CS1231 Part 11 Graphs
Based on lectures by Terence Sim and Aaron Tan
Notes taken by Andrew Tan
AY18/19 Semester 1

These notes are not endorsed by the lecturers, and I have modified them (often significantly) after lectures. They are nowhere near accurate representations of what was actually lectured, and in particular, all errors are almost surely mine.

1 Introduction

In graph theory, a **graph** is a structure amounting to a set of objects in which some pairs of the objects are "related". We can abstract the idea of the objects as **vertices**, and the relations as **edges**.

Mathematically speaking, we define a graph as follows:

A **graph** G consists of 2 finite sets: a nonempty set $V(G)$ of **vertices** and a set $E(G)$ of edges, where each edge is associated with a set consisting of either one or two vertices called its **endpoints**.

An edge is said to **connect** its endpoints; two vertices that are connected by an edge are called **adjacent vertices**. A vertex that is an endpoint of a loop is said to be adjacent to itself.

An edge is said to be **incident on** each of its endpoints, and two edges incident on the same endpoint are called **adjacent edges**.

We write $e = \{v, w\}$ for an edge e incident on vertices v and w.

The **degree** of a vertex v of G, denoted **deg(v)**, equals the number of edges that are incident on v, with an edge that is a loop counted twice. The **total degree** of G is the sum of the degrees of all the vertices of G.

With the definition of the degree of a vertex, we introduce the **Handshake Theorem**: If G is any graph, then the sum of the degrees of all the vertices of G equals twice the number of edges of G. Specifically, if the vertices of G are $v_1, v_2, \ldots, v_n$, where $n \geq 0$, then

The total degree of $G = \deg(v_1) + \deg(v_2) + \ldots + \deg(v_n) = 2\times$ the number of edges of G.

As such, we extend the following corollary: The toal degree of a graph is even.
And the following proposition: In any graph, there are an even number of vertices of odd degree.

1.1 Directed graph

A **directed graph**, or **digraph**, G, consists of 2 finite sets: a nonempty set $V(G)$ of **vertices** and a set $D(G)$ of **directed edges**, where each edge is acssociated with an ordered pair of vertices called its **endpoints**.

If edge e is associated with the pair (v, w) of vertices, then e is said to be the (**directed**) **edge** from v to w. We write $e = (v, w)$.

1.2 Definitions

We extend the following definitions:

> **Simple graph**: A simple graph is an undirected graph that does not have any loops or parallel edges.

Complete graph: A complete graph on n vertices, $n > 0$, denoted K_n, is a simple graph with n vertices and exactly one edge connecting each pair of distinct vertices.

Complete bipartite graph: A complete bipartite graph on (m, n) vertices, where $m, n > 0$, denoted $K_{m,n}$, is a simple graph with distinct vertices $v_1, v_2, \ldots, v_m$, and $w_1, w_2, \ldots, 2_n$ that satisfies the following properties:
For all $i, k = 1, 2, \ldots, m$ and for all $j, l = 1, 2, \ldots, n$,

1. There is an edge from each vertex v_i to each vertex w_j.
2. There is no edge from any vertex v_i to any other vertex v_k.
3. There is no edge from any vertex w_j to any other vertex w_l.

Subgraph: A graph H is said to be a subgraph of graph G if, and only if, every vertex in H is also a vertex in G, every edge in H is also an edge in G, and every edge in H has the same endpoints as it has in G.

2   Trails, paths, and circuits

We extend the following definitions:
Let G be a graph, and let v and w be vertices of G.

Walk: a walk from v to w is a finite alternating sequence of adjacent vertices and edges of G. Thus the walk has the form

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

where the v's represent vertices, the e's represent edges, $v_0 = v, v_n = w$, and for all $i \in \{1, 2, \cdots, n\}$, v_{i-1} and v_i are the endpoints of e_i.

Trivial walk: the trivial walk from v to v consists of the single vertex v.

Trail: A trail from v to w is a walk from v to w that does not contain a repeated edge.

Path: A path from v to w is a trail that does not contain a repeated vertex.

Closed walk: A closed walk is a walk that starts and ends at the same vertex.

