MATHS 2P : Graphs & Networks

Professor Tara Brendle

Joao Almeida-Domingues[*]

University of Glasgow

September 25th, 2019 – December 4th, 2019

Contents

[*] 2334590D@student.gla.ac.uk

These lecture notes were collated by me from a mixture of sources , the two main sources being the lecture notes provided by the lecturer and the content presented in-lecture. All other referenced material (if used) can be found in the *Bibliography* and *References* sections.

The primary goal of these notes is to function as a succinct but comprehensive revision aid, hence if you came by them via a search engine , please note that they're not intended to be a reflection of the quality of the materials referenced or the content lectured.

Lastly, with regards to formatting, the pdf doc was typeset in LaTeX, using a modified version of Stefano Maggiolo's class

1 Fundamentals

Lecture 1
September 25^{th}, 2019

1.1 *Graphs*

A graph G is a pair (V, E), where V is any finite set, and E is a set whose elements are pairs of elements of V. We call the elements of V the *vertices** of G and those of E its *edges*. e.g. $G = \{\{a, b, c\}, \{ab, ac\}\}$

** often also referred as nodes*

1.1 definition. Adjacent Vertices are vertices connected directly through an edge. Formally, if $e = \{u, v\} \in E$, then u, v are adjacent

1.2 remark. Also referred to as *neighbours*

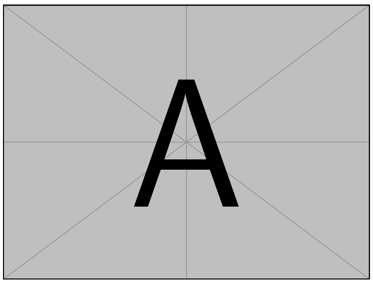
1.3 definition. Incident Edges are edges which share a vertex. We say that they are *"incident to v"*

Lecture 2
September 27^{th}, 19

Representing Graphs

Pictorially

Note that the representation need not be unique



Adjacency Matrix

Note that this definition only holds for simple graphs, i.e without loops. But , it is easily generalised if the binary requirement is dropped

1.4 definition. Adjacency Matrix is the $n \times n$ binary matrix , where $n = |V|$ and $a_{ij} = 1 \iff e = \{u, v\} \in E$; i.e iff u, v are adjacent

$$A = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

1.5 remark. In this course, we'll only deal with *simple*, *undirected* graphs. Note that AMs of this type have the nice property of being symmetric (see 2B notes for properties)

Subgraphs

1.6 definition. Subgraphs are graphs obtained by deleting edges and/or edges of another graph

1.7 definition. Induced Subgraph is a graph formed by deleting only nodes and their incident edges. Formally: Let $W \subset V$, then the induced subgraph of G is given by $G[W] = \{W, \{\{xy\}|xy \in G\}\}$. We say that *"G is induced by W"*

1.8 example.

$$G(V) = \{\{a, b, c\}, \{ab, ac\}\} \text{ and } U = \{c\} \text{ then } G[U] = \{\{a, b, \cancel{c}\}, \{ab, \cancel{ac}\}\}$$

1.9 definition. Spanning Subgraph similar to the induced, but edges are deleted instead

Lecture 3
October 2^{nd}, 19

1.2 *Graph Properties*

1.10 definition. Walk from u to v is a sequence of vertices $w1, \dots, wp$ (for some natural number $p \geq 2$), with $w_1 = u$ and $w_p = v$, such that $w_i w_{i+1}$ is an edge for every $1 \leq i \leq p - 1$

Informally, a walk is just a sequence of vertices, where each subsequent vertex added to the sequence forms an edge with the preceding one

1.11 definition. Trail a walk with distinct edges

1.12 definition. Path a walk with distinct vertices

1.13 remark. In general, every walk between two vertices contains a path

1.14 example. For a graph $P(\{a, b, c, d, e, f\}, \{ab, ac, ad, bc, bd, cd, de, ef,\})$,

Walk: $W = \{abcdeacd\}$

Trail : $T = \{abcdea\} = W \setminus \{cd\}_2$

where $\{x\}_2$ is improper notation for the repeated instances of x in a set

Path: $P = \{abcde\} = W \setminus \{acd\}_2$

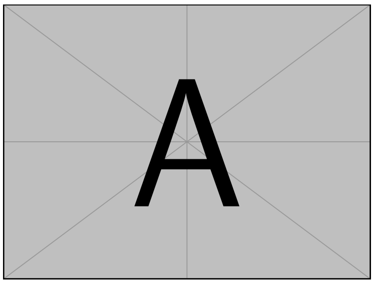
1.15 proposition. ***Number of Paths*** *For a graph with n vertices, there are $(n-1)^n$ paths (see proof below)*

