Contents

1

Graph Structures

1.1 Multigraphs

1.1.1 Multigraphs

Intuitively, we would describe a *graph* as a collection of nodes (or vertices) plus some lines (or edges) joining some nodes together. This can be rigorously defined as follows:

Definition 1.1.1 ▸ Multigraph

A **multigraph** G consists of a non-empty finite set of vertices denoted by $V(G)$ and a finite set of edges denoted by $E(G)$. $|V(G)|$ is known as the **order** of G, denoted by $v(G)$ and $|E(G)|$ is known as the **size** of G, denoted by $e(G)$.

In particular, if $v(G) = m$ and $e(G) = n$, we say that G is an **(m, n)-graph**.

G is said to be **trivial** if $v(G) = 1$ and **non-trivial** otherwise.

Note that in a multigraph, by default there can be a plural number of edges between any two vertices. We would define the notion of *simple graph* as a special multigraph.

Definition 1.1.2 ▸ Simple Graph

A multigraph G is said to be **simple** if there is at most one edge between any two distinct vertices.

Remark. Note that if G is simple and undirected, then

$$E(G) \subseteq \left\{(v_j, v_i) : v_j, v_i \in V(G), j \geq i\right\}.$$

Notice that if (v_i, v_i) is an edge, it connects a vertex to itself. This is known as a *loop*.

In a graph, it is important to know the layout of the vertices and edges. For this purpose, we define the notion of *adjacency* in a multigraph.

Definition 1.1.3 ▶ Adjacency and Neighbourhood

Let $v_i, v_j \in V(G)$, we say that they are **adjacent** if $v_iv_j \in E(G)$. Alternatively, we say that v_i and v_j are **neighbours** to each other. The edge v_iv_j is said to be **incident** with v_i and v_j. Two edges e and f are said to be **adjacent** if there exists some $v \in V(G)$ such that both e and f are incident with v.

The set of all neighbours to some $v_i \in V(G)$ is called the **neighbourhood set** of v_i, denoted by $N_G(v_i)$. In particular, the set $N_G[v_i] \coloneqq N_G(v_i) \cup \{v_i\}$ is known as the **closed neighbourhood set** of v_i.

Alternatively, one may write $v \sim u$ if v and u are adjacent vertices. We can further add on to the definition by discussing the size of the neighbourhood of a vertex.

Definition 1.1.4 ▶ Degree

The **degree** of v, denoted by $d_G(v)$ is defined as the number of edges incident to v. If $d_G(v)$ is even (repectively, odd), then we say that v is an even (respectively, odd) vertex. If $d_G(v) = 0$, we say that v is **isolated**; if $d_G(v) = 1$, we say that v is an **end** vertex.

In particular, we define

$$\Delta(G) = \max_{v \in G} d_G(v), \qquad \delta(G) = \min_{v \in G} d_G(v).$$

In particular, we can denote the average degree of a graph G by

$$\bar{d}(G) = \frac{\sum_{v \in V(G)} d_G(v)}{v(G)}.$$

Remark. Note that a high maximal degree does not imply a high average degree. A classic counter example is a complete graph plus an isolated vertex.

With the notion of degree established, we can now define a *regular graph*.

Definition 1.1.5 ▶ Regular Graph

A **regular graph** is a graph in which every vertex has the same degree. In particular, if $d_G(v) = k$ for all $v \in V(G)$, G is known as a **k-regular graph**.

Remark. A graph G is regular if and only if $\Delta(G) = \delta(G)$.

Since a graph essentially consists of two sets, it is natural to consider the notion of graph complementation. We now proceed to introducing the notion of *complement*.

Definition 1.1.6 ▸ Complement

Let G be a graph of order n, the **complement** of G, denoted by $\overline{G}$, is the graph of order n where

$$V\left(\overline{G}\right) = V(G), \qquad E\left(\overline{G}\right) = \{(u, v) : (u, v) \notin E(G)\}.$$

We discuss the following interesting question:

> 15 students went to a party. During the party some of them shook hands with each other. At the end of the party, the number of handshakes made by each student was recorded and it was reported that the sum was 39. Was this possible?

Note that if we represent each student as a vertex, then we can use $V(G) = \{v_1, v_2, \cdots, v_{15}\}$ to construct a graph, in which an edge $v_i v_j \in E(G)$ if and only if students i and j shook hands. As such, $d_G(v_i)$ is the number of persons student i shook hands with.

Note that if we sum up $d_G(v_i)$ for all the vertices, every edge will be counted exactly twice! This means that

$$\sum_{i=1}^{15} d_G(v_i) = 39$$

is impossible since the left-hand side must be even. In fact, by the above reasoning, we see that in any graph, the sum of degrees of its vertices must be even.

Lemma 1.1.7 ▸ Handshaking Lemma

If G is a graph of order n and size m, then

$$\sum_{i=1}^{n} d_G(v_i) = 2m.$$

Remark. It can be easily deduced from the above lemma that in any graph, the number of vertices with odd degrees must be even.

Relating to Definition 1.1.4, we see that the average degree of a graph G can be computed as

$$\bar{d}(G) = \frac{2e(G)}{v(G)}.$$

As an extension of Lemma 1.1.7, the minimum size of any graph is obviously 0, and the maximum size of a simple graph occurs when there is an edge between any two vertices.

Definition 1.1.8 ▸ Empty and Complete Graphs

Let G be a simple graph of order n. G is said to be an **empty graph** or **null graph**, denoted by 0_n, if $e(G) = 0$, and a **complete graph**, denoted by K_n if for all $u, v \in V(G)$, we have $(u, v) \in E(G)$.

1.1.2 Subgraphs

Since a multigraph is just two sets, we can define a "subset" relation between graphs.

Definition 1.1.9 ▸ Subgraph

Let G, H be graphs, then H is a **subgraph** of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. In particular, a subgraph H is a **proper subgraph** of G if $V(H) \neq V(G)$ or $E(H) \neq E(G)$.

