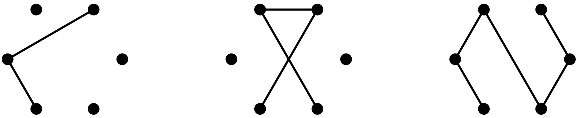
MATH1081 – Discrete Mathematics

Topic 5 – Graph theory
Lecture 5.01 – Graph and multigraph terminology

Lecturer: Dr Sean Gardiner – sean.gardiner@unsw.edu.au

Introduction to graph theory

In very simple terms, graphs are mathematical constructions represented by dots and connecting lines.



These constructions can represent many real-world situations, such as social networks, transportation options, or linked pages of the internet.

For this topic we will be investigating the abstract notion of a graph and the many properties different graphs can have.

Much of the topic will be spent establishing precise definitions and proving useful facts about graphs, but in almost all cases it is usually useful to accompany your learning with plenty of diagrams.

In lectures, we will sometimes prove statements loosely using diagrammatic arguments, while properly-written proofs will be provided in the completed versions of the slides.

Definition. A graph G is a mathematical object consisting of two related sets:

- The vertex set of G, written as $V(G)$ or just V, is a set of vertices (often labelled $v_1, v_2, v_3, \dots$).
- The edge set of G, written as $E(G)$ or just E, is a set of edges (sometimes labelled $e_1, e_2, e_3, \dots$). Each edge is itself a subset of $V(G)$ with cardinality 2, representing the two vertices it connects, which are known as the edge's endpoints.

Notice that since each edge is a set of cardinality 2, no edge can have a single vertex as both of its endpoints in a graph.

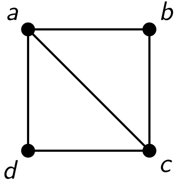
Notice also that since $E(G)$ is a set, there can be at most one edge between any two distinct vertices in a graph.

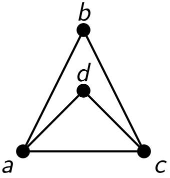
Notation. An edge with endpoints v and w is written as $\{v, w\}$, but is more often represented by the shorthand vw, or equivalently, wv.

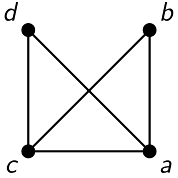
Representing graphs

Graphs can be represented diagrammatically as dots (vertices) joined by connecting line or curve segments (edges). The relative positions of the vertices and edges do not matter.

For example, the graph G with vertex set $V(G) = \{a, b, c, d\}$ and edge set $E(G) = \{ab, ac, ad, bc, cd\}$ can be represented by any of these diagrams:







Definition. Given a graph G, if a vertex $v \in V(G)$ is an endpoint of an edge $e \in E(G)$, then we say v is incident with e and e is incident with v. For example, ac is incident with a but not with b above.

Definition. Given a graph G, two vertices $v, w \in V(G)$ are said to be adjacent or neighbours if they are connected by an edge in $E(G)$, that is, if $\{v, w\} \in E(G)$. For example, b is adjacent to a above, but not to d.

Multigraphs

Definition. A multigraph G is a mathematical object consisting of two sets and a function:

- The vertex set $V(G)$ is simply a set of vertices.
- The edge set $E(G)$ is simply a set of edges.
- The endpoint function $f : E(G) \to \mathcal{P}(V(G))$ is a function mapping edges of G to subsets of $V(G)$ that have cardinality 2 or 1.

Notice that for multigraphs, an edge's endpoints may be a set of cardinality 1, meaning an edge may start and end at the same vertex.

Definition. A loop is an edge in a multigraph with exactly one endpoint.

Notice also that for multigraphs, the endpoint function might map two distinct edges to the same pair of endpoints, meaning there can be more than one edge between any two vertices.

Definition. If two edges in a multigraph have identical endpoints, they are called parallel edges or multiple edges. If a multigraph has exactly k edges with endpoints v and w, we say $\{v, w\}$ is an edge of multiplicity k.

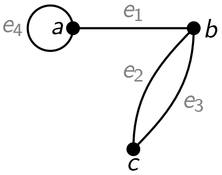
Representing multigraphs

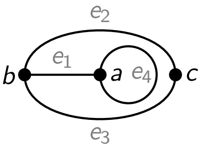
Multigraphs can be represented diagrammatically in the same way as graphs.

For example, the multigraph G with vertex set $V(G) = \{a, b, c\}$, edge set $E(G) = \{e_1, e_2, e_3, e_4\}$, and endpoint function

$$f = \{\, (e_1, \{a, b\}), (e_2, \{b, c\}), (e_3, \{b, c\}), (e_4, \{a\})\, \}$$

can be represented by either of these diagrams:





Notice that all graphs are multigraphs, with the restrictions that the endpoint function cannot map any edge to a single vertex, nor map any two distinct edges to the same pair of vertices.

Definition. A simple multigraph is a multigraph without any loops or parallel edges. That is, a simple multigraph is a graph.

Directed graphs

Definition. A directed graph G is a mathematical object consisting of two related sets:

- The vertex set $V(G)$ is a set of vertices.
- The edge set $E(G)$, for which each edge is an ordered pair of distinct vertices of G, representing the edge's start and end vertices in order.

Definition. A directed multigraph G is a mathematical object consisting of two sets and a function:

- The vertex set $V(G)$ is a set of vertices.
- The edge set $E(G)$ is a set of edges.
- The endpoint function $f : E(G) \to V(G) \times V(G)$ is a function mapping edges of G to ordered pairs of vertices.

When representing a directed (multi)graph diagrammatically, each edge is represented by an arrow pointing from its start vertex to its end vertex.

For example, the arrow diagram for a function $f : X \to Y$ is a directed graph, and the arrow diagram for a relation R on X is a directed multigraph. The Hasse diagram for any poset is also a directed graph, since all its edges are technically directed upwards.

Special graphs: Path graphs

Most of the Topic 5 content is focused specifically on graphs, as opposed to multigraphs or directed (multi)graphs.

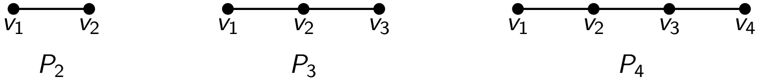
Some graphs are common or useful enough that they have special names.

We will define several such named graphs in terms of labelled vertices, though they are often also interpreted as being unlabelled.

Definition. The path graph on $n \geq 1$ vertices, written as P_n, is the graph with vertex set $V(P_n) = \{v_1, v_2, \dots, v_n\}$ and edge set

$$E(P_n) = \{v_1v_2, v_2v_3, \dots, v_{n-1}v_n\}.$$

For example,



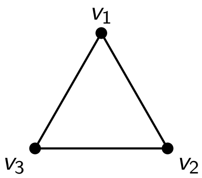
Exercise. How many vertices and edges does P_n have?

Solution. The number of vertices is $|V(P_n)| = n$, and the number of edges is $|E(P_n)| = n - 1$.

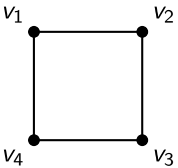
Definition. The cycle graph (or cyclic graph) on $n \geq 3$ vertices, written as C_n, is the graph with vertex set $V(C_n) = \{v_1, v_2, \ldots, v_n\}$ and edge set

$$E(C_n) = \{v_1v_2, v_2v_3, \ldots, v_{n-1}v_n, v_nv_1\}.$$

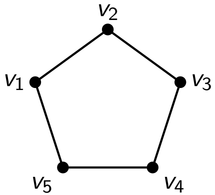
For example,



C_3



C_4



C_5

Exercise. How many vertices and edges does C_n have?

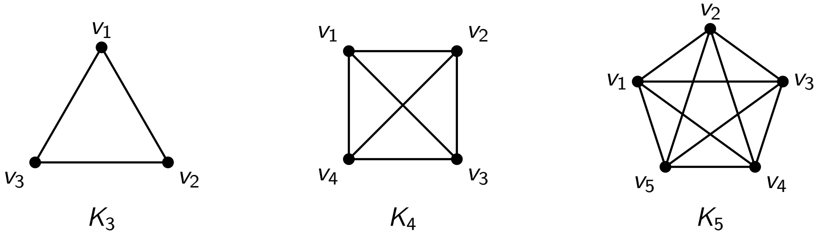
Solution. The number of vertices is $|V(C_n)| = n$, and the number of edges is also $|E(C_n)| = n$.

Special graphs: Complete graphs

Definition. The complete graph on $n \geq 1$ vertices, written as K_n, is the graph with vertex set $V(K_n) = \{v_1, v_2, \ldots, v_n\}$ and edge set

$$E(K_n) = \{\, \{v, w\} \subseteq V(K_n) \;:\; v \neq w \,\} = \{v_i v_j \;:\; 1 \leq i < j \leq n\}.$$

That is, the complete graph K_n is the graph with n vertices and all possible edges between these vertices. For example,



Exercise. How many vertices and edges does K_n have?

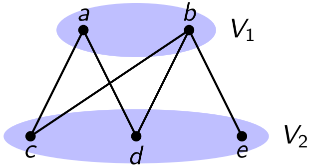
Solution. The number of vertices is $|V(K_n)| = n$, and the number of edges is given by the number of ways to choose two distinct unordered vertices, so $|E(K_n)| = \binom{n}{2}$.

