Course Notes

Graph Theory

Alex Rutar

BSM Fall 2018

Contents

Chapter 1

Basic Structure of Graphs

1.1 A Brief Introduction

1.1.1 Basic Definitions

Def'n. 1.1.1 *A* ***graph*** $G = (V, E)$ *consists of a vertex set* V *and edge set* E *where* $E \subseteq \binom{V}{2}$.

Note that we write $\binom{V}{2}$ instead of $V \times V$ to make it clear that we cannot have loops and multiple edges.

Def'n. 1.1.2 *Two graphs F and G are* ***isomorphic*** *if there exists a bijective mapping* $f : V(F) \to V(G)$ *such that for every* $a, b \in V(F)$: $\{a, b\} \in E(F) \Leftrightarrow \{F(a), f(b)\} \in E(G)$.

Def'n. 1.1.3 *The* ***degree*** *of a vertex* $v \in V(G)$ *is the number of edges having* v *as an endpoint.*

Def'n. 1.1.4 *A* ***path*** *is asequence* $v_1 e_1 v_2 e_2 \ldots v_i e_i v_{i+1} \ldots e_j v_{j+1}$ *where each* $v_i \in V(G)$ *and* $e_i = \{v_i, v_{i+1}\} \in E(G)$ *where all* v_i*'s are different. A* ***cycle*** *is a path in which* $v_1 = v_{j+1}$.

Def'n. 1.1.5 *The* ***complementary graph*** *of* $G = (V, E)$ *is* $\overline{G} = \left(V, \binom{V}{2} \setminus E\right)$.

Def'n. 1.1.6 *A graph G is called* ***connected*** *if for every* $u, v \in V(G)$ *there exists a path between* u *and* v.

Def'n. 1.1.7 *A connected graph that becomes disconnected with the removal of any edge is called a* ***tree****. Equivalently, a* ***tree*** *is a graph which is connected and contains no cycle.*

Prop. 1.1.8 *Any tree on at least two vertices contains at least two vertices of degree 1 ("leaves").*

Proof Consider a path of maximal length. We claim that both endpoints of P have degree one. Suppose for contradiction that an endpoint has greater than one. Then the endpoint has another neighbour on the path (in which case we have a cycle), or a unique neighbour (in which case the path is not maximal), a contradiction in either case. □

Prop. 1.1.9 *A tree on* n *vertices always has* $n - 1$ *edges.*

Proof Delete the edges one by one. Each time, the number of connected components increases by one. After deleting all edges, we have n components, at the beginning, we have one, so we deleted $n-1$ edges. □

We now have an interesting question: how many different trees can be given on n labelled vertices? To investigate this, we consider the Prüfer code. Delete the smallest labelled degree one vertex and write up its unique neighbour's label. Continue doing this until only one point remains. The obtained sequences of labels is the Prüfer code.

Properties:

- The length is $n-1$
- The last digit must be n

Thm. 1.1.10 (Cayley) *The number of different trees on n labelled vertices is n^{n-2}.*

Proof We will show that sequences $x \in \{1,2,\ldots,n\}^{n-1}$ with $x_{n-1} = n$ are in bijective correspondence with the trees on n labelled vertices. First, given $a_1a_2\ldots a_{n-2}a_{n-1}$ with $a_{n-1} = n$ we want to decode it. Let $b_1, b_2, \ldots, b_{n-1}$ be the sequence of labels of vertices deleted in the order of the indices. If we "decode" $b_1b_2\ldots b_{n-1}$, we know the tree, since we have the $n-1$ edges $\{a_i, b_i\}$.