Note that we a graph can possibly be reproduced by connecting its vertices correctly, so a subgraph containing all vertices of a graph "spans" the original graph.

Definition 1.1.10 ▸ Spanning Subgraph

Let H be a subgraph of G. H is called a **spanning subgraph** of G if $V(H) = V(G)$.

Note that by definition, a spanning subgraph retains all vertices of the original graph. Therefore, a way to quickly generate a subgraph of a given graph G is to keep the vertex set and delete some edges from the edge set. We denote such a graph by $H = G - F$ for some $F \subset E(G)$. Observe that

$$V(H) = V(G), \qquad E(H) = E(G) - F.$$

On the other hand, by deleting some vertices together with edges incident to them, we can produce a subgraph from any given graph.

Definition 1.1.11 ▸ Induced Subgraph

Let H be a subgraph of G. H is called a **induced subgraph** of G if

$$E(H) = \{uv \in E(G) : \ u, v \in V(H)\}.$$

Let $S \subseteq V(G)$, the subgraph of G induced by S is denoted by $[S]$.

Alternatively, let $F \subseteq E(G)$. Define

$$V' := \bigcup \{\{u, v\} : uv \in F\},$$

then (V', F) is the subgraph of G induced by F, denoted by $G[F]$.

Intuitively, an induced subgraph pf G consists of a selected subset of the vertices of G together with all edges in G connecting any vertices in this subset.

Using the idea of deletion, we can quickly generate an induced subgraph H of G as follows: first, set $V(H) = V(G) - A$ for some $A \subseteq V(G)$, i.e., remove some vertices; then, we will remove all edges from $E(G)$ which are incident to some vertices in A, i.e., set

$$E(H) = \{e \in E(G) : \ e \text{ is not incident to any } v \in A\}.$$

Then, $H = \big(V(H), E(H)\big)$ is an induced subgraph of G. In fact, we can denote H as

$$H = G - A = [V(G) - A].$$

This leads to the following proposition:

Proposition 1.1.12

Let G be a graph. If $A \subseteq V(G)$, then $G - A = [V(G) - A]$. Suppose H is a subgraph of G, then H is an induced subgraph of G if and only if $H = G - \big(V(G) - V(H)\big)$.

We would like to consider the relations between the properties of a graph to those of its subgraphs. In particular, it follows from intuition that if a graph has a high average degree, then we can naturally find a subgraph with a high minimal degree.

This seems trivial from intuition, as we can always keep deleting vertices with the minimal degree from a graph until we cannot find any vertex whose degree is less than $\frac{1}{2}\bar{d}(G)$, but there is more to that — for instance, how do we know if we would not end up deleting all vertices from the original graph? Thus, to prove our claim is essentially asking to prove the correctness of this greedy deletion, which we shall do in the following proposition.

Proposition 1.1.13

Every non-empty graph G has a subgraph H such that $\delta(H) \geq \frac{1}{2}\bar{d}(G)$.

Proof. Define $H_0 = G$ and $H_{i+1} = H_i - v_i$ where v_i is a vertex in H_i whose degree is the smallest. Note that we can always find this v_i for any H_i with $V(H_i) \neq \varnothing$

because $V(H_i)$ is a finite well-ordered set. We will repeatedly perform the deletion until we obtain some H_k with $d_{H_k}(v_k) \geq \frac{1}{2}\bar{d}(G)$. We shall proceed to proving that this algorithm always terminates with $V(H_k) \neq \varnothing$.

Suppose on contrary that $V(H_k) = \varnothing$, then we have performed the deletion for $v(G)$ times. Suppose we have deleted N edges in total, then clearly,

$$N < v(G)\frac{1}{2}\bar{d}(G) = v(G)\frac{e(G)}{v(G)} = e(G),$$

which means $e(H_k) = e(G) - N > 0$. However, this is impossible since $v(H_k) = 0$, which is a contradiction. □

1.1.3 Graph Isomorphism

Given two graphs G and H, are they the same graph? This seemingly innocent question proves to be extremely hard to answer. Two graphs can look drastically different but be structurally identical in reality. For example, we could shift around the vertices of a graph without changing any edge to alter the shape of the graph dramatically. Therefore, to compare the structures of graphs, we require some rigorous definition.

Definition 1.1.14 ▸ Graph Isomorphism

Two graphs G and H are said to be **isomorphic**, denoted by $G \cong H$, if there exists a bijection $f : V(G) \to V(H)$ such that

$$uv \in E(G) \quad \text{if and only if } f(u)f(v) \in E(H).$$

Remark. One may check that graph isomorphism is obviously an equivalence relation.

Such a bijection f is said to **preserve adjacency**, i.e., if u and v are neighbours in G, then their images are also neighbours in H. In particular, it is possible to map a graph to its own complement via an isomorphism.

Definition 1.1.15 ▸ Self-Complementary Graph

A graph G is said to be **self-complementary** if $G \cong \overline{G}$.

It turns out that a self-complementary graph satisfies some special properties.

Proposition 1.1.16 ▸ Order of Self-Complementray Graphs

If G is a self-complementary graph of order n, then $n = 4k$ or $n = 4k+1$ for some $k \in \mathbb{Z}^+$.

Proof. Since G is self-complementary, $G \cong \overline{G}$ and so $e(G) = e\left(\overline{G}\right)$. Note that

$$e(G) + e\left(\overline{G}\right) = \binom{n}{2} = \frac{n(n-1)}{2},$$

so we have $e(G) = \frac{n(n-1)}{4}$. Since $e(G) \in \mathbb{Z}^+$, either n or $n-1$ is divisible by 4, so $n = 4k$ or $n = 4k+1$ for some $k \in \mathbb{Z}^+$. □

It is very hard to determine whether two specific graphs are isomorphic, but there are some considerations in the general case. For example, some trivial conclusions include:

- Two graphs with different orders cannot be isomorphic.
- Two graphs with different sizes cannot be isomorphic.
- Two graphs with different numbers of components cannot be isomorphic.
- If the numbers of vertices with degree k are different in two graphs, they cannot be isomorphic.