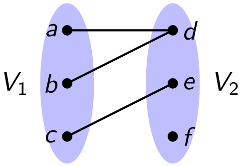
Bipartite graphs

Definition. A (multi)graph G is called bipartite if its vertex set $V(G)$ can be partitioned into two sets V_1 and V_2 such that each edge in G has one endpoint in V_1 and one endpoint in V_2.

That is, G is bipartite if and only if its edge set $E(G)$ is a subset of $\{\, \{v, w\} \subseteq V(G) \;:\; v \in V_1, w \in V_2 \,\}$ where $V(G) = V_1 \cup V_2$ and $V_1 \cap V_2 = \varnothing$.

It is standard to represent bipartite graphs by separating the two partitions of $V(G)$. Some examples of standard bipartite graph representations are:





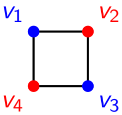
Notice that the arrow diagram for a function $f : X \to Y$ is a bipartite directed graph whose vertex set can be partitioned into the sets X and Y.

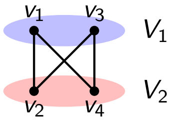
To decide whether a graph is bipartite, we must construct a valid partition of its vertex set, or show that no such partition exists.

Example – Bipartite graphs

Example. Is the cycle graph C_4 bipartite?

Solution. The graph C_4 is bipartite, since using the standard labelling, its vertex set can be partitioned into the sets $V_1 = \{v_1, v_3\}$ and $V_2 = \{v_2, v_4\}$ such that every edge in C_4 has one endpoint in V_1 and one endpoint in V_2.





Example. Is the cycle graph C_5 bipartite?

Solution. Suppose by way of contradiction that C_5 is bipartite, so that its vertex set can be partitioned into sets V_1 and V_2 where each edge in C_5 has one endpoint in V_1 and one endpoint in V_2. Using the standard labelling, let v_1 be an element of V_1. Then we must have $v_2 \in V_2$ since $v_1 v_2 \in E(C_5)$, so $v_3 \in V_1$ since $v_2 v_3 \in E(C_5)$, so $v_4 \in V_2$ since $v_3 v_4 \in E(C_5)$, so $v_5 \in V_1$ since $v_4 v_5 \in E(C_5)$. But then the edge $v_5 v_1$ in C_5 has both its endpoints in V_1, which is a contradiction. Thus C_5 is not bipartite.

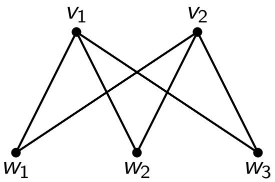
We will see another way to identify non-bipartite graphs in Lecture 5.03.

Special graphs: Complete bipartite graphs

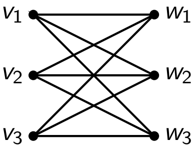
Definition. The complete bipartite graph on $m \geq 1$ and $n \geq 1$ vertices, written as $K_{m,n}$, is the graph with vertex set $V(K_{m,n}) = A \cup B$ where $A = \{v_1, v_2, \dots, v_m\}$ and $B = \{w_1, w_2, \dots, w_n\}$ with $A \cap B = \varnothing$, and edge set

$$E(K_{m,n}) = \{vw \ : \ v \in A, w \in B\}.$$

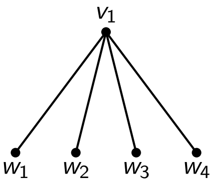
That is, the complete bipartite graph $K_{m,n}$ is the graph with a vertex set partitioned into sets of size m and n, where every vertex in one partition is adjacent to every vertex in the other partition (and no others). For example,



$K_{2,3}$



$K_{3,3}$



$K_{1,4}$

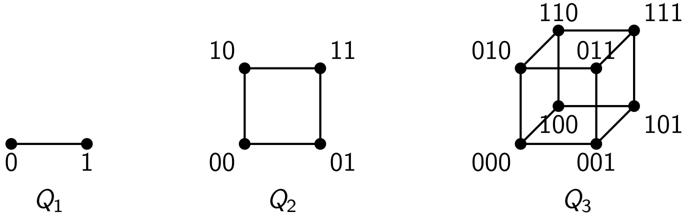
Exercise. How many vertices and edges does $K_{m,n}$ have?

Solution. The number of vertices is $|V(K_{m,n})| = m + n$, and the number of edges is $|E(K_{m,n})| = mn$.

Definition. The cube graph (or hypercube graph) of dimension n, written as Q_n, is a graph that uses a special labelling of its vertices as binary strings. Its vertex set is given by $V(Q_n) = \{\text{binary strings of length } n\}$, and its edge set is

$$E(Q_n) = \{\, \{v, w\} \subseteq V(Q_n) \;:\; v \text{ and } w \text{ differ by exactly one digit} \,\}.$$

The cube graph Q_n is a representation of an n-dimensional hypercube. For example,



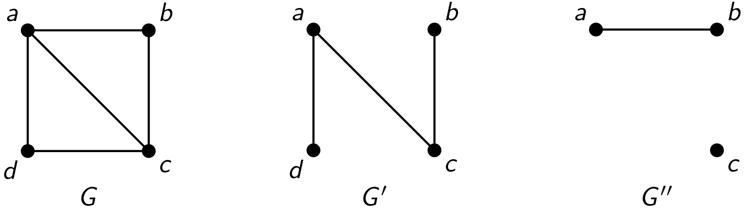
Exercise. How many vertices and edges does Q_n have?

Solution. The number of vertices is $|V(Q_n)| = 2^n$, and the number of edges is $|E(Q_n)| = n2^{n-1}$, which will be explained in Lecture 5.02.

Subgraphs

Definition. Given a graph G, a subgraph of G is a graph H composed of some subset of the vertices and edges of G. That is, $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$, and the endpoints of each edge in $E(H)$ belong to $V(H)$. The same definition also applies for multigraphs.

For example, G' and G'' below are both subgraphs of the given graph G, but G'' is not a subgraph of G'.



Notice that:

- For every integer $n \geq 3$, we have that P_n is a subgraph of C_n.
- For any $n \in \mathbb{Z}^+$, the (unlabelled) cube graph Q_n is a subgraph of the cube graph Q_m for all integers $m \geq n$.
- Every (unlabelled) graph with n vertices is a subgraph of K_n, and indeed a subgraph of K_m for all integers $m \geq n$.

Example – Finding subgraphs

Example. How many subgraphs are there of the graph G with vertex set $V(G) = \{a, b, c\}$ and edge set $E(G) = \{ab, ac, bc\}$?

Solution. Let H be a subgraph of G. We consider different cases for different sizes of $V(H)$.

- If $V(H) = \{a, b, c\}$, then $E(H) \subseteq \{ab, ac, bc\}$, so there are $2^3 = 8$ possible values for $E(H)$.
- If $V(H) = \{a, b\}$, then $E(H) \subseteq \{ab\}$, so there are $2^1 = 2$ possible values for $E(H)$. Similarly, there are two subgraphs with $V(H) = \{a, c\}$ and with $V(H) = \{b, c\}$.
- If $V(H) = \{a\}$, then $E(H) \subseteq \{\}$, so there is $2^0 = 1$ possible value for $E(H)$. Similarly, there is one subgraph with $V(H) = \{b\}$ and with $V(H) = \{c\}$.
- If $V(H) = \{\}$, then $E(H) \subseteq \{\}$, so there is 1 possible value for $E(H)$.

So altogether, there are $8 + 3 \times 2 + 3 \times 1 + 1 = 18$ different subgraphs of G.

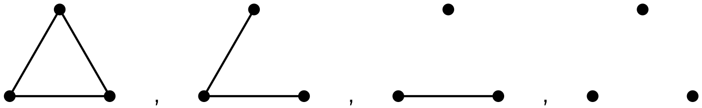
Notice this includes the empty graph and the whole graph G.

Example – Finding unlabelled subgraphs

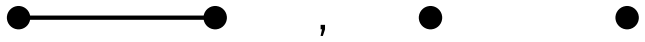
Example. How many subgraphs are there of the unlabelled cyclic graph C_3?

Solution. In this case, it is easier to simply list the subgraphs. They are:

- Subgraphs with exactly 3 vertices:



- Subgraphs with exactly 2 vertices:



- Subgraphs with exactly 1 vertex:

- Subgraphs with exactly 0 vertices:

(empty graph)

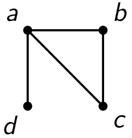
So there are $4 + 2 + 1 + 1 = 8$ subgraphs of the unlabelled cyclic graph C_3.

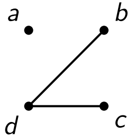
Graph complement

Definition. The complement of a graph G, written as $\overline{G}$, is the graph with the same vertex set as G, but with an edge set whose edges have endpoints v and w if and only if $vw \notin E(G)$. That is, $V(\overline{G}) = V(G)$ and $E(\overline{G}) = E(K_n) - E(G)$ where $n = |V(G)|$.

Note the graph complement is so-called because if a graph G has n vertices, then setting the universal set of edges to be $\mathcal{U} = E(K_n)$, we have that $E(\overline{G}) = (E(G))^c$.