1. $b_1 := \min\{k \in \{1,2,\ldots,n\} : k \notin \{a_1,\ldots,a_{n-1}\}\}$
2. $b_2 := \min\{k \in \{1,2,\ldots,n\} : k \notin \{b_1, a_2,\ldots,a_{n-1}\}\}$

(∗) $\boxed{b_i := \min\{k \in \{1,2,\ldots,n\} : k \notin \{b_1,\ldots,b_{i-1},a_i,\ldots,a_{n-1}\}\}}$

We show that taking any sequence $a_1a_2\ldots a_{n-1}$ with $a_{n-1} = n$ and applying (∗) to obtain $b_1,\ldots,b_{n-1}$, the graph we obtain on vertices $1,\ldots,n$ with the $n-1$ edges $\{a_i, b_i\}$ (1) is a tree, and (2) has Prüfer code is just $a_1a_2\ldots a_{n-1}$.

Note that (∗) implies that $\{b_1, b_2, \ldots, b_{n-1}, a_{n-1}\} = \{1,2,\ldots,n\}$. Define graphs T_i for $i = n-1, n-2, \ldots, 2, 1$ on the graph spanned by the edges $\{a_{n-1}, b_{n-1}\}, \{a_{n-2}, b_{n-2}\}, \ldots, \{a_i, b_i\}$. It suffices to prove that T_i is a tree for every i and b_i is its smallest labelled degree 1 vertex.

We do this by induction. Clearly, it is true for $i = n-1$. Once it is true for $i = n-1,\ldots,j+1$, we prove this for $i = j$. We know that T_{j+1} is a tree, and we wish to add the edge $\{b_j, a_j\}$. Thus $b_j \notin V(T_{j+1}) = \{b_{j+1}, b_{j+2}, \ldots, b_{n-1}, a_{n-1}\}$ so b_j is indeed degree one; $a_j \in V(T_{j+1})$ and T_j is a tree. If b_j was not the smallest degree 1 vertex, then there exists some $k > j$ such that $b_k < b_j$ and b_k has degree one in T_j. But then $b_k \notin \{b_1, \ldots, b_{j-1}, a_j, \ldots, a_{n-1}\}$ so (∗) would have chosen it in place of b_j, a contradiction. □

1.2 Paths, Circuits, and Cycles

1.2.1 Eulerian Circuits

Königsberg (modern Kaliningrad)

Def'n. 1.2.1 *An* ***Eulerian circuit*** *is a closed walk in a graph that contains every edge exactly once. An* ***Eulerian path*** *is a walk containing every edge exactly once and not (necessarily) ending at the same vertex.*

Note that we do allow multiple edges between vertices (graph is necessary simple). We also assume that our graph is connected.

Thm. 1.2.2 (Euler) *A graph contains an Eulerian circuit if and only if every vertex has even degree.*

Cor. 1.2.3 *A graph contains an Eulerian path if and only if all but two vertices have even degree.*

In both cases, necessity is obvious: every time a path arrives at a vertex, it adds two to the degree since there is a unique edge in and out from the vertex. Thus for the corollary the only odd vertices can be the endpoints, and for the theorem, there are no endpoints in the path.

Proof (cor.) To see the corollary, first add an edge connecting the two odd degree vertices. Then by the theorem, we have an Eulerian circit walk through it so that the added edge is the last one traversed. Delete it, and the remaining part of the walk gives our Eulerian path □

Proof (thm.) We can now prove the theorem. Consider a maximal walk P on G that does not repeat edges. Because of the evenness of all degrees, it must be closed. If every edge is contained, it is an Eulerian circuit and we are done. If there exists some $e \in E(G)$ that is not in the walk, then by connectedness, there must be a path from a vertex in the walk to an endpoint of e. Considering a shortest such path (perhaps containing 0 edges), all edges on it are unused so far. Now take the starting point of this path on our walk, go through the closed path on our walk, go through the closed walk we have starting here and thus contine on the said path and include e, a contradiction. □