1.16 definition. Connected A graph $G = (V, E)$ is connected if, for every two distinct vertices $u, v \in V$, there is a path in G from u to v

1.17 remark. A single vertex graph is connected. Since it has not distinct vertices, we say that the definition holds *vacuously*

1.18 definition. Connected Component H is a connected component of G if H is a connected induced subgraph of G and, for any subgraph H' of G such that $V(H) \subset V(H')$, H' is not connected

1.19 remark. The vertex sets of distinct connected components are necessarily disjoint



1.20 definition. Vertex Degree $d(v) = |E(v)|$, i.e. it is the size of the set of all edges connected to v

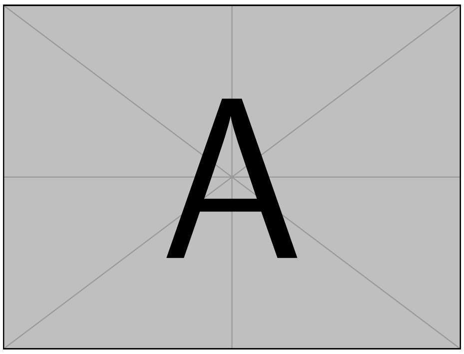
1.21 definition. Minimum Degree $\min_{v \in V(G)} d(v)$, i.e. a graph's minimum degree is equal to the lowest degree of its vertices

The converse is true of the maximum

1.3 *Isomorphisms*

1.22 definition. Isomorphism from $G_1 = (V_1, E_1)$ to $G_2 = (V_2, E_2)$ is a bijection $f : V_1 \to V_2$ such that, for every $u, v \in V_1, f(u)f(v) \in E_2 \equiv uv \in E_1$

1.23 remark. Specifically, we can consider f to be a process whereby one *relabels* the vertices



To do (4)

2 Special Graphs

2.1 definition. Complete Graphs Every pair of distinct vertices forms an edge

2.2 notation. K_n , for a graph with n vertices

2.3 proposition. ***Number of Edges*** *K_n has $\frac{1}{2}n(n-1)$ edges (see proof below)*
To do (5)

2.4 definition. Paths a path on n vertices is a graph that is isomorphic to the graph (V, E) where $V = \{v_1, \ldots, v_n\}$ and $E = \{v_i v_{i+1}\} : 1 \leq i \leq n-1$

2.5 notation. P_n

2.6 remark. P_n has $n-1$ edges

2.7 definition. **Cycle** a cycle on n vertices is a graph that is isomorphic to the graph (V, E) where $V = \{v_1, \ldots, v_n\}$ and $E = \{v_i v_{i+1}\} : 1 \leq i \leq n-1 \cup \{v_n v_1\}$. i.e, it's a path with the end vertices connected

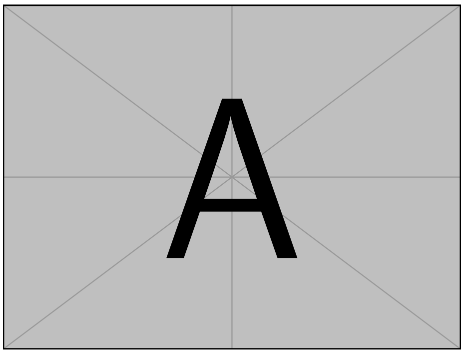
2.8 notation. C_n

3 TREES

3.1 definition. **Forest** an *acyclic* graph, i.e. without cycles

3.2 definition. **Tree** connected acyclic graph

3.3 definition. **Leaf** vertex of degree 1



To do (6)

3.1 *Basic Properties*

3.4 proposition. ***Leafs*** *Every tree as at least one leaf (see proof below)*

3.5 proposition. ***Number of Edges*** $T_n \implies |E(T)| = n-1$ *(see proof below)*