For example, a graph G and its complement $\overline{G}$ are:





Example. If G is a graph with n vertices and m edges, how many vertices and edges does its complement graph $\overline{G}$ have?

Solution. We have $|V(\overline{G})| = |V(G)| = n$, and since G is a subgraph of K_n,

$$|E(\overline{G})| = |E(K_n) - E(G)| = |E(K_n)| - |E(G)| = \binom{n}{2} - m.$$

MATH1081 – Discrete Mathematics

Topic 5 – Graph theory
Lecture 5.02 – Vertex degree and isomorphisms

Lecturer: Dr Sean Gardiner – sean.gardiner@unsw.edu.au

Vertex degrees

Definition. Given a (multi)graph G, the degree of any vertex v in G, written as $\deg(v)$, is the number of times it is incident with an edge in G.

Notice that in a graph G, the degree of a vertex v is just the number of edges incident with it. For a multigraph G, the degree of a vertex v is the number of edges incident with it, except that loops must be counted twice.

Definition. A vertex of degree 0 is called an isolated vertex. A vertex of degree 1 is called a pendant vertex.

Example. Describe the degrees of the vertices in the following graphs.

- P_n has 2 vertices of degree 1, and $n-2$ vertices of degree 2.
- C_n has n vertices of degree 2.
- K_n has n vertices of degree $n-1$.
- $K_{m,n}$ has m vertices of degree n, and n vertices of degree m.
- Q_n has 2^n vertices of degree n.

Definition. A regular (multi)graph is a (multi)graph whose vertices all have the same degree. The graphs C_n, K_n, $K_{n,n}$, and Q_n (for appropriate values of n) are all examples of regular graphs.

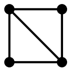
Handshaking lemma

Theorem. (Handshaking lemma)
For any (multi)graph G, the sum of the degrees of all the vertices in G equals twice the number of edges in G. That is, for any multigraph G,

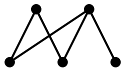
$$\sum_{v \in V(G)} \deg(v) = 2\,|E(G)|.$$

Proof. Each edge in a multigraph G is either incident with exactly two vertices, or in the case of a loop, exactly one vertex twice. Thus summing the degrees of every vertex in G counts each edge in G exactly twice.

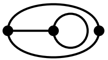
For example, check the vertex degree sums of the following multigraphs:



$3+2+3+2 = 2 \times 5.$



$2+3+2+2+1 = 2 \times 5.$



$3+3+2 = 2 \times 4.$

Example. How many edges does Q_n have?

Solution. We have seen that $|V(Q_n)| = 2^n$ and that the degree of each vertex in Q_n is n. So by the handshaking lemma, we have

$$|E(Q_n)| = \frac{1}{2} \sum_{v \in V(Q_n)} \deg(v) = \frac{1}{2}(n2^n) = n2^{n-1}.$$

Degree sequences

Definition. A vertex degree sequence of a (multi)graph is a sequence of the degrees of all vertices of the graph. It is usually written in increasing or decreasing order.

Example. Does a graph or multigraph with the vertex degree sequence $3, 3, 2, 2, 2, 1$ exist?

Solution. If such a multigraph existed, its vertex degree sum would be $3 + 3 + 2 + 2 + 2 + 1 = 13$, which is odd. But by the handshaking lemma, this total must be even, since it should equal twice the number of edges in the multigraph. So no such multigraph exists.

Fact. No multigraph has a vertex degree sequence whose sum is odd.

Proof. This is a direct result of the handshaking lemma.

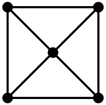
Fact. Every multigraph has an even number of vertices of odd degree.

Proof. If a multigraph had an odd number of vertices of odd degree, its vertex degree sum would be odd, which is impossible.

Example – Degree sequences

Example. Does a graph or multigraph with the vertex degree sequence $4, 3, 3, 3, 3$ exist?

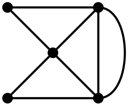
Solution. One such graph (which is also a multigraph) is given below.



Example. Does a graph or multigraph with the vertex degree sequence $4, 4, 4, 2, 2$ exist?

Solution. Label the vertices with degrees $4, 4, 4, 2, 2$ as v_1, v_2, v_3, v_4, v_5 respectively. If a graph with these vertices were to exist, we would require each of v_1, v_2, and v_3 to be adjacent to each of the other 4 vertices. This would mean v_4 and v_5 would each have degree at least 3, which is greater than their expected degree of 2. So no such graph exists.

However, a multigraph with the degree sequence does exist, for example:



Degree sequences for graphs and multigraphs

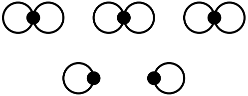
Fact. Not every vertex degree sequence whose sum is even has a corresponding graph.

Fact. Every vertex degree sequence whose sum is even has a corresponding multigraph.

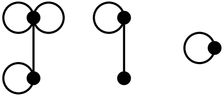
Proof. Since the vertex degree sum is even, there must be an even number of vertices of odd degree. Partition the vertices of odd degree into pairs, and construct a single edge between each pair. All vertices now have degree that differs from their expected degree by an even number. If a vertex has degree differing from its expected degree by $2k$ for some integer k, construct k loops at that vertex. Once completed, we have constructed a multigraph with the required vertex degree sequence.

Example. Find multigraphs with the following degree sequences.

$4, 4, 4, 2, 2.$

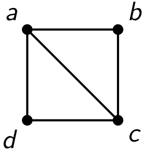


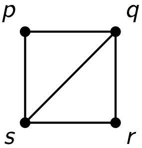
$5, 3, 3, 2, 1.$

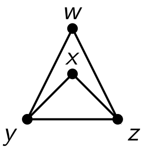


Graph isomorphism

We have seen that graphs can be represented diagrammatically in many different ways, but are always considered the same graph. We might also wish to treat two graphs that have the same shape but different vertex or edge labellings as equivalent in some way, like the following examples:







Definition. Two graphs G and H are isomorphic, written $G \simeq H$, if and only if there exists a bijective function $f : V(G) \to V(H)$ such that vertex adjacency is preserved, that is, for all $v, w \in V(G)$, we have that $\{v, w\} \in E(G)$ if and only if $\{f(v), f(w)\} \in E(H)$.

Equivalently, $G \simeq H$ if and only if there exists a bijective function $f : V(G) \to V(H)$ such that $V(H) = f(V(G))$ and

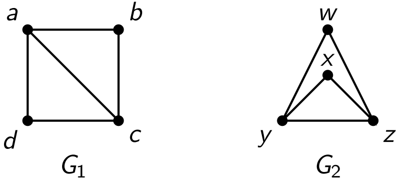
$$E(H) = \{\, \{f(v), f(w)\} \;:\; \{v, w\} \in E(G)\,\}\,.$$

If two graphs are isomorphic, we call the function f a graph isomorphism. Graph isomorphism is an equivalence relation on the set of all graphs.

Example – Isomorphic graphs

To prove that two graphs are isomorphic, we must provide a graph isomorphism between them.

Example. Prove that the following two graphs are isomorphic.



Solution. Consider the function $f : V(G_1) \to V(G_2)$ defined as follows.

$v_i \in V(G_1)$	a	b	c	d
$f(v_i)$	y	w	z	x

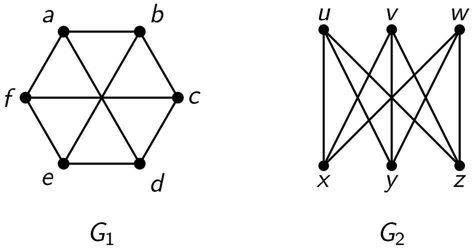
The edges of G_1 under this mapping f are given by the following table.

$v_i v_j \in E(G_1)$	ab	ac	ad	bc	cd
$f(v_i)f(v_j)$	yw	yz	yx	wz	zx

Since the second row is precisely $E(G_2)$, we can conclude that f is a graph isomorphism, and so $G_1 \simeq G_2$.

Example – Isomorphic graphs

Example. Are the following two graphs isomorphic?



Solution. Consider the function $f : V(G_1) \to V(G_2)$ defined as follows.

$v_i \in V(G_1)$	a	b	c	d	e	f
$f(v_i)$	u	x	v	y	w	z

The edges of G_1 under this mapping f are given by the following table.

$v_i v_j \in E(G_1)$	ab	bc	cd	de	ef	fa	ad	be	cf
$f(v_i)f(v_j)$	ux	xv	vy	yw	wz	zu	uy	xw	vz

Since the second row is precisely $E(G_2)$, we can conclude that f is a graph isomorphism, and so $G_1 \simeq G_2$.

Graph invariants

To decide whether two graphs are isomorphic, we can try identifying properties or features that are common to both graphs.

Definition. Any property of a graph G that is also a property of any graph isomorphic to G is called a graph invariant.

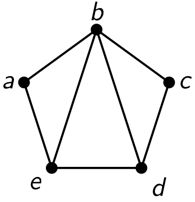
Graph invariants include:

- Number of vertices; number of edges.
- Degree sequence; regularity; number of vertices of a particular degree.
- Degrees of adjacent vertices.
- Properties of subgraphs.
- Bipartiteness.
- Connectedness (see Lecture 5.03).
- Number of paths or cycles of a particular length (see Lecture 5.03).
- Existence of Euler or Hamilton walks (see Lecture 5.03).
- Planarity (see Lecture 5.04).