1.2.2 Hamiltonian Cycles

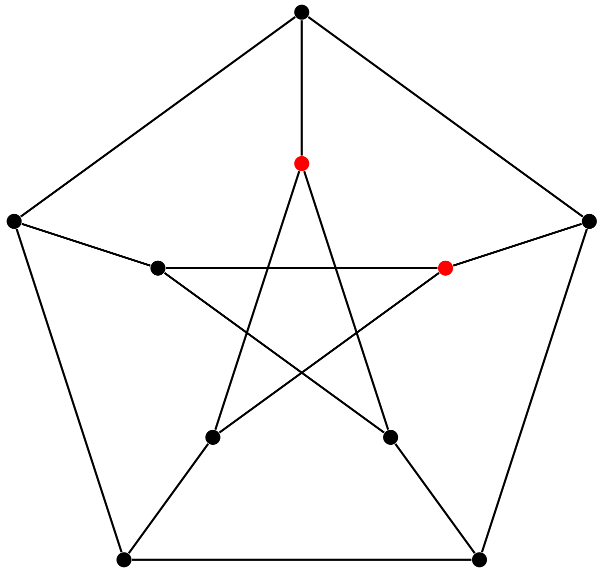
Def'n. 1.2.4 *A **Hamiltonian cycle** in a graph is a cycle containing every vertex exactly once. A **Hamiltonian path** is a path that contains every vertex exactly once.*

1.2.3 Necessary Conditions

Prop. 1.2.5 *If G contains a Hamiltonian cycle, then after deleting any k of its vertices, the remaining graph cannot have more than k components. Similarly, if G contains a Hamiltonian path, then deleting any k of it vertices yields a graph with at most $k+1$ components.*

Proof This can be easily proven by induction. □

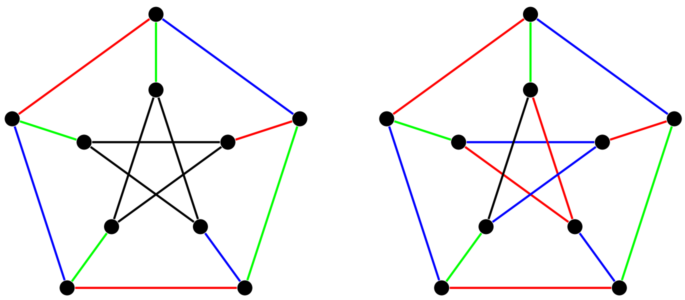
Here is a graph for which this condition holds, but does not have a Hamiltonian cycle.



The condition holds: delete $k = l_1 + l_2$ vertices, where l_1 is in the "outer cycle" and l_2 is in the "inner cycle". (Above, $l_2 = 2$ and $l_1 = 0$.) Then the outer cycle may not fall apart into more than l_1 components, the inner cycle may not fall apart into more than l_2 component, so if $l_1, l_2 > 0$. If l_1 or l_2 is 0, then the graph remains connected.

Furthermore, there does not exist a Hamiltonian cycle. Suppose such a cycle exists, then it is a cycle containing 10 edges. Colour the alternating edges red and blue. Thus every vertex is adjacent to a red edge and a blue edge. Colour the remaining 5 edges white, so every vertex will be the end vertex of exactly 1 white edge. However, such an edge-coloring is impossible.

Up to isomorphism, the outer edges must be coloured with two of colour 1, two of colour 2, and one of colour 3. We then fill in the interior edges as required, until the inner cycle, in which we are forced to draw the following edges, yielding our contradiction.



1.2.4 Sufficient Conditions

Thm. 1.2.6 (Dirac, 1952) *If G is a graph on n vertices, with every degree being at least $\frac{n}{2}$, then G contains a Hamiltonian cycle.*

As a fun fact, this is not Paul Dirac, but rather Gabriel Andrew Dirac (Paul Dirac's stepson). Note that we must have $f(n) \geq n/2$. If not, then the graph composed of two disconnected components $K_{n/2}$ has degree $n/2 - 1$ everywhere but does not have a hamiltonian cycle.

Thm. 1.2.7 (Ora, 1960) *If G is a graph on n vertices such that for every nonadjacent pair of vertices $u, v \in V(G)$, $d(u) + d(v) \geq n$ is satisfied, then H contains a Hamiltonian cycle.*

Ora's Theorem is stronger since every graph which satisfies Dirac's condition certainly satisfies Ora's condition.