Notice that by Definition 1.1.14, we can define a function g such that $uv \notin E(G)$ if and only if $g(u)g(v) \notin E(H)$ and relate this function to the complement graphs of G and H. Here we introduce a way to determine isomorphism by considering complement graphs, the proof of which is left to the reader as an exercise.

Theorem 1.1.17 ▸ Complementation Preserves Isomorphism

Let G and H be two graphs of the same order. $G \cong H$ if and only if $\overline{G} \cong \overline{H}$.

Now let us think the reverse: if it is not easy to prove isomorphism, can we find a way to quickly determine that two graphs are not isomorphic? Here we present the necessary conditions for isomorphism:

Theorem 1.1.18 ▸ Necessary Conditions for Isomorphism

If $G \cong H$, then

1. *G and H must have the same order and size.*
2. *$\delta(G) = \delta(H)$ and $\Delta(G) = \Delta(H)$.*
3. *The number of vertices with degree i in G and H is the same for all $i \in \mathbb{N}$.*

The first two conditions are easy to observe. For the third condition, we introduce a tool known as *degree sequences*.

Definition 1.1.19 ▸ Degree Sequence

Let G be a graph of order n. If we label its vertices by $v_1, v_2, \cdots, v_n$ such that

$$d(v_1) \geq d(v_2) \geq \cdots \geq d(v_n),$$

then the non-increasing sequence $\big(d(v_n)\big)$ is known as the **degree sequence** of G.

Now we consider the following question:

> Let (d_n) be a non-increasing sequence, is there some graph whose degree sequence is (d_n)?

We will make use of the following definition:

Definition 1.1.20 ▸ Graphic Sequence

Let (d_n) be a sequence of non-negative integers at most $n-1$. (d_n) is said to be **graphic** if there exists a graph G whose degree sequence is (d_n).

To determine whether a sequence is graphic by eye power is difficult. Fortunately, we have the following recursive algorithm:

Theorem 1.1.21 ▸ Havel-Hakimi Algorithm

Let (d_n) be a non-increasing sequence of non-negative integers at most $n - 1$. Define

$$d_m^* = \begin{cases} d_{m+1} - 1 & \text{if } 1 \leq m \leq d_1 \\ d_{m+1} & \text{if } d_1 + 1 \leq m \leq n - 1 \end{cases}.$$

(d_n) is graphic if and only if (d_m^*) is graphic.

Proof. Suppose (d_n) is graphic,we consider the following lemma:

Lemma 1.1.22

For any graphic sequence (d_n)*, there is some graph* G *of order* n *with* $d_G(v_i) = d_i$ *for* $i = 1, 2, \cdots, n$ *such that* v_1 *is adjacent to* v_j *for* $j = 2, 3, \cdots, d_1 + 1$.

Proof. Suppose on contrary there is no such a graph G, then for any graph G with degree sequence (d_n), v_1 is adjacent to at most $d_1 - 1$ vertices in

$$A := \{v_j : j = 2, 3, \cdots, d_1 + 1\}.$$

Thus, there exists some $v_j \in A$ such that $v_1 v_j \notin E(G)$. However, $d_G(v_1) = d_1$, so there exists some $v_k \in V(G) - A$ with $v_k \neq v_1$ such that $v_1 v_k \in E(G)$.

Notice that since $v_k \notin A \cup \{v_1\}$, $k > j$ and so $d_G(v_k) = d_k \leq d_j = d_G(v_j)$. Therefore, v_j has at least as many neighbours as v_k, which means there must exists some v_r with $v_j v_r \in E(G)$ such that $v_k v_r \notin E(G)$.

Define a graph G' by

$$V(G') = V(G), \qquad E(G') = E(G) - v_j v_r - v_1 v_k + v_j v_k + v_1 v_j.$$

Note that $d_{G'}(v_i) = d_G(v_i)$ for all $i = 1, 2, \cdots, n$, so (d_n) is also a degree sequence for G'. However, now v_1 in G' is adjacent to all vertices in A, which is a contradiction. □

By Lemma 1.1.22, we can choose some graph G whose degree sequence is (d_n) such that v_1 is adjacent to d_1 vertices $v_2, v_3, \cdots, v_{d_1+1}$. Let $H = G - v_1$, then for any $u_i \in H$,

$$d_H(u_i) = \begin{cases} d_G(v_{i+1}) - 1 & \text{if } 1 \leq i \leq d_1 \\ d_G(v_{i+1}) & \text{if } d_1 + 1 \leq i \leq n - 1 \end{cases}.$$

Therefore, (d_m^*) is graphic.

Conversely, suppose (d_m^*) is graphic, then there is some graph H of order $n - 1$ such that $d_H(u_i) = d_i^*$ for $i = 1, 2, \cdots, d_1$. Construct a graph $G = H + v_1$ such that $v_i = u_{i-1}$ for $i = 2, 3, \cdots, n$ and v_1 is adjacent to $v_2, v_3, \cdots, v_{d_1+1}$. Therefore,

$$d_G(v_i) = \begin{cases} d_1 & \text{if } i = 1 \\ d^*_{i-1} + 1 & \text{if } 2 \le i \le d_1 + 1 \\ d^*_i & \text{if } d_1 + 2 \le i \le n \end{cases}.$$

Note that $d_G(v_i) = d_i$, so (d_n) is graphic. □

By Theorem 1.1.21, we can reduce a sequence recursively until we reach some obvious case. This obvious case is graphic if and only if the original sequence is graphic. We can also build the graph corresponding to the original sequence from a simple graphic sequence.