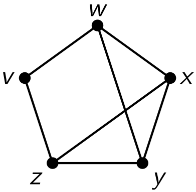
Example – Non-isomorphic graphs

To prove that two graphs are not isomorphic, we can identify an invariant property that one graph has and the other does not.

Example. Prove that the following two graphs are not isomorphic.



G_1



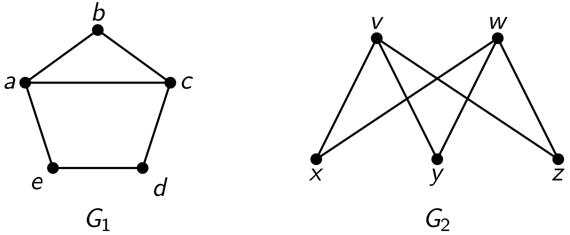
G_2

Solution. In G_1, the vertex b has degree 4, while in G_2, there is no vertex of degree 4. Since vertex degrees are a graph invariant, we can conclude that G_1 and G_2 are not isomorphic.

We could also have used the fact that G_1 has two vertices of degree 2 while G_2 only has one, or that G_1 has two vertices of degree 3 while G_2 has four.

Example – Non-isomorphic graphs

Example. Are the following two graphs isomorphic?



Solution. In G_1, the only two vertices of degree 3 (a and c) are adjacent, while in G_2, the only two vertices of degree 3 (v and w) are not adjacent. Since the degrees of adjacent vertices are a graph invariant, we can conclude that G_1 and G_2 are not isomorphic.

Alternate solution. Notice that G_2 is clearly bipartite (with vertex partitions $\{v, w\}$ and $\{x, y, z\}$). However, if G_1 were bipartite with vertex partitions V_1 and V_2, supposing vertex b belonged to V_1, then both a and c would have to belong to V_2 since they are both adjacent to b, which is a contradiction since a and c are adjacent. Since bipartiteness is a graph invariant, we can conclude that G_1 and G_2 are not isomorphic.

MATH1081 – Discrete Mathematics

Topic 5 – Graph theory
Lecture 5.03 – Standard, Euler, and Hamilton walks

Lecturer: Dr Sean Gardiner – sean.gardiner@unsw.edu.au

Walks

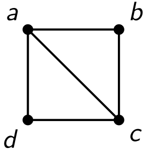
Definition. A walk of length n in a (multi)graph G is any alternating sequence of vertices $v_i \in V(G)$ and edges $e_i \in E(G)$ of the form $v_0, e_1, v_1, e_2, v_2, \ldots, v_{n-1}, e_n, v_n$ such that every pair of consecutive terms in the sequence are incident (that is, every edge e_i has endpoints v_{i-1} and v_i).

We say such a walk starts at v_0 and ends at v_n, or it travels from v_0 to v_n.

Definition. A walk in a (multi)graph G is closed if it starts and ends at the same vertex (that is, if it travels from v_0 to v_0).

Notation. In a graph G, since there cannot be any parallel edges, instead of representing a walk by the sequence $v_0, e_1, v_1, e_2, v_2, \ldots, v_{n-1}, e_n, v_n$ we can use the shorthand $v_0 v_1 v_2 \ldots v_{n-1} v_n$. We can also denote a walk by $W = v_0 v_1 \ldots v_n$, and denote the reverse walk by $W^{-1} = v_n v_{n-1} \ldots v_0$.

Example. Which sequences below represent walks in the given graph?



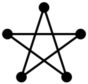
- *abcda* is a closed walk.
- *abcdcda* is a closed walk.
- *abcd* is a non-closed walk.
- *abcadc* is a non-closed walk.
- *abcdba* is not a walk.

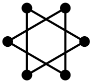
Connected graphs

Definition. A (multi)graph G is connected if and only if for every pair of vertices $v_i, v_j \in V(G)$, there exists a walk from v_i to v_j.

Definition. The maximal connected subgraphs of any (multi)graph G are called its connected components.

For example, the first graph below is connected, while the second is not. They have 1 and 2 connected components respectively.





Theorem. For all (multi)graphs G, the relation $\sim$ on $V(G)$ defined by $v_i \sim v_j$ if and only if there exists a walk from v_i to v_j is an equivalence relation, and its equivalence classes are the connected components of G.

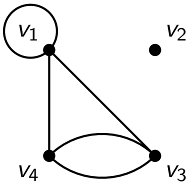
Proof. For all $v \in V(G)$, there exists a walk of length 0 from v to itself, so $v \sim v$ and $\sim$ is reflexive. For all $u, v \in V(G)$, if $u \sim v$ then there is a walk W from u to v, in which case W^{-1} is a walk from v to u, so $v \sim u$ and $\sim$ is symmetric. For all $u, v, w \in V(G)$, if $u \sim v$ and $v \sim w$ then there is a walk W_1 from u to v and a walk W_2 from v to w, in which case $W_1 W_2$ is a walk from u to w, so $u \sim w$ and $\sim$ is transitive.

Adjacency matrices

Recall that a matrix is a rectangular array of entries (usually numbers), and that the entry in the ith row and jth column is called the (i, j) entry of the matrix. If the matrix is denoted by $A = (a_{i,j})$, then its (i, j) entry is $a_{i,j}$.

Definition. For any finite (multi)graph G and particular ordering of its vertices $v_1, v_2, \ldots, v_n$, an adjacency matrix for G is the $n \times n$ matrix A for which each (i, j) entry is the number of edges in G with endpoints v_i and v_j.

For example,



has adjacency matrix $\begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 2 \\ 1 & 0 & 2 & 0 \end{pmatrix}$.

Notice that any adjacency matrix for a graph can only have entries that are 0 or 1 (since there are no parallel edges), and every entry on the main diagonal must be 0 (since there are no loops).

Notice that an adjacency matrix $A = (a_{i,j})$ for any (multi)graph must be symmetric, that is, $a_{i,j} = a_{j,i}$ for all i, j. (An adjacency matrix for a directed (multi)graph might not be symmetric.)

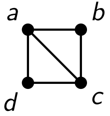
Counting walks

Recall that the product of the $n \times n$ matrices $A = (a_{i,j})$ and $B = (b_{i,j})$ is the matrix $C = (c_{i,j})$ whose entries are given by

$$c_{i,j} = \sum_{k=1}^{n} a_{i,k} b_{k,j} = a_{i,1} b_{1,j} + a_{i,2} b_{2,j} + \cdots + a_{i,n} b_{n,j}.$$

Theorem. Given a (multi)graph G with adjacency matrix $A = (a_{i,j})$ for vertex ordering $v_1, v_2, \ldots, v_n$, the number of walks of length m from v_i to v_j is the (i,j) entry of A^m (sometimes written as $a_{i,j}^{(m)}$).

Example. How many walks of length 3 are there from a to a, and from a to b, in the following graph?



Solution. Ordering the vertices as a, b, c, d, a corresponding adjacency matrix is $A = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$, so we can find $A^3 = \begin{pmatrix} 4 & 5 & 5 & 5 \\ 5 & 2 & 5 & 2 \\ 5 & 5 & 4 & 5 \\ 5 & 2 & 5 & 2 \end{pmatrix}$.

The $(1,1)$ entry of A^3 is 4, so there are 4 different walks of length 3 from a to a, and since $a_{1,2}^{(3)} = 5$, there are 5 different walks of length 3 from a to b.

Theorem. Given a (multi)graph G with adjacency matrix $A = (a_{i,j})$ for vertex ordering $v_1, v_2, \dots, v_n$, the number of walks of length m from v_i to v_j is the (i,j) entry of A^m (sometimes written as $a^{(m)}_{i,j}$).

Proof. Let $P(m)$ be the statement "The (i,j) entry of A^m is the number of walks of length m from v_i to v_j" for any $m \in \mathbb{N}$.

When $m = 0$, we have $A^m = A^0 = I$, the identity matrix whose main diagonal entries are each 1 and whose all other entries are 0. Since there is exactly one walk of length 0 from a vertex to itself, and no walks of length 0 between two distinct vertices, we can conclude that $P(0)$ is true.

Next assume that $P(k)$ is true, so that the (i,j) entry of A^k is the number of walks of length k from v_i to v_j. Consider the matrix $A^{k+1} = (A^k)A$. Then

$$a^{(k+1)}_{i,j} = a^{(k)}_{i,1} a_{1,j} + a^{(k)}_{i,2} a_{2,j} + \cdots + a^{(k)}_{i,n} a_{n,j}.$$

By the inductive hypothesis, each term $a^{(k)}_{i,p} a_{p,j}$ counts the number of walks of length k from v_i to v_p times the number of walks of length 1 from v_p to v_j. This is precisely the number of walks of length $k+1$ from v_i to v_j where the penultimate vertex is v_p, so taking the sum of these terms for all possible v_p must give the total number of walks of length $k+1$ from v_i to v_j. Thus $P(k) \Rightarrow P(k+1)$, and so $P(m)$ is true for all $m \in \mathbb{N}$ by induction.