Circuit / Cycle: A circuit or cycle is a closed walk that contains at least one edge and does not contain a repeated edge.

Simple circuit / Simple cycle: A simple circuit or simple cycle is a circuit that does not have any other repeated vertex except the first and last.

2.1   Connectedness

Two vertices v and w of a graph G are **connected** if, and only if, there is a walk from v to w.

The graph G is connected if, and only if, given *any* two vertices v and w in G, there is a walk from v to w. Symbolically,

$$G \text{ is connected iff } \forall \text{ vertices } v, w \in V(G), \exists \text{ a walk from } v \text{ to } w.$$

From here, we note the following additional properties:
Let G be a graph.

1. If G is connected, then any two distinct vertices of G can be connected by a path.

2. If vertices v and w are part of a circuit in G and one edge is removed from the circuit, then there still exists a trail from v to w in G.

3. If G is connected and G contains a circuit, then an edge of the circuit can be removed without disconnecting G.

2.1.1 Connected component

A **connected component** of a graph is a connected subgraph of the largest possible size. More formally, A graph H is a connected component of a graph G if, and only if,

1. The graph H is a subgraph of G.

2. The graph H is connected.

3. No connected subgraph of G has H as a subgraph and contains vertices or edges that are not in H.

We can think of any given graph as a union of its connected components.

2.2 Euler circuits

Let G be a graph. An **Euler circuit** for G is a circuit that contains every vertex and every edge of G.

That is, an Euler circuit for G is a sequence of adjacent vertices and edges in G that has at least one edge, starts and ends at the same vertex, uses every vertex of G at least once, and uses every edge of G exactly once.

Also, an **Eulerian graph** is a graph that contains an Euler circuit.

We will introduce a few theorems:

Theorem 10.2.2 (Epp) - If a graph has an Euler circuit, then every vertex of the graph has positive even degree.

Theorem 10.2.3 (Epp) - If a graph G is connected and the degree of every vertex of G is a positive even integer, then G has Euler circuit.

Theorem 10.2.4 (Epp) - A graph G has an Euler circuit if, and only if, G is connected and every vertex of G has positive even degree.

We also define an **Euler trail** as follows:
Let G be a graph, and let v and w be two distinct vertices of G. An **Euler trail**/**path** from v to w is a sequence of adjacent edges and vertices that start at v, ends at w, passes through every vertex of G at least once, and traverses every edge of G exactly once.

A corollary follows as such: There is an Euler trail from v to w if, and only if, G is connected, v and w have odd degree, and all other vertices of G have positive even degree.

2.3 Hamiltonian circuits

Given a graph G, a **Hamiltonian circuit** for G is a simple circuit that includes every vertex of G.

That is, a Hamiltonian circuit for G is a sequence of adjacent vertices and distinct edges in which every vertex of G appears exactly once, except for the first and the last, which are the same.

Also, a **Hamiltonian graph** or **Hamilton graph** is a graph that contains a Hamiltonian circuit.

Note that Euler circuits and Hamiltonian circuits are very different, and the mathematics of the two are handled much differently. While Theorem 10.2.4 allows us to easily determine if a given graph has an Euler circuit, there is none for determining if a graph is a Hamiltonian graph, and there is no efficient algorithm for finding a Hamiltonian circuit.

If a graph G has a Hamiltonian circuit, then G has a subgraph H with the following properties:

1. H contains every vertex of G.

2. H is connected.

3. H has the same number of edges as vertices.

4. Every vertex of H has degree 2.

If G does not have a subgraph with these properties, then G is not a Hamiltonian graph.

2.4 Travelling salesman problem

Given a map with multiple cities and the respective distances between the cities, suppose a salesman must travel to each city exactly once, starting and ending in a given city. Our hypothetical salesman wishes to find the route with the minimum distance.

We can solve this problem by writing down all possible Hamiltonian circuits starting at that city and calculating the total distances accordingly.

The general travelling salesman problem involves finding a Hamiltonian circuit to minimize the total distance travelled for an arbitrary graph with n vertices in which each edge is marked with a distance. Our method of writing down all Hamiltonian circuits is impractical even for medium-sized values of n. For $n = 30$ vertices, there would be $(29!)/2 \approx 4.42 \times 10^{30}$ Hamiltonian circuits to check for!