Alternatively, the following is another algorithmic way to determine whether a sequence is graphic.

Theorem 1.1.23 ▸ Erdos-Gallai Algorithm

A non-increasing sequence $(d_1, d_2, \cdots, d_n)$ is graphic if and only if $\sum_{i=1}^n d_i$ is even and for all $k = 1, 2, \cdots, n$,

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

We will not give the full proof but only briefly justify the "only if" direction. Let G be any graph. Define $d_i = d_G(v_i)$ where $v_i \in V(G)$ for $i = 1, 2, \cdots, n$, then $(d_1, d_2, \cdots, d_n)$ is graphic. Note that $\sum_{i=1}^n d_i$ is obviously even by Lemma 1.1.7.

Take some $k \in \mathbb{Z}^+$ with $k \le n$ and let $S = \{v_1, v_2, \cdots, v_k\}$, then by Lemma 1.1.7 again,

$$\begin{aligned}\sum_{i=1}^k d_i &= 2e(G[S]) + e_G\big(S, V(G) - S\big) \\ &\le k(k-1) + e_G\big(S, V(G) - S\big),\end{aligned}$$

where $e_G\big(S, V(G) - S\big)$ is the number of edges in G which connect a vertex in S to a vertex in $V(G) - S$. Notice that

$$\begin{aligned} e_G\big(S, V(G) - S\big) &= \sum_{i=k+1}^n |\{u \in S : uv_i \in E(G)\}| \\ \sum_{i=k+1}^n &\le \min\{k, d_i\}, \end{aligned}$$

and so it follows that

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

Remark. Both Theorems 1.1.21 and 1.1.23 are $\mathcal{O}(n^2)$ algorithms.

1.2 Graph Traversal

1.2.1 Paths and Cycles

We are often interested in the "connectedness" in a graph. In particular, we would wish to study how to traverse a graph from some vertices to others.

Definition 1.2.1 ▶ Walk, Trail, Path

Let G be a graph and $x, y \in V(G)$, then an **x-y walk** is an alternating sequence

$$W := v_0 e_1 v_1 e_2 v_2 \cdots v_{n-1} e_n v_n$$

where $v_i \in V(G)$, $e_i = v_{i-1}v_i \in E(G)$, $x = v_0$ and $y = v_n$. In particular, W is **open** if $v_0 \neq v_n$ and **closed** otherwise. The **length** of a walk is defined as the number of edges in it.

If $e_i \neq e_j$ whenever $i \neq j$, then W is called a **trail**. A closed trail is called a **circuit**. If $v_i \neq v_j$ whenever $i \neq j$, then W is called a **path**. A closed path is called a **cycle**.

Remark. All paths are trails and all trails are walks, but the converses are not true.

In more informal terms, a trail is a walk with no repeated edge and a path is a trail with no repeated vertex.

Note that a path is essentially a graph. We denote a path of order n by P_n and a cycle of order n by C_n. It is easy to see that all cycles are circuits and all circuits are all closed walks, but the converses are not true. However, if we have a circuit, we can guarantee that this circuit contains at least one cycle.

Intuitively, all walks can be reduced to a path by removing duplicated edges and vertices repeatedly.

Proposition 1.2.2 ▸ Any Walk Contains A Path

If a graph G contains a u-v walk of length k, then G contains a u-v path of length at most k.

Proof. Let S be the set of all u-v walks in G, Note that $S \neq \varnothing$ since there is a u-v walk of length k. Let P be a walk of the shortest length. We claim that P must be a path.

Suppose P is not a path, then there exists some vertices $w_1 = w_2$ in P. Suppose Q is obtained by removing the w_1-w_2 walk from P, then Q is a walk. However, Q is shorter than P which is not possible. Therefore, P must be a path.

Note that the length of P is at most k, so G contains a u-v path of length at most k. □

We also have the following relevant definitions with regard to cycles:

Definition 1.2.3 ▸ Girth and Circumference

Let G be a graph. The **girth** of G is the size of the shortest non-trivial cycle in G and the **circumference** of G is the size of the longest non-trivial cycle in G.

If G is acyclic, we define the girth of G to be ∞ and the circumference to be 0. In some special graphs, we realise that we can traverse through every vertex and return to the starting point in a single traversal. Such graphs are said to be *Hamiltonian*.

Definition 1.2.4 ▸ Hamiltonian Graph

A graph G is **Hamiltonian** if it contains a cycle of size $v(G)$.

1.2.2 Connected Components

An important question we are interested in with graphs is the reachability of a vertex, i.e., given two vertices u and v, we want to know whether we can reach one vertex from another. Intuitively, we can traverse between two vertices if there is a path between them. From here we define the notion of a *connected graph*.

Definition 1.2.5 ▸ Connected Graph

Let G be a graph. Two vertices u and v are **connected** if there is a path between them. G is said to be **connected** if for any $u, v \in V(G)$, there exists a path from u to v. For any $u \in V(G)$, we denote the set of all vertices connected to u (inclusive of u) as $c(u)$.

Intuitively, we of course always can find a connected subgraph of a connected graph, but we can in fact produce a much stronger result.

Proposition 1.2.6 ▸ Labelling of Vertices of Connected Graphs

For every connected graph G, $V(G)$ can be labelled as $\{v_1, v_2, \cdots, v_n\}$ such that the graph $G_i \coloneqq [\{v_1, v_2, \cdots, v_i\}]$ is connected for every $i = 1, 2, \cdots, n$.

Proof. For $i = 1$, we can pick any vertex in $V(G)$ as v_1 and G_1 is always trivially connected.