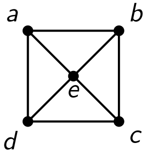
Trails, tours, paths, cycles

Definition. In any (multi)graph:

- A trail is any non-closed walk with no repeated edges.
- A tour is any closed walk with no repeated edges.
- A path is any trail with no repeated vertices, that is, a non-closed walk with no repeated (edges or) vertices.
- A cycle is any tour with no repeated vertices other than its first and last vertices, that is, a closed walk with no repeated edges or vertices (except for its first/last vertex).

We can think of a trail or tour as a walk that can only intersect with itself at vertices, and a path or cycle as a walk that cannot intersect with itself anywhere (except at its start and end vertex in the case of a cycle).

Example. Classify each of the below walks in the given graph.



- *abcde* is a path (and a trail).
- *aecbed* is a trail (but not a path).
- *abcdea* is a cycle (and a tour).
- *aecbeda* is a tour (but not a cycle).
- *abedceb* is a walk (but not a trail or tour).

Proofs involving paths/cycles

Theorem. In a (multi)graph G, if there is a non-closed walk from vertices v_0 to v_n, then there is a path from vertices v_0 to v_n.

Proof. Suppose there is a walk $v_0, e_1, v_1, e_2, v_2, \dots, v_{n-1}, e_n, v_n$ in G. If none of the vertices are repeated, then the walk is already a path. Otherwise, suppose $v_i = v_j$ for some $i < j$. Then by removing all steps in the walk between v_i and v_j, we obtain a smaller walk $v_0, \dots, v_{i-1}, e_i, v_i, e_j, v_{j+1}, \dots, v_n$, which is valid since $v_i = v_j$ is an endpoint of e_j.

This process can be repeated until there are no repeated vertices remaining in the walk, reducing the walk to a path from v_0 to v_n.

Corollary. In a (multi)graph G with n vertices, if there is a non-closed walk from vertices v_i to v_j, then there is a walk of length at most $n-1$ from v_i to v_j.

Proof. If there is a non-closed walk from v_i to v_j, then there is a path from v_i to v_j, which has no repeated vertices and therefore at most n vertices and $n-1$ edges.

Cycles in bipartite graphs

Theorem. A (multi)graph G is not bipartite if and only if it contains a cycle of odd length.

Equivalently, a (multi)graph G is bipartite if and only if every cycle in G has even length.

Proof of forward implication. First suppose G is a bipartite multigraph with vertex set partitions V_1 and V_2. Then any walk in G must alternate vertices between V_1 and V_2. So for any cycle in G, since it must start and end in the same vertex partition, its length must be even.

Proof of backward implication. Now suppose that G is a multigraph for which every cycle in G has even length. For simplicity, we shall assume G is connected – if it is not, a similar approach can be applied to each connected component of G.

Choose some vertex w in G, and construct the sets

$$V_1 = \{v \in V(G) \ : \ \text{there is a path in } G \text{ from } v \text{ to } w \text{ of even length}\},$$
$$V_2 = \{v \in V(G) \ : \ \text{there is a path in } G \text{ from } v \text{ to } w \text{ of odd length}\}.$$

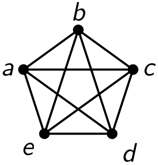
Certainly $V_1 \cup V_2 = V(G)$, since every vertex must be either an odd or even distance from a particular vertex in a connected graph. We also have that $V_1 \cap V_2 = \varnothing$, since if there were some vertex $v \in V_1 \cap V_2$, then there would exist a path P_1 from v to w of even length, and a path P_2 from v to w of odd length, meaning the walk $P_1 P_2^{-1}$ would be a cycle of odd length, which is not possible in G. So V_1 and V_2 partition $V(G)$.

Furthermore, if two vertices $u, v \in V_1$ or $u, v \in V_2$ were adjacent, the walk $w \ldots uv \ldots w$ would be a cycle of odd length, which again is not possible in G. So no two vertices both in V_1 or both in V_2 are adjacent, meaning G is bipartite.

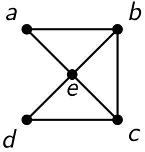
Euler walks

Definition. In any connected (multi)graph G, an Euler trail is a trail that visits each edge of G exactly once, and an Euler tour is a tour that visits each edge of G exactly once.

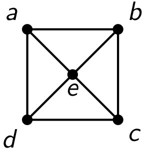
For example, consider the following graphs:



G_1



G_2



G_3

- The graph G_1 has an Euler tour, for example $abcdeacebda$.
- The graph G_2 has an Euler trail, for example $baedcebc$.
- The graph G_3 does not have an Euler walk.

To prove the existence of an Euler walk, we can simply construct one. To prove the non-existence of an Euler walk, we shall establish some necessary and sufficient conditions.

Existence of Euler tours

Theorem. A connected (multi)graph G contains an Euler tour if and only if every vertex has even degree.

Proof of forward implication. First suppose G contains an Euler tour. Then the tour starts and ends at some vertex v_0, and for each vertex v_i that the tour passes through, it must enter that vertex the same number of times it exits it. So the number of times the tour enters or exits each vertex must be an even number, and since the tour passes over every edge of G, this number must also be the vertex's degree. So every vertex has even degree.

Proof of backward implication. First note that it is sufficient to prove the statement only for multigraphs that contain no loops, since if a multigraph G becomes the multigraph G' upon removal of all loop edges, G has n vertices all of even degree if and only if G' does (since each loop edge adds 2 to its endpoint vertex's degree), and G has an Euler tour if and only if G' does (since each loop e can be trivially included in a tour at its endpoint vertex v by inserting the walk vev).

Let $P(n)$ be the statement "Any connected multigraph with n vertices all of even degree contains an Euler tour" for any $n \in \mathbb{Z}^+$. The only multigraph with 1 vertex and no loops has no edges, so it vacuously satisfies the conditions for containing an Euler tour. So $P(1)$ is true.

Existence of Euler tours (continued)

Proof of backward implication (cont'd).

Now suppose that $P(1), P(2), P(3), \dots, P(k)$ are all true for some $k \in \mathbb{Z}^+$. Consider an arbitrary connected multigraph G with $k+1$ vertices all of even degree (with no loops). Choose a vertex w in G and consider all the edges incident with it. Since w has even degree, it must have an even number of incident edges, so they can be arbitrarily labelled $e_1, e_1', e_2, e_2', \dots, e_m, e_m'$ for some $m \in \mathbb{N}$. For each edge e_i, we label its other endpoint v_i, and similarly label the other endpoint of each e_i' as v_i'.

Now consider the graph H formed by removing the vertex w and replacing each pair of edges e_i and e_i' with a new edge e_i'' whose endpoints are v_i and v_i'. Constructing this new graph H does not change the degrees of any of the other vertices in G, so H is a graph with k vertices all of even degree. By the inductive hypotheses, H must contain an Euler tour in each of its connected components. We can convert each Euler tour in H into a tour in G by replacing each instance of $v_i e_i'' v_i'$ (or $v_i' e_i'' v_i$) in the tour with $v_i e_i w e_i' v_i'$ (or $v_i' e_i' w e_i v_i$). So we can construct an Euler tour in G by starting at w and tracing out the Euler tour of each connected component of H, revisiting w between each tour.

Thus $P(k+1)$ is true, and so $P(n)$ is true for all $n \in \mathbb{Z}^+$ by strong induction.

Existence of Euler trails

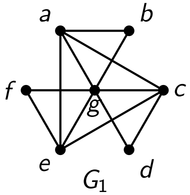
Theorem. A connected (multi)graph G contains an Euler trail if and only if exactly two vertices have odd degree.

Proof. Suppose G contains an Euler trail with start vertex u and end vertex v. Then by adding an edge between u and v, we can create a new graph G' which has an Euler tour. By the previous theorem, every vertex of G' has even degree, so in G every vertex must have had even degree except for u and v (since their degrees are each increased by 1 when the extra edge is added).

Next suppose G has exactly two vertices of odd degree, labelled u and v. Then by adding an edge between u and v, we can create a new graph G' in which every vertex has even degree (since the degrees of u and v are each increased by 1 when the extra edge is added). By the previous theorem, G' must have an Euler tour, and so G must have an Euler trail that starts at u and ends at v.

Example – Euler walks

Example. Decide whether each graph below has an Euler trail or tour.



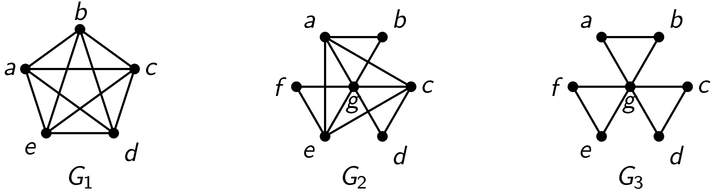
Solution. Since every vertex of G_1 has even degree, an Euler tour exists in G_1. For example, the tour *abgcdgefgacea*.

Since exactly two vertices of G_2 have odd degree (d and e), an Euler trail exists in G_2, and its start and end points must be at d and e in some order. For example, the trail *degdcgbagfe*.