Presently, there is no known algorithm for solving the general travelling salesman problem that is more efficient, but there are efficient algorithms that find "pretty good" solutions - that is, circuits that, while not necessarily having the least possible total distances, have smaller total distances than most other Hamiltonian circuits.

3 Adjacency matrix representations of graphs

3.1 Directed graph

Let G be a directed graph with ordered vertices $v_1 v_2, \ldots, v_n$. The **adjacency matrix** of G is the $n \times n$ matrix $A = (a_{ij})$ over the set of non-negative integers such that

$$a_{ij} = \text{the number of arrows from } v_i \text{ to } v_j \text{ for all } i, j = 1, 2, \ldots, n.$$

3.2 Undirected graph

Let G be an undirected graph with ordered vertices $v_1 v_2, \ldots, v_n$. The **adjacency matrix** of G is the $n \times n$ matrix $A = (a_{ij})$ over the set of non-negative integers such that

$$a_{ij} = \text{the number of edges connecting } v_i \text{ and } v_j \text{ for all } i, j = 1, 2, \ldots, n.$$

Note that the matrix is *symmetric*, that is, for all $i, j = 1, 2, \ldots, n$, $a_{ij} = a_{ji}$.

3.3 Connected components

For a graph G with several connected components, the resulting adjacency matrix A will consist of square matrix blocks down its diagonal and blocks of 0's everywhere else, as each connected component share no edges with vertices in other connected components.

We can generalize this reasoning to prove the following theorem:

Theorem 10.3.1 (Epp) - Let G be a graph with connected components $G_1, G_2, \ldots, G_k$. If there are n_i vertices in each connected component G_i and these vertices are numbered consecutively, then the adjacency matrix of G has the form:

$$\begin{bmatrix} A_1 & O & O & \ldots & O \\ O & A_2 & O & \ldots & O \\ O & O & A_3 & \ldots & O \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ O & O & O & \ldots & A_k \end{bmatrix}$$

where each A_i is the $n_i \times n_i$ adjacency matrix of G_i, for all $i = 1, 2, \ldots, k$, and the O's represent matrices whose entries are all 0s.

3.4   Counting walks of length N

If G is a graph with vertices $v_1, v_2, \ldots, v_m$ and A is the adjacency matrix of G, then for each positive integer n and for all integers $i, j = 1, 2, \ldots, m$, the ij-th entry of A^n = the number of walks of length n from v_i to v_j.

Essentially, if n is any positive integer, the ij-th entry of A^n equals the total number of walks of length n connecting the i-th and the j-th vertices of G.

4   Planar graphs

4.1   Isomorphisms of graphs

Two graphs that are the same except for the labeling of their vertices and edges are called **isomorphic**.

More formally, let G and G' be graphs with vertex sets $V(G)$ and $V(G')$ and edge sets $E(G)$ and $E(G')$ respectively.

G is isomorphic to G' if, and only if, there exist one-to-one correspondences $g : V(G) \to V(G')$ and $h : E(G) \to E(G')$ that preserve the edge-endpoint functions of G and G' in the sense that for all $v \in V(G)$ and $e \in E(G)$,

$$v \text{ is an endpoint of } e \iff g(v) \text{ is an endpoint of } h(e).$$

We can also show that graph isomorphism is an *equivalence relation* on a set of graphs; i.e. it is reflexive, symmetric, and transitive.

Theorem 10.4.1 - Let S be a set of graphs and let R be the relation of graph isomorphism on S. Then R is an equivalence relation on S.

4.2   Planar graph

A **planar graph** is a graph that can be drawn on a (two-dimensional) place without edges crossing.

4.2.1   Euler's Formula

When we draw a planar representation of a planar graph, it divides the plane up into *regions* or *faces*.

Euler's formula stats that for a connected planar simple graph $G = (V, E)$ with $e = |E|$ and $v = |V|$, if we let f be the number of faces, then

$$f = e - v + 2$$