Suppose we have fixed up to k vertices for some $k = 1, 2, \cdots, n-1$, then $G_1, G_2, \cdots, G_k$ are connected subgraphs of G. Note that $v(G) = n > k$, so there exists some $v \in V(G) - V(G_k)$ such that v and v_1 are connected by a path P. Note that there exists some $u \in V(P)$ such that $u \sim v$ for some $v \in V(G_k)$, so taking $v_{k+1} = u$ guarantees that G_{k+1} is connected. □

Intuitively, if a graph has many edges, we would believe that it is easy to find a long path in it. Note that the notion of "having many edges" can be related to the minimal degree of a graph. More formally, the following result is true:

Proposition 1.2.7 ▸ Finding Paths and Cycles in Graphs

Every graph G with $\delta(G) \geq 2$ contains a path of length $\delta(G)$ and a cycle of length at least $\delta(G) + 1$.

Proof. Let $P = x_0 x_1 \cdots x_k$ be the longest path in G. Note that it suffices to prove that $k \geq \delta(G)$. Suppose that there exists some $v \in N(x_k)$ such that $v \notin V(P)$, then we can find a longer path which is not possible. Therefore, $N(x_k) \subseteq V(P)$. Note that $|N(x_k)| \geq \delta(G)$, so $k \geq \delta(G)$ and so G contains a $P_{\delta(G)} \subseteq P$.

Let $i \in \mathbb{N}$ be such that $x_j x_k \notin E(G)$ for all $0 \leq j < i$. Take $P' = x_i x_{i+1} \cdots x_k$, then $e(P') \geq \delta(G)$ by the previous argument. However, $x_i \sim x_k$, so $P' + x_i x_k$ is a cycle of length at least $\delta(G) + 1$ in G. □

A classic problem regarding graph traversal in connected graphs is the Könisburg Bridge Problem:

> Let A, B, C, D be 4 distinct vertices such that there are 2 edges between A and B, 2 edges between A and C, and 1 edge between A and D, B and D, C and D. Can all the edges be visited exactly once in a single traversal of the graph produced?

This can be generalised as: given a connected multigraph G, does there exist a circuit in G

which contains all the edges of G? Such multigraphs are said to be *Eulerian*.

Definition 1.2.8 ▸ Eulerian Circuit

Let G be a connected multigraph, an **Eulerian circuit** in G is a circuit which uses all edges of G exactly once.

Intuitively, if a graph is Eulerian, then following any Eulerian circuit will visit each of its vertices for an even number of times as there are equally many incoming edges as outgoing edges for every vertex in a circuit. This implies that an Eulerian graph will necessarily force all of its vertices to have even degrees. However, it turns out that actually this is also a sufficient condition for the existence of Eulerian circuits!

Theorem 1.2.9 ▸ Necessary and Sufficient Condition for Eulerian Graphs

A connected multigraph G is Eulerian if and only if every vertex in G has an even degree.

Proof. The necessity is trivial, so we will only prove for sufficiency.

Consider the case where $e(G) = 0$, then the only connected multigraph is the trivial graph consisting of a singleton vertex, which is Eulerian.

Suppose that for all $k < m$ where $m \in \mathbb{Z}^+$, a connected multigraph H with $e(H) = k$ is Eulerian if every vertex in H has an even degree. Let G be a connected multigraph with $e(G) = m > 0$ such that every vertex in G has an even degree, then $\delta(G) \geq 2$. By Proposition **??**, there exists a cycle $S \subseteq G$ where $e(S) \geq 3$.

Let G' be the multigraph such that $V(G') = V(G)$ and $E(G') = E(G) - E(S)$, then clearly, $e(G') < m$ and every vertex in G' has an even degree. Therefore, each connected component of G' is Eulerian.

Let C_1 be a connected component of G' and let $v_1 \in V(C_1) \cap V(S)$. Note that there exists an Eulerian circuit in C_1 which starts and ends with v_1. We can traverse along this circuit and then traverse in the clockwise orientation along S until we reach another vertex in S which is in some connected component of G' and repeat the process. Note that $v_1 \in V(S)$, so we will eventually traverse back to v_1, which gives an Eulerian circuit in G. □

Intuitively, even if a graph G is not Eulerian, it is possible to traverse all edges in G with finitely many trails. This is known as a decomposition of G into trails.

Proposition 1.2.10 ▶ Minimal Number of Trails to Decompose A Multigraph

Let G be a graph with $2k$ vertices having odd degrees for some $k \in \mathbb{N}$, then the minimal number of trails needed to decompose G is $\max\{k, 1\}$.

Proof. The case where $k = 0$ is immediate by Theorem 1.2.9. For $k \geq 1$, suppose on contrary that G can be decomposed with $k-1$ trails labelled as $T_1, T_2, \cdots, T_{k-1}$. Since a trail contains at most 2 vertices with odd degrees, then

$$\left|\left\{v \in \bigcup_{i=1}^{k-1} T_i : d_G(v) \text{ is odd}\right\}\right| \leq 2(k-1) < 2k,$$

which is a contradiction. Therefore, we need at least k trails to decompose G.

Let $v_1, v_2, \cdots, v_{2k}$ be the vertices in G with odd degrees. Let G' be the graph produced by adding the edges $v_1v_2, v_3v_4, \cdots, v_{2k-1}v_{2k}$ to G, then the vertices in G' all have even degrees. Therefore, there exists an Eulerian circuit in G' by Theorem 1.2.9.

Note that for all $i = 1, 2, \cdots, k-1$, $v_{2i-1}v_{2i}$ and $v_{2i+1}v_{2i+2}$ are disjoint, so there exists a v_{2i}-v_{2i+1} trail. Note also that there exists a v_{2k}-v_1 trail. The union of these trails is exactly $G' - v_1v_2 - v_3v_4 - \cdots - v_{2k-1}v_{2k} = G$. □

Note that connectedness is an equivalence relation. Let P be a partition formed by the equivalence classes of G under the connectedness relation, then for any $u, v \in V(G)$, u and v are in the same equivalence class if and only if u and v are connected. Any $c(u) \in P$ induces a subgraph $[c(u)]$.