Since more than two vertices of G_3 have odd degree (a, b, c, and f), no Euler path exists in G_3.

Definition. In any connected (multi)graph G, a Hamilton path is a path that visits each vertex of G exactly once, and a Hamilton cycle is a cycle that visits each vertex of G exactly once (except for its first and last vertex).

For example, consider the following graphs:



- The graph G_1 has a Hamilton path, for example *abcde*, and a Hamilton cycle, for example *abcdea*.
- The graph G_2 has a Hamilton path, for example *bacefgd*, but no Hamilton cycle.
- The graph G_3 does not have a Hamilton walk.

To prove the existence of a Hamilton walk, we can simply construct one. It is generally difficult to prove non-existence of a Hamilton walk, but we can still establish some useful partial results.

Properties of Hamilton walks

Unlike for Euler walks, there is no known necessary and sufficient condition for a graph to contain a Hamilton walk. The following partial results are provided without proof.

Fact. If any vertex v in a connected (multi)graph G has degree 1, there is no Hamilton cycle, and if there is a Hamilton path, it must start or end at v.

Fact. If any vertex v in a connected (multi)graph G has degree 2, then both edges incident with v must be part of a Hamilton cycle if it exists.

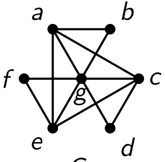
Fact. If a Hamilton cycle exists for a (multi)graph G, every vertex of G must be incident with exactly 2 edges that are used in the walk. If a Hamilton path exists for a (multi)graph G, exactly two vertices of G must be incident with exactly 1 edge that is used in the walk, and all other vertices of G must be incident with exactly 2 edges that are used in the walk.

Theorem. (Dirac) If a graph G with $n \geq 3$ vertices has the property that every vertex has degree at least $\frac{n}{2}$, then G contains a Hamilton cycle.

Theorem. (Ore) If a graph G with $n \geq 3$ vertices has the property that every pair of non-adjacent vertices has degree sum at least n, then G contains a Hamilton cycle.

Example – Hamilton walks

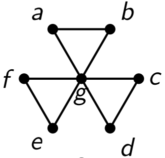
Example. Prove that G_2 cannot have a Hamilton cycle.



G_2

Solution. Suppose a Hamilton cycle exists for G_2. Any Hamilton cycle must pass through both edges incident with each vertex of degree 2. Since b has degree 2, edges ab and bg must be included in the Hamilton cycle. Since d has degree 2, edges dc and dg must be included in the Hamilton cycle. Since f has degree 2, edges fe and fg must be included in the Hamilton cycle. But this means g is incident with more than 2 edges in the Hamilton cycle, which is impossible. So no Hamilton cycle exists for G_2.

Example. Prove that G_3 cannot have a Hamilton walk.



G_3

Solution. Suppose a Hamilton walk exists for G_3. There are 6 vertices of degree 2, and any Hamilton walk must pass through both edges incident with at least 4 of these vertices. By the pigeonhole principle (with the 3 pigeonholes $\{a, b\}$, $\{c, d\}$, and $\{e, f\}$), at least 2 of these vertices must be adjacent, for example a and b. But then the walk must include edges ab, ag, and bg, which forms a cycle that does not visit every vertex of G_3. So there cannot exist a Hamilton path nor a Hamilton cycle.

MATH1081 – Discrete Mathematics

Topic 5 – Graph theory
Lecture 5.04 – Planar graphs and maps

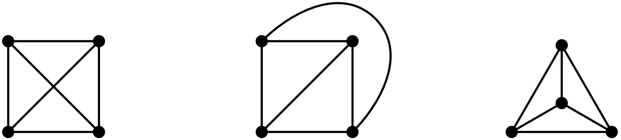
Lecturer: Dr Sean Gardiner – sean.gardiner@unsw.edu.au

Planar graphs

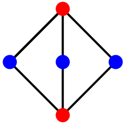
Definition. A (multi)graph G is called planar if it has a diagrammatic representation in the 2-dimensional plane in which no edges cross.

Definition. If a (multi)graph G is planar, a representation of G with no edges crossing is called a planar map or planar representation of G.

For example, all graphs below are representations of K_4, but only the second and third representations are planar maps of K_4 (since in the first representation, two of the edges cross each other).



Example. Is $K_{2,3}$ planar?

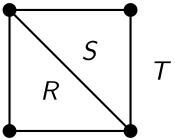


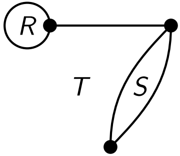
Solution. The graph $K_{2,3}$ is planar, as demonstrated by the planar map provided on the right (each of the 2 red vertices is adjacent to each of the 3 blue vertices, and no edges cross).

Regions

Definition. In any planar map of a planar (multi)graph G, the plane is divided into regions (often labelled $R_1, R_2, R_3, \dots$) whose boundaries are the edges of G. Exactly one of these regions is unbounded, in the sense that it extends infinitely outside of the planar map.

For example, each of the planar multigraphs below has 3 regions labelled R, S, and T, and in both cases T is the unbounded region.





Definition. Given a planar (multi)graph G and a planar map of G, if an edge $e \in E(G)$ lies on the boundary for a region R in the planar map, then we say e is incident with R and R is incident with e.

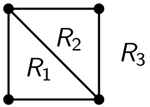
Definition. Given a planar (multi)graph G and a planar map of G, two regions R and S in the planar map are said to be adjacent or neighbours if there is an edge in G that they are both incident with. For example, in the second graph, T is adjacent to R and to S, but R and S are not adjacent.

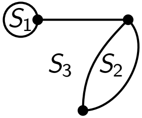
Region degrees

Definition. Given a planar (multi)graph G and a planar map of G, the degree of any region R in the planar map, written as $\deg(R)$, is the number of times it is incident with an edge in the planar map.

One way to evaluate the degree of a region is to count the number of times an edge is passed when traversing its boundary.

Example. Find the degrees of the regions in the following multigraphs.





Solution. We have $\deg(R_1) = \deg(R_2) = 3$, $\deg(R_3) = 4$, $\deg(S_1) = 1$, $\deg(S_2) = 2$, and $\deg(S_3) = 5$.

Example. Describe the degrees of the regions in the following graphs.

- C_n has 2 regions of degree n.
- P_n has 1 region of degree $2(n-1)$.
- K_4 has 4 regions of degree 3.
- K_5 is not planar (to be proven!), so it does not have well-defined regions.

Handshaking lemma

Notice that the region degree sequence for a graph might be different depending on the planar map chosen. For example, the two graphs below are different planar maps of the same planar graph, but the planar maps have region degree sequences $3, 7$ and $5, 5$ respectively.

However, regardless of the planar map chosen, we have an analogue for the handshaking lemma on degrees of regions.

Theorem. (Handshaking lemma for regions)
For any planar (multi)graph G and planar map of G, the sum of the degrees of all the regions in the planar map equals twice the number of edges in G. That is, for any planar multigraph G and planar map of G,

$$\sum_{\text{regions } R} \deg(R) = 2\,|E(G)|.$$

Proof. Each edge in a multigraph G is either incident with exactly two regions, or in the case of an edge that contributes twice to a region's boundary, exactly one region twice. Thus summing the degrees of every region in a planar map of G counts each edge in G exactly twice.

Euler's formula

Lemma. Every connected (multi)graph with at least one edge either has a pendant vertex or contains a cycle (of positive length).

Proof. See Lecture 5.05.

We shall use this lemma to help prove an important result relating the number of regions in a planar map to the number of vertices and edges.

Theorem. (Euler's formula)
Suppose a (multi)graph G is connected and planar, with $v \geq 1$ vertices, e edges, and r regions in a planar map of G. Then $r - e + v = 2$.

Proof. We shall induct on the number of edges e. Let $P(e)$ be the statement "For any connected planar multigraph G with e edges, v vertices, and r regions in a planar map of G, we have $r - e + v = 2$" for any $e \in \mathbb{N}$.

When $e = 0$, since G is connected, it can only have one vertex, so $v = 1$ and $r = 1$. Then $r - e + v = 1 - 0 + 1 = 2$, so $P(0)$ is true.

Now suppose $P(k)$ is true for some $k \in \mathbb{N}$, so that Euler's formula holds for all connected planar graphs with k edges. Consider an arbitrary connected planar multigraph G with $e_G = k + 1$ edges, v_G vertices, and r_G regions in a planar map of G. We wish to show that $P(k + 1)$ holds for G, that is, that $r_G - e_G + v_G = 2$.

Theorem. (Euler's formula)
Suppose a (multi)graph G is connected and planar, with $v \geq 1$ vertices, e edges, and r regions in a planar map of G. Then $r - e + v = 2$.

Proof (cont'd). As per the earlier lemma, first suppose G has a pendant vertex. Then we can remove this pendant vertex and its incident edge to create a new connected planar graph G' with $e_G - 1 = k$ edges, r_G regions, and $v_G - 1$ vertices. Then since $P(k)$ is true for G', we know

$$2 = r_G - (e_G - 1) + (v_G - 1) = r_G - e_G + v_G.$$

So Euler's formula holds for G, meaning $P(k+1)$ is true in this case.