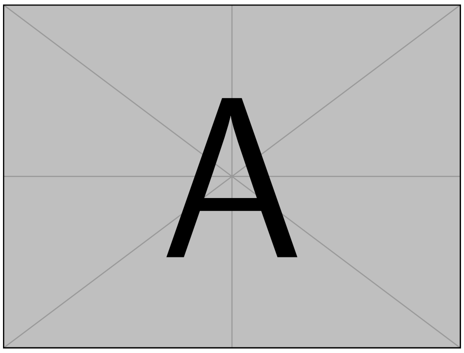
3.6 proposition. ***Connected Graph*** *Every connected graph with n vertices and* $n-1$ *edges is a Tree (see proof below)*

3.7 proposition. ***Forests*** *For a forest* F_n *with c connected components ,* $|E(T)| = n - c$ *(see proof below)*

3.8 remark. Hence, note that a tree is just a special case of a forest, where $c = 1$

3.2 *Spanning Trees*

3.9 definition. **Spanning Tree** Spanning subgraph which is not a tree
To do (7)



3.10 proposition. ***Necessity****Every connected graph contains a spanning tree* *(see proof below)*

3.11 theorem. *Cayley's Formula : A complete graph with n vertices has n^{n-2} (labelled) spanning trees*

It follows from 3.11 that , if we're interested in finding a *minimum spanning tree* (a weighted sp.tree of minimum weight), an exhaustive search through all possible trees becomes a gruelling task very quickly. There are however two *greedy* algorithms which help

3.3 *Kruskal's Algorithm*

For every edge not in the tree, add the one which has minimum weight and does not form a cycle. Stop when connected

Data: this text
Result: how to write algorithm with LaTeX2e
initialization;
while *not at end of this document* **do**
  read current;
  if *understand* **then**
    go to next section;
    current section becomes this one;
  else
    go back to the beginning of current section;
  end
end

Algorithm 1: How to write algorithms

3.12 theorem. *Kruskal's will always output a M.S.T*

Proof. QED

Reproduction not examinable, merely analysis

3.4 *Prim's*

Similar to Kruskal's but instead of looking for $min(E)$, we look for the smallest which adjacent to a node in the last iteration

4 Proof Techniques

5 Bipartite Graphs

5.1 definition. Bipartite Graph The set of vertices can be partitioned into two, and every edge has one endpoint in each partition

5.2 remark. A graph is bipartite *iff* every connected component is bipartite

To do *(9)*

Check if bipartite : (1) pick a v in V , and add it to V1 . Set V2 to empty ; (2) While V1 U V2 != V , keep picking vertices which are not in V1 or V2, but are adjacent to V1 or V2; (3) If adjacent to V1 and not V2 add to V2 and vice versa (4) if adjacent to both then not bipartite

5.3 remark. For disconnected graphs, this would be repeated for each connected component

5.4 definition. Complete Bipartite For $G(V, E)$, where $V = V_1 \cup V_2$ such that $E = \{v_1 v_2 : v_1 \in V_1, v_2 \in V2\}$. Informally, it is a complete graph which is also bipartite

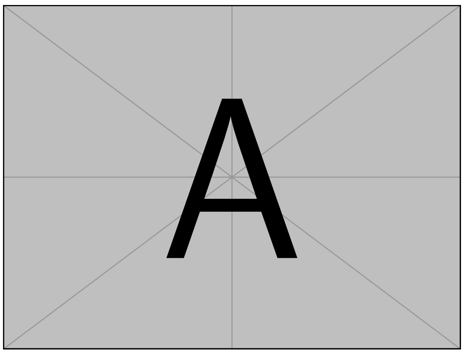
5.5 notation. $K_{p,q}$, where $p, q = |V_1|, |V_2|$

Practical modelling applications will be all cases where one is not interested , or there's simple no connection between elements within the elements of a set. For example, matching patients with a certain disease to genetic anomalies in certain genes

5.1 *Characterisation*

5.6 definition. Closed Walk Sequence of vertices $w_1, \ldots, w_p$, such that $w_i w_{i+1}$, for $p \geq 3$ and $1 \leq i \leq p-1$ and with $w_1 w_p$ as an edge

5.7 remark. It is essentially a sort of cycle which allows for repeated vertices



To do (10)

5.8 definition. Distance between two vertices u, v is the number of edges in the shortest path between u and v

5.9 notation. $d(u, v)$

5.10 theorem. *G is bipartite iff it contains no cycle of odd length as a subgraph*

expected to reproduce some part of

Proof. QED

5.2 *Matchings*

5.11 definition. Matching set of independent edges , i.e. a set of edges from which no two have a common endpoint.