Definition 1.2.11 ▶ Connected Component

Let G be a graph and R be a relation such that for any $u, v \in V(G)$, uRv if and only if u and v are connected. Let

$$C := V(G)/R$$

be the quotient set, then any $c(u) \in C$ is known as a (connected) **component** of G.

The number of components of G is denoted by $\omega(G)$.

Alternatively, a connected component is a maximally connected subgraph, that is, a connected component of G is a subgraph $H \subseteq G$ such that for all $e \in E(G)$, $H + e$ is not connected and for all $v \in V(G) - V(H)$, $[V(H) \cup \{v\}]$ is not connected.

It can be easily seen that G is connected if and only if $\omega(G) = 1$.

We will now establish a relationship between connectedness and complementation.

Theorem 1.2.12 ▶ Connectedness of Complement

If G is disconnected, then $\overline{G}$ is connected.

Proof. Let $u, v \in V\left(\overline{G}\right)$ be two arbitrary vertices. If $uv \notin E(G)$, then $uv \in \left(\overline{G}\right)$ and so u and v are connected in $\overline{G}$.

If $uv \in E(G)$, since G is disconnected, there exists some $w \in V(G)$ such that $uw, wv \notin E(G)$. Therefore, $uw, wv \in E\left(\overline{G}\right)$. This means that there is a u-v path in $\overline{G}$ and so u and v are connected in $\overline{G}$.

Therefore, $\overline{G}$ is connected. □

Intuitively, we can transform a connected graph into a disconnected one by deleting some vertices or edges. It can be easily observed that in certain graphs, deleting one particular vertex or edge will immediately disconnect the graph.

Definition 1.2.13 ▶ Cut-Vertex, Bridge

Let G be a non-trivial graph. $v \in V(G)$ is called a **cut-vertex** of G if $\omega(G - v) > \omega(G)$. $e \in E(G)$ is called a **bridge** of G if $\omega(G - e) > \omega(G)$.

Note that a graph does not have to be connected to have cut vertices and bridges. Essentially, a cut-vertex (or bridge) is just a vertex (or edge) which disconnects a component upon removal.

Now the next question is how do we identify a cut-vertex or a bridge in a graph? Intuitively, a cut-vertex divides a graph into two portions such that traversal between the two portions has to pass through it.

Theorem 1.2.14 ▶ Cut-Vertex Characterisation

Let G be a graph. $v \in V(G)$ is a cut-vertex if and only if there exists $a, b \in V(G)$ such that v is in every path between a and b.

Proof. Let $v \in V(G)$ be a cut-vertex and consider $H = G - v$, then we can find two non-empty disjoint connected components of H, say H_1 and H_2. Now, take $a \in V(H_1)$ and $b \in V_{(}H_2)$, then they are disconnected. However, a and b are connected in G, which means v is in every a-b path.

We will prove the converse by contrapositive. Suppose v is not a cut-vertex of G, then $G - v$ is connected. Therefore, for any $u, w \in V(G)$, there is a path between them which does not contain v. □

A similar argument can be established for bridges. Furthermore, since we have to pass through the bridge when traversing between the two portions, it can be easily seen that we have to reuse the bridge in order to traverse back.

Theorem 1.2.15 ▸ Bridge Characterisation

Let G be a graph. $e \in E(G)$ is a bridge if and only if e is not contained by any cycle in G.

Proof. We will prove the forward direction by contrapositive. Suppose $e = xy$ is contained in some cycle in G, then there is some x-y path in $G - e$. Suppose $a, b \in V(G)$ are connected in G via a path containing e, this implies that there is some a-b path in $G - e$ and so $G - e$ is connected, which means that e is not a bridge.

We will prove the converse also by contrapositive. Suppose $e = xy$ is not a bridge, the $G - e$ is connected and so there is some x-y path in $G - e$. Let this path be P, then $P \cup \{e\}$ is a cycle in G. □

Combining the two characterisations, we can relate cut-vertices to bridges in the following proposition, the proof of which is left to the reader as an exercise:

Proposition 1.2.16

Let G be a graph. If $uv \in E(G)$ is a bridge and u is not an end vertex, then u is a cut-vertex.

A direct consequence of this is the following corollary:

Corollary 1.2.17

If G is a graph with order at least 3 and G contains a bridge, then G contains a cut-vertex.

1.2.3 Graphs as Metric Spaces

Given a graph G with u, v being two connected vertices, we are interested in the notion of distance between them. Intuitively, if there are multiple paths between u and v, we would take the length of the shortest one to represent their distance.

Definition 1.2.18 ▸ Distance, Eccentricity, Diameter

Let G be a connected graph and let $u, v \in V(G)$. The **distance** between u and v, denoted by $d(u, v)$, is the length of the shortest path between u and v. The **eccentricity** of u is defined to be

$$e(u) = \max_{v \in V(G)} \{d(u, v)\}.$$

The **diameter** of G is defined by

$$\operatorname{diam}(G) = \max_{u \in V(G)} \{e(u)\}.$$

The **radius** of G is defined by

$$\operatorname{rad}(G) = \min_{u \in V(G)} \{e(u)\}.$$

A vertex v is called a **central** vertex if

$$e(v) = \operatorname{rad}(v).$$

The subgraph induced by the set of central vertices of G is known as the **centre** of G.

This also justifies the use of words “diameter” and “radius” in a circle. We can view a circle as a graph consisting of a centre and infinitely many vertices in the circumference, with edges of length r connecting the centre to the circumference. Indeed, in this definition, the farthest vertices will be any two on the circumference with a distance of $2r$, and the nearest vertices will be the centre and any vertex on the circumference with a distance of r.

Intuitively, having a large diameter means that the graph is sparse, i.e., one has to take a very long path to traverse between two vertices. Now let us consider a connected but sparse graph, the above intuition means that we may expect to find that the cycles in this graph are very large.