Next suppose G has no pendant vertex, so that by the earlier lemma it must contain a non-trivial cycle. Then we can remove one of the edges from this cycle to create a new connected planar graph G' with $e_G - 1 = k$ edges, $r_G - 1$ regions, and v_G vertices. Then since $P(k)$ is true for G', we know

$$2 = (r_G - 1) - (e_G - 1) + v_G = r_G - e_G + v_G.$$

So again Euler's formula holds for G, meaning $P(k+1)$ is true in this case.

Thus in all possible cases we have $P(k) \Rightarrow P(k+1)$, and so $P(e)$ is true for all $e \in \mathbb{N}$ by induction.

Planarity conditions

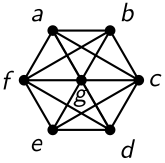
Theorem. For any connected planar graph G with $v \geq 3$ vertices and e edges, we have $e \leq 3v - 6$.

Proof. Let G be a connected planar graph with $v \geq 3$ vertices, e edges, and r regions in a planar map of G. Since G is a graph, it contains no loops, which means in a planar map of G, there can be no regions of degree 1. Similarly, since G is a graph, it contains no parallel edges, which means in a planar map of G, there can be no regions of degree 2. Hence every region R in a planar map of G has degree at least 3, so the sum of all region degrees in a planar map of G is at least $3r$. Using the handshaking lemma for regions, we find $2e = \sum_R \deg(R) \geq 3r = 3(e - v + 2)$ by Euler's formula. This rearranges to give $e \leq 3v - 6$ as required.

Theorem. For any connected planar graph G with $v \geq 3$ vertices and e edges which contains no cycles of length 3, we have $e \leq 2v - 4$.

Proof. Again since G is a graph, a planar map of G contains no regions of degree 1 or 2, nor 3 in this case since it contains no cycles of length 3. So by a similar argument, we have $2e = \sum_R \deg(R) \geq 4r = 4(e - v + 2)$ by Euler's formula, which rearranges to give $e \leq 2v - 4$ as required.

Example. Show that the graph below is not planar.



Solution. The graph is connected, so if it were planar we would require that $e \leq 3v - 6$. However here the graph has $v = 7$ vertices and $e = 16$ edges, so we have $3v - 6 = 15 \ngeq 16 = e$, meaning the graph cannot be planar.

Example. Show that K_5 is not planar.

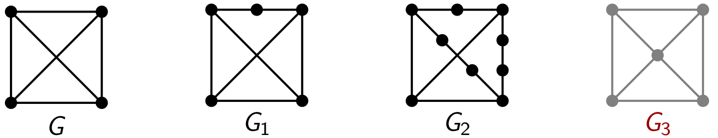
Solution. We know that K_5 is a connected graph with $v = 5$ vertices and $e = \binom{5}{2} = 10$ edges. Since $3v - 6 = 9 \ngeq 10 = e$, we can conclude that K_5 is not planar.

Example. Show that $K_{3,3}$ is not planar.

Solution. We know that $K_{3,3}$ is a connected graph with $3 + 3 = 6$ vertices and $3 \times 3 = 9$ edges. Because $K_{3,3}$ is bipartite, it has no cycles of odd length, and so in particular has no cycles of length 3. So since $2v - 4 = 8 \ngeq 9 = e$, we can conclude that $K_{3,3}$ is not planar.

Kuratowski's theorem

Definition. A subdivision of a (multi)graph G is a new multigraph G' that has had additional vertices inserted by replacing some edges with non-intersecting paths of arbitrary length. For example, G_1 and G_2 below are subdivisions of G, but G_3 is not a subdivision of G.



We can create any subdivision of a graph G by repeating the following process: remove an edge uw from $E(G)$, add a new vertex v to $V(G)$, and add new edges uv and vw to $E(G)$.

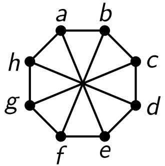
Clearly a (multi)graph G is planar iff any subdivision of G is planar.

Theorem. (Kuratowski)
A (multi)graph is planar if and only if it does not contain a subgraph which is a subdivision of K_5 or $K_{3,3}$.

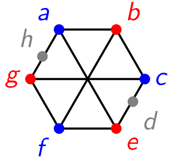
While the forward implication is obvious, the backward implication is very difficult to prove, and so for this course its proof is not provided.

Example – Kuratowski's theorem

Example. Decide whether the graph below is planar.

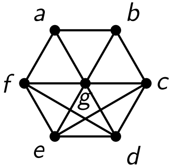


has subgraph

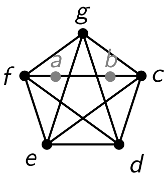


Solution. Consider the subgraph with the edge dh removed. This is a subdivision of $K_{3,3}$, with additional vertices d and h. So the graph is not planar by Kuratowski's theorem.

Example. Decide whether the graph below is planar.



has subgraph

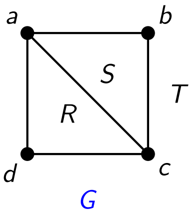


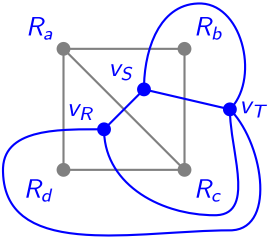
Solution. Consider the subgraph with the edges ag and bg removed. This is a subdivision of K_5, with additional vertices a and b. So the graph is not planar by Kuratowski's theorem.

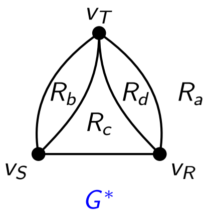
Dual maps

Definition. Given a planar (multi)graph G and a planar map of G, we can construct its dual planar map of a multigraph G^* by doing the following:

- For each region R in the planar map of G, create a corresponding vertex v_R in the planar map of G^*.
- For each vertex v in the planar map of G, create a corresponding region R_v in the planar map of G^*.
- For each edge e in the planar map of G that is incident with regions R and S and incident with vertices v and w, create a corresponding edge e^* in the planar map of G^* that is incident with vertices v_R and v_S and incident with regions R_v and R_w.

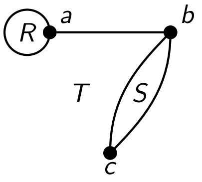


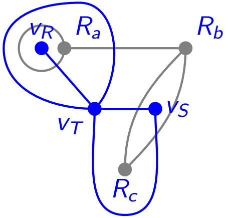


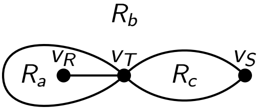


Properties of dual maps

Example. Find the dual planar map of the planar map below.







Fact. Given a planar (multi)graph G and a planar map of G with v vertices, e edges, and r regions, its dual planar map has r vertices, e edges, and v regions.

Fact. Given a planar (multi)graph G and a planar map of G, two vertices v and w are adjacent if and only if their corresponding regions R_v and R_w in the dual planar map are adjacent.

Fact. Given a planar (multi)graph G, the vertex degree sequence of G is the same as the region degree sequence for any dual planar map of G^*, and the vertex degree sequence of any dual graph G^* is the same as the region degree sequence for some planar map of G.

MATH1081 – Discrete Mathematics

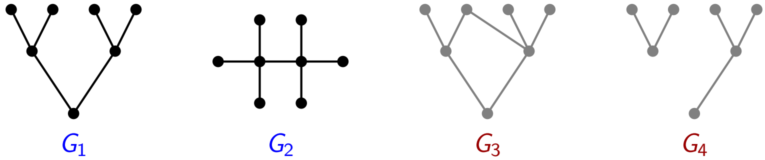
Topic 5 – Graph theory
Lecture 5.05 – Trees and weighted graphs

Lecturer: Dr Sean Gardiner – sean.gardiner@unsw.edu.au

Trees

Definition. A tree is a connected graph which contains no cycles (of positive length).

For example, G_1 and G_2 below are trees, while G_3 and G_4 are not.



Theorem. Any tree with at least one edge contains at least 2 pendant vertices.

Proof. Suppose G is a tree with at least one edge. Since the edge cannot be a loop, G must contain a path between two distinct vertices. Find a path $P = v_0 v_1 \dots v_n$ of maximal length in G. Clearly v_0 has degree at least 1. Suppose v_0 has degree greater than 1. Then v_0 is adjacent to some other vertex u in G. If u is not part of P, then we can extend P to make a longer path $uv_0 v_1 \dots v_n$, which contradicts the maximality of P. If u is part of P, then $v_0 v_1 \dots uv_0$ is a cycle, which contradicts that G is a tree and must contain no non-trivial cycles. So $\deg(v_0) = 1$, and by a similar argument we also have that $\deg(v_n) = 1$, meaning G contains at least 2 pendant vertices.

Number of edges in a tree

Theorem. A connected nonempty multigraph G is a tree if and only if we have $|E(G)| = |V(G)| - 1$.

Proof of forward implication. Let $P(n)$ be the statement "Any tree with n vertices has $n-1$ edges".

The only tree with 1 vertex is the graph with 1 vertex and 0 edges, since it cannot contain any loops (which are cycles), so $P(1)$ is true.

