n$ then the expected number of cycles $E(X)$ is less than ε for large enough n, so Markov's inequality gives us that the probability that $G_{n,p}$ has a cycle, it that $X \geq 1$ is less than ε.

That is, when $p < n$, $G_{n,p}$ is *asymptotically almost surely (aas)* acyclic: the probability that $G_{n,p}$ is acyclic goes to 1 as n goes to ∞.

Problem 13.1. Show that when $p = 1/n$ the expected number of cycles of any length is is less than $\ln(n/2)$.

Problem 13.2. Show that if $1/n = o(p)$ then $G_{n,p}$ has a cycle asymptotically almost surely.

Note 13.1

n fact, letting p increase from 0 to 1 we can look at $G_{n,p}$ as evolving from the empty graph on n vertices to the complete graph. Some epochs of the evolution are as follows.

- If $p = o(1/n)$ then $G_{n,p}$ is asymptotically almost surely a forest.
- If $p = c/n$ for $c < 1$ then $G_{n,p}$ is aas a union of single cycle components.
- If $p = c/n$ for $c > 1$ then $G_{n,p}$ ass has a giant component of $\Theta(n^{2/3})$ vertices.
- If $p = c\log n/n$ then $G_{n,p}$ is aas connected.

Now, we prove Theorem 11.1, which says that there is a graph of girth at least g and chromatic number at least k.

Proof of Theorem 11.1. The main idea is to choosing p so that $G_{n,p}$

- has as most $n/2$ cycles of length at most k, with probability less than $1/2$
- has an independent set of size at least $n/2k$, with probability less than $1/2$.

It follows that there exists a graph on n vertices with both of these properties. Then we can kill all cycles of length k or less by removing at most $n/2$ vertices. The remaining graph has at least $n/2$ vertices and independent set of size at most $n/2k$, so has chromatic number at least k. This is enough.

Let $k \geq 3$ be given. To choose p, fix some ε with $0 < \varepsilon < 1/k$ such that

$$\frac{6k \ln n}{n} < \frac{n^{\varepsilon}}{n} < \frac{n^{1/k}}{n},$$

and let $p = n^{\varepsilon}/n$.

We first show that the probability that the independence number α of $G = G_{n,p}$ is greater than $n/2k$ goes to 0 as n goes to ∞. Indeed, the probability thah $G_{n,p}$ has an independent set of size t (or greater) is less than

$$\binom{n}{t}(1-p)^{\binom{t}{2}} < \left(n(1-p)^{(t-1)/2}\right)^t \leq (ne^{-p(t-1)/2})^t$$

where the last inequality uses that $1 - x < e^{-x}$ for $x > 0$.

Using, in this, that $p \geq \frac{6k \ln n}{n}$ and $t > n/(2k)$ we get that

$$\begin{aligned} ne^{-p(t-1)/2} &< ne^{\frac{-pt}{2}+\frac{p}{2}} \\ &\leq ne^{\frac{-3\ln n}{2}+\frac{p}{2}} \\ &\leq nn^{-3/2}e^{p/2} \\ &\leq n^{-1/2}e^{1/2} = \sqrt{e}/\sqrt{n} \to 0 \end{aligned}$$

Now we show that the probability that the number of cycles of length k of less is greater than $n/2$, also goes to 0. Indeed, using Markov's inequality and the expected number of cycles we computed above, this probability is at most

$$(k-2)n^{k-1}p^k = (k-2)n^{k-1}n^{(\varepsilon-1)k} = (k-2)n^{k\varepsilon-1} < (k-2)n^{<0} \to 0.$$

As both probabilities go to 0, we have for large enough n that they are both less than $1/2$, as needed.

□

References

[1] M. Devos *Graph Theory Class Notes*
http://www.sfu.ca/∼mdevos/notes/graph/

[2] R. Diestal *Graph Theory (4th edition)* Springer 2010.
http://diestel-graph-theory.com/

[3] P. Hell, J, Nešetřil *Graphs and Homomorphism* Oxford university press 2004.

[4] D. West *An Introduction to Graph Theory (2nd edition)* Pearson Education 2001.