Proposition 1.2.19 ▸ Graphs with High Girth Have High Diameter

Let G be a graph with girth $\mathrm{g}(G)$. If G contains a cycle, then $\mathrm{g}(G) \leq 2\operatorname{diam}(G) + 1$.

Proof. Suppose on contrary that $\mathrm{g}(G) \geq 2\operatorname{diam}(G) + 2$. Let $C \subseteq G$ be the shortest cycle in G, then we can take $x, y \in V(C)$ such that $d_C(x, y)$ is the greatest. Clearly, we have $d_C(x, y) \geq \operatorname{diam}(G) + 1$.

By Definition 1.2.18, there exists some path P between x and y with $e(P) \leq \operatorname{diam}(G)$. Let $P' \subseteq C$ be the x-y path in C, then $e(P) < e(P')$. This means that P and P' are distinct paths and so there is some $q \in V(P') - V(P)$.

Since P and P' are both x-y walks, $|P \cap P'| \geq 2$. Therefore, there are $p_1, p_2 \in P \cap P'$ which are connected to q, which implies that there is a p_1-p_2 path in P'. Since p_1 and p_2 are connected in P, we have found 2 different p_1-p_2 paths and so there is some cycle $Q \subseteq P \cup P'$. Therefore,

$$\begin{aligned} e(Q) &\leq e(P \cup P') \\ &\leq e(P) + e(P') \\ &\leq \operatorname{diam}(G) + d_C(x, y) \\ &\leq 2d_C(x, y) - 1 \\ &= 2\left\lfloor \frac{e(C)}{2} \right\rfloor - 1 \\ &< e(C), \end{aligned}$$

which is a contradiction. □

For readers with knowledge in real analysis or topology, it is easy to see that an undirected unweighted connected graph is a metric space with distance between vertices as its metric.

Theorem 1.2.20 ▶ Connected Graph as A Metric Space

If G is connected, then $(V(G), d_G)$ is a metric space.

Proof. Clearly, for all $x, y \in V(G)$, $d_G(x, y) \geq 0 \in V(G)$ with $d_G(x, y) = 0$ if and only if $x = y$ and $d_G(x, y) = d_G(y, x)$.
Let U, V be the shortest u-w path and shortest w-v path respectively, then $U + V$ is a u-v walk with length $d(u, w) + d(w, v)$. Since the shortest u-v path has length $d(u, v)$, we have

$$d(u, v) \leq d(u, w) + d(w, v).$$

Therefore, $(V(G), d_G)$ is a metric space. □

An application of Theorem 1.2.20 allows us to establish the following:

Theorem 1.2.21 ▶ Boundedness of Diameter

If G is connected, then

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

Proof. $\mathrm{rad}(G) \leq \mathrm{diam}(G)$ is immediate from Definition 1.2.18. Take $u, v \in V(G)$ with $d(u,v) = \mathrm{diam}(G)$ and take $w \in V(G)$ such that $e(w) = \mathrm{rad}(G)$. Notice that this implies

$$\begin{aligned} d(u,w) &\leq e(w) = \mathrm{rad}(G) \\ d(w,v) &\leq e(w) = \mathrm{rad}(G). \end{aligned}$$

By Theorem **??**,

$$\mathrm{diam}(G) = d(u,v) \leq d(u,w) + d(w,v) = 2\mathrm{rad}(G).$$

□

1.3 Bipartite Graphs

Definition 1.3.1 ▸ Bipartite Graph

A graph G is **bipartite** if $V(G)$ can be partitioned into two disjointed subsets V_1 and V_2 such that every edge of G joins a vertex in V_1 to a vertex in V_2. (V_1, V_2) is known as a **bipartition** of G.

Note that the size of a bipartite graph G with partite sets V_1 and V_2 is just

$$e(G) = \sum_{x \in V_1} d_G(x) = \sum_{y \in V_2} d_G(y).$$

It can be deduced that the maximal size of a bipartite graph is achieved when every vertex in V_1 is connected to every vertex in V_2. To define this rigorously, we first introduce the notion of *join*.

Definition 1.3.2 ▸ Join

Let G_1 and G_2 be two disjoint graphs of order n_1 and n_2 respectively. The **join** of G_1 and G_2, deonoted by $G_1 + G_2$ is the graph such that

$$\begin{aligned} V(G_1 + G_2) &= V(G_1) \cup V(G_2) \\ E(G_1 + G_2) &= E(G_1) \cup E(G_2) \cup \{uv : u \in V(G_1), v \in V(G_2)\}. \end{aligned}$$

Informally, we can describe the join of G_1 and G_2 as a graph which contains all the vertices from G_1 and G_2, all existing edges from G_1 and G_2, and new edges connecting every vertex in G_1 to every vertex in G_2. Thus, we can define a complete bipartite graph using a join.

Definition 1.3.3 ▸ Complete Bipartite Graph

A **complete bipartite graph** is defined as $K_{p,q} = 0_p + 0_q$.

Consider a k-regular bipartite graph. Intuitively, if every vertex in the first partite set must be adjacent to $k > 0$ vertices in the second partite set and vice versa, then it is only natural that the two partite sets have equal cardinalities.

Proposition 1.3.4 ▸ Regular Bipartite Graph

For all $k \in \mathbb{Z}^+$, a k-regular bipartite graph with bipartition (U, V) is such that $|U| = |V|$.

Proof. Note that we have

$$k|U| = e(G) = k|V|.$$

Since $k > 0$, this implies $|U| = |V|$. □

Sometimes, it is not easy to identify the bipartition in a bipartite graph. Therefore, to quickly determine whether a graph is bipartite, we consider the following theorem:

Theorem 1.3.5 ▸ Bipartite Graph Characterisation

A graph is bipartite if and only if it contains no odd cycles.