5.12 definition. Perfect Matching a matching that covers all the vertices, i.e. a bijection $f_M : V_1 \to V_2$ where each $v \in V_1$ is mapped to the other endpoint of the edge in M that is incident with v

5.13 definition. Neighbourhood If X is a set of vertices in G, then the neighbourhood of X in G is the set of all vertices in G which have a neighbour in X

5.14 notation. $N_G(X)$

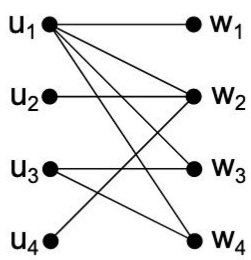
5.15 theorem. ***Hall's Marriage Theorem :*** *Let G be a bipartite graph with bipartition (U, W), where $|U| = |W| = p$. Then G contains a perfect matching iff for all $U' \subseteq U$, we have $|N_G(U')| \geq |U'|$*

5.16 remark. every subset U' of U has sufficiently many adjacent vertices in W.

Complex

Proof. See lecture notes QED

5.17 example. A common illustration of the theorem is in pairing couples. If in a group of k people there's a total of $k - 1$ options, then there can be no perfect matching. Say for example, that two people find the same and only that person acceptable, then one of them will be left unmatched. In this case the subset of people $U = \{u_1, u_2\}$, while their matching set , their neighbourhood $W = \{w_2\}$. We have then, $|U| = 2 \leq |W| = 1$



6 COLOURING

6.1 definition. Proper k-colouring $f : V(G) \to \{1, \ldots, k\}$ such that $f(u) \neq f(v)$, whenever $uv \in E(G)$, i.e we assign *"colours"* to each vertex, using only k colours, so that no two adjacent vertices have the same colour

This can be done trivially, by simply assigning a different colour to each vertex, however often colouring problems fall within the optimisation category, where we look to minimise the number of colours used

6.2 definition. Chromatic Number Smallest number k such that G is colourable

6.3 notation. $\chi(G)$

6.4 definition. Clique Number Largest t , such that K_t is an induced subgraph of G

6.5 remark. In other words, the subgraph formed by removing the least amount of vertices, so as to make G complete

6.6 notation. $\omega(G)$

6.1 *Chromatic Number and Graph Properties*

Finding the chromatic number of a graph is an np-hard problem, so in general we use certain techniques in order to find a lower bound

1. If H is a subgraph of G and $\chi(H) \geq k$, then $\chi(G) \geq k$
2. $\chi(K_n) = n$

From this, we deduce that $\chi(G) \geq \omega(G)$

6.7 definition. Independent Set For $U \subseteq V$ if no edge in E has both endpoints in U

6.8 definition. Independence Number size of the largest independence set

6.9 notation. $\alpha(G)$

6.10 remark. Note that by definition, every partition in a bipartite graph is an independent set

Note then that

6.11 theorem. *For* $|V(G)| = n$, $\chi(G) \geq \frac{n}{\alpha(G)}$

expected to reproduce some part of

To do (11)

Proof. QED

6.2 *Greedy Algorithm*

6.12 lemma. *For* $max(V(G)) = d$, $\chi(G) \leq d+1$ *Full reproduction expected*
***To do** (12)*

Proof.

To do...

- ☐ 1 (p. 4): Remove commented out cits
- ☐ 2 (p. 4): Add pictorial representation
- ☐ 3 (p. 4): proof paths
- ☐ 4 (p. 5): methods to determine isomorphism
- ☐ 5 (p. 5): proof #edges
- ☐ 6 (p. 6): Take fig.1 from lectures and connect the adjacent endpoints of all trees do contrast with forest
- ☐ 7 (p. 6): Add example of sp.sub is vs is not
- ☐ 8 (p. 7): Add Kruskal's typesetting. Change label to header
- ☐ 9 (p. 8): Write it in algo form
- ☐ 10 (p. 8): `https://slideplayer.com/slide/14901866/`
- ☐ 11 (p. 10): proof lower bound chrom&indp
- ☐ 12 (p. 11): lower bound chromatic
- ☐ 13 (p. **??**): BibTex : Diestel,Reinhard ; Graph Theory