Next suppose $P(k)$ is true for some positive integer k, and consider a tree G with $k+1$ vertices. Because G is a tree, it contains at least two pendant vertices, so consider the new graph G' formed by removing a pendant vertex and its incident edge from G. Then G' has k vertices, and is also a tree since it is still connected and contains no cycles. So by the inductive hypothesis, G' has $k-1$ edges, meaning the original graph G has k edges. Thus we have shown $P(k) \Rightarrow P(k+1)$, meaning $P(n)$ is true for all $n \in \mathbb{Z}^+$ by induction.

Number of edges in a tree

Theorem. A connected nonempty multigraph G is a tree if and only if we have $|E(G)| = |V(G)| - 1$.

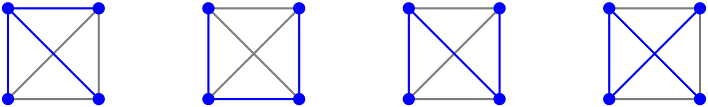
Proof of backward implication. Suppose G is a connected multigraph with $|E(G)| = |V(G)| - 1$. Further suppose that G contains a cycle. Then we may remove this cycle by deleting an edge in the cycle, which reduces the number of edges by 1 but does not change the number of vertices and does not disconnect the graph. Repeating this procedure for all cycles in G, we eventually reach a new graph G' that is connected and has no cycles, and is therefore a tree. By the forward implication proof, we must have $|E(G')| = |V(G')| - 1$. But then we have

$$|E(G)| > |E(G')| = |V(G')| - 1 = |V(G)| - 1,$$

which contradicts the fact that $|E(G)| = |V(G)| - 1$. So G cannot contain a cycle, and is therefore a tree.

Definition. A spanning tree of a (multi)graph G is a tree which is subgraph of G and which contains all vertices of G. That is, a spanning tree T of a multigraph G is a tree for which $V(T) = V(G)$ and $E(T) \subseteq E(G)$.

For example, the following blue graphs are all spanning trees of K_4.



Theorem. Every connected multigraph G has a spanning tree.

Proof. Suppose G is a connected multigraph. For each cycle in G, we can remove it by deleting an edge from the cycle, which does not disconnect the graph. Once all cycles are removed, we are left with a new graph G' that contains all vertices of G, is connected, and has no cycles. So G' is a spanning tree of G.

Weighted graphs

Definition. A weighted (multi)graph is a multigraph whose edges are associated with numbers called weights. Each edge is assigned a particular number, and typically these numbers are positive and rational or natural.

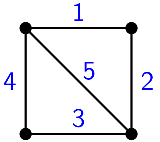
In general, we can say a weighted multigraph G is a multigraph equipped with a weight function $w : E(G) \to \mathbb{R}$.

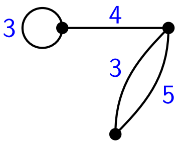
Notation. The weight of an edge e is denoted $w(e)$.

Definition. The weight of a (multi)graph G, sometimes denoted $w(G)$, is the sum of all its edge weights. That is,

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

In a diagrammatic representation of a weighted multigraph, the weight of each edge is indicated by writing a number near the edge. For example, each multigraph below has weight 15.

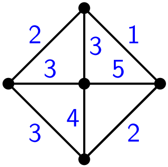




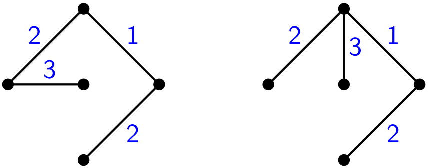
Minimal spanning trees

Definition. A minimal spanning tree of a weighted (multi)graph G is a spanning tree T of G with minimal weight $w(T)$.

Note that for any weighted (multi)graph G, there may not be a unique minimal spanning tree.



For example, the graph above has two minimal spanning trees, each with weight 8:



Kruskal's algorithm

A procedure for finding a minimal spanning tree for any connected weighted (multi)graph G is Kruskal's algorithm:

- Begin with a subgraph T with $V(T) = V(G)$ and $E(T) = \{\}$.
- Sort the edges in G by increasing weight order.
- For each edge in the ordered sequence, add the edge to $E(T)$ if its inclusion will not introduce a cycle in T.
- Once $|E(T)| = |V(G)| - 1$, the process ends and T is a minimal spanning tree of G.

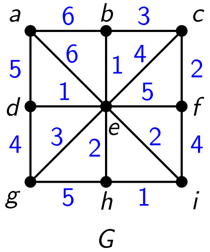
Kruskal's algorithm can be carried out efficiently by filling out a table like the following, with the edges ordered by increasing weight:

edge e_i	weight $w(e_i)$	included?
$\vdots$	$\vdots$	$\vdots$

Example – Kruskal's algorithm

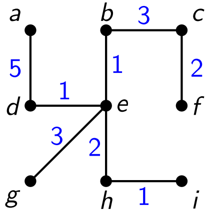
Example. Find a minimal spanning tree for G.

Solution. Applying Kruskal's algorithm gives:



edge e_i	weight $w(e_i)$	included?
be	1	yes
de	1	yes
hi	1	yes
cf	2	yes
eh	2	yes
ei	2	no ($ehie$ is a cycle)
bc	3	yes
eg	3	yes
ce	4	no ($bceb$ is a cycle)
dg	4	no ($degd$ is a cycle)
fi	4	no ($bcfiheb$ is a cycle)
ad	5	yes
ef	5	no (8 edges have been selected)
gh	5	no (8 edges have been selected)
ab	6	no (8 edges have been selected)
ae	6	no (8 edges have been selected)

Thus a minimal spanning tree (with weight 18) is:



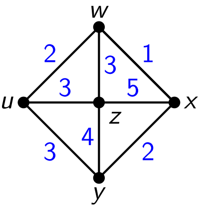
Note that if we had ordered the edges by weight differently, we might have included ei instead of eh and found a different minimal spanning tree.

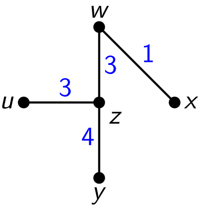
Minimal v-path spanning trees

Definition. Given a weighted (multi)graph G and vertices $u, v \in V(G)$, a shortest path from u to v is a path P from u to v in G with minimal weight $w(P)$. We call the weight of this shortest path the distance from u to v, and write $d(u, v) = w(P)$.

Definition. Given a weighted (multi)graph G and a vertex $v \in V(G)$, a minimal v-path spanning tree is a spanning tree T of G in which for every vertex $u \in V(G)$, the unique path from v to u in T is the shortest path from v to u in G.

Note that for any weighted (multi)graph G and vertex v in G, a minimal v-path spanning tree of G is usually not a minimal spanning tree of G. For example, the graph G on the left has a minimal z-path spanning tree shown on the right.





So in G, we know $d(z, u) = 3$, $d(z, w) = 3$, $d(z, x) = 4$, and $d(z, y) = 4$.

Dijkstra's algorithm

A procedure for finding a minimal v-path spanning tree for any connected weighted (multi)graph G and vertex $v \in V(G)$ is Dijkstra's algorithm:

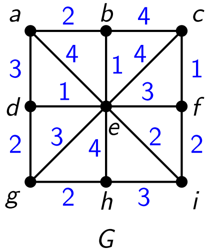
- Begin with a tree T with $V(T) = \{v\}$ and $E(T) = \{\}$.
- Consider all edges $e_i \in E(G)$ that have one endpoint $u_i \in V(T)$ and one endpoint $w_i \notin V(T)$. Out of all these edges, choose the one for which $d(v, u_i) + w(e_i)$ is minimal, and add e_i to $E(T)$ and w_i to $V(T)$.
- Repeat the previous step until $V(T) = V(G)$, at which point the process ends and T is a minimal v-path spanning tree of G.

Dijkstra's algorithm can be carried out efficiently by filling out a table like the following:

possible edges $u_i w_i$ (weight $d(v, u_i) + w(u_i w_i)$)	included edge $u_i w_i$	included vertex w_i	distance $d(v, w_i)$
		v	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$

Example – Dijkstra's algorithm

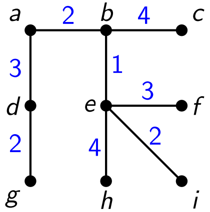
Example. Find a minimal a-path spanning tree for G.



Solution. Applying Dijkstra's algorithm gives:

possible edges $u_i w_i$ (weight $d(a, u_i) + w(u_i w_i)$)	included edge $u_i w_i$	included vertex w_i	distance $d(a, w_i)$
		a	0
ab (2), ad (3), ae (4)	ab	b	2
ad (3), ae (4), bc (6), be (3)	ad	d	3
ae (4), bc (6), be (3), de (4), dg (5)	be	e	3
bc (6), dg (5), ec (7), ef (6), eg (6), eh (7), ei (5)	dg	g	5
bc (6), ec (7), ef (6), eh (7), ei (5), gh (7)	ei	i	5
bc (6), ec (7), ef (6), eh (7), gh (7), if (7), ih (8)	bc	c	6
cf (7), ef (6), eh (7), gh (7), if (7), ih (8)	ef	f	6
eh (7), gh (7), ih (8)	eh	h	7

Thus a minimal a-path spanning tree for the graph is:



Note that if we had included gh instead of eh in the last step, we would have found a different minimal a-path spanning tree.