Proof. Suppose G is bipartite with bipartition (U, V). If G contains no cycle then we are done. Suppose G contains some cycle. Let u_i be the i-th visited vertex from U in the cycle. Note that between u_i and u_{i+1} there exists one and only one vertex from V. Suppose k vertices from U are contained in the cycle, then it is easy to see that there are also k vertices from V contained in the cycle. Therefore, every cycle in G must be even.

Suppose conversely that G contains no odd cycles. Note that it suffices to prove that every component of G is bipartite. Let C be a component of G, then C contains no odd cycles. Take some $w \in V(C)$ and define the sets

$$\begin{aligned} V_1 &\coloneqq \{v : d(v, w) \text{ is even}\} \\ V_2 &\coloneqq \{v : d(v, w) \text{ is odd}\}. \end{aligned}$$

We will prove (V_1, V_2) is a bipartition. Suppose it is not a bipartition, then without loss of generality, let $a, b \in V_1$ be neighbours. Note that there is a path A between w and a of even length and a path B between w and b of even length, so $A \cup B \cup \{ab\}$ is a closed walk of odd length. Consider the following lemma:

Lemma 1.3.6

A closed walk of odd length in a graph contains an odd cycle.

Proof. Let the walk have a length of p. The case where $p = 3$ is trivially true since the only closed walk of length 3 is C_3.

Suppose any closed walk of odd length less than p contains an odd cycle. Let P be a closed walk of length p. If P is a cycle then we are done. If P is not a cycle, then there exist some $w_1 = w_2$ in P. Note that $w_1 = w_2$ is a cut-vertex. So there exists two closed walks Q_1 and Q_2 such that $Q_1 \cap Q_2 = \{w_1\}$. Notice that one of Q_1 and Q_2 must have an odd length and so contains an odd cycle, so P contains an odd cycle. □

By Lemma 1.3.6, $A \cup B \cup \{ab\}$ contains an odd cycle, which is a contradiction. □

1.4 Graphs and Matrices

So far we have been using hand-drawn illustrations to aid us in graph visualisation. However, for large graphs with many vertices and egdes, drawing out the graph structures becomes nearly impossible. Therefore, we need to use some algebraic tools to help us represent a large graph.

Definition 1.4.1 ▸ Adjacency Matrix

Let G be a multigraph with $v(G) = n$, then the **adjacency matrix** denoted by $\boldsymbol{A}(G)$ is an $n \times n$ matrix such that a_{ij} is the number of edges incident to both $v_i, v_j \in V(G)$.

Remark. For an undirected multigraph, $\boldsymbol{A}$ is always symmetric.

Note that an adjacency matrix does not tell us which edges are incident to a particular vertex, so we may consider the following alternative:

Definition 1.4.2 ▸ Incidence Matrix

Let G be a multigraph with $v(G) = n$ and $e(G) = m$, then the **incidence matrix** denoted by $\boldsymbol{M}(G)$ is an $n \times m$ matrix such that

$$m_{ij} = \begin{cases} 1, & \text{if } e_j \text{ is incident with } v_i \\ 0, & \text{otherwise} \end{cases}.$$

Remark. For an undirected multigraph, consider an arbitrary k-th column of $\boldsymbol{M}$. Note that any column will have exactly 2 non-zero entries. Suppose $m_{ik} = m_{jk} = 1$, it means that $e_k = v_i v_j$.

For readers who have learnt about basic data structures in computer science, an incidence matrix is just an adjacency list with each list of neighbours converted to a (non-compact) boolean array.

Note that the sum of the i-th row for both $\boldsymbol{A}$ and $\boldsymbol{M}$ is the degree of vertex i.

Since $\boldsymbol{A}$ and $\boldsymbol{M}$ carries information about adjacencies between vertices, we can use them to draw some conclusions regarding graph traversal.

Proposition 1.4.3 ▸ Counting the Number of Walks between Two Vertices

Let G be a simple graph of order n with adjacency matrix $\boldsymbol{A}$, then the (i, j) entry of $\boldsymbol{A}^k$ is the number of v_i-v_j walks of length k.

Proof. The case where $k = 1$ is trivially true since there is a v_i-v_j walk of length 1 if and only if v_i and v_j are neighbours.

Let the (i, j) entry of $\boldsymbol{A}^m$ be m_{ij}. Suppose m_{ij} is the number of v_i-v_j walks of length m. Then the (i, j) entry of $\boldsymbol{A}^{m+1}$ is given by

$$\sum_{r=1}^{n} m_{ir} a_{rj}.$$

Notice that m_{ir} is the number of v_i-v_r walks of length m and a_{rj} is the number of v_r-v_j walks of length 1, so by Theorem **??**, $m_{ir} a_{rj}$ is the number of v_i-v_j walks of length n that passes through v_r. Therefore, summing up $m_{ir} a_{rj}$ for $r = 1, 2, \cdots, n$ gives the total number of v_i-v_j walks of length $n + 1$. □

Additionally, we have the following corollary:

Corollary 1.4.4 ▸ Connectedness Test

Let G be a graph with $v(G) \geq 2$ and adjacency matrix $\boldsymbol{A}$. Let

$$\boldsymbol{B} = \sum_{i=1}^{n-1} \boldsymbol{A}^i,$$

then G is connected if and only if $b_{ij} \neq 0$ for all $i \neq j$.

As it turns out, the incidence and adjacency matrices are in fact related by a simple equation.

Proposition 1.4.5

Let $\boldsymbol{M}$ and $\boldsymbol{A}$ be the incidence matrix and adjacency matrix of a simple graph G respectively, then

$$\boldsymbol{M}\boldsymbol{M}^{\mathrm{T}} = \begin{bmatrix} d_1 & 0 & 0 & \cdots & 0 \\ 0 & d_2 & 0 & \cdots & 0 \\ 0 & 0 & d_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & d_n \end{bmatrix} + \boldsymbol{A}.$$