MAT344 Lecture 10

2019/June/6

1 Announcements

2 This week

This week, we are talking about

1. Graphs

3 Recap

Last time we talked about

1. Graph isomorphism

4 The Handshaking Lemma

If there are many people shaking hands, the total number of hands shaken is twice the number of all handshakes.

Theorem 4.1 (Handshaking Lemma, Theorem 5.1 in [KT17])**.** *Let* $\deg_G(v)$ *denote the degree of vertex* v *in a graph* $G = (V, E)$*. Then*

$$\sum_{v \in V} \deg_G(v) = 2|E|.$$

Proof. We will give a combinatorial proof. For the left hand at every vertex we count the number of edges incident to that vertex. For the right hand side, notice that this way we counted each edge twice, as every edge is incident to two vertices.

Q.E.D.

Theorem 4.1 seems obvious, but it has at least one useful corollary.

Corollary 4.2 (Corollary 5.2 in [KT17])**.** *Every graph has an even number of vertices of odd degree.*

Proof. The right hand side of the equation in Theorem 4.1 is even, so on the left hand side, we must have an even number of odd numbers.

Q.E.D.

5 Forests and Trees

Definition 5.1. *A graph that contains no cycles is called a* ***forest****. We call a graph* ***connected*** *when there is a path from* x *to* y *in* G *for every pair* x, y *of vertices. A connected forest is called a* ***tree****.*

Trees have some really important applications, as they are "minimal connected graphs" in some sense.

Definition 5.2. *If* $G = (V, E)$ *is a graph, then* $T = (V, F)$ *with* $F \subseteq E$ *is a* ***spanning tree*** *of* G *if* T *is a tree (note that the vertex set of both graphs is the same).*

Later in the class we will study algorithms to find spanning trees.

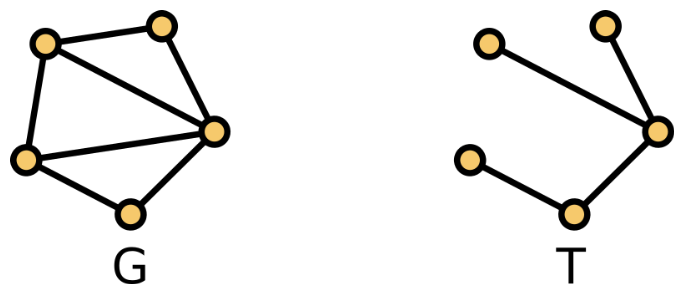


Figure 1: A graph and a spanning tree

6 Eulerian Graphs (Chapter 5.3 in [KT17])

In this section, by "graph", we will mean a multigraph.

Definition 6.1. *A walk on a graph is called an* ***Euler walk*** *if it traverses every edge exactly once. An Euler walk is called an* ***Euler circuit*** *or* ***Euler cycle*** *if the walk finishes at the same vertex where it started.*

In the previous lecture we have seen that a graph can not have an Euler walk if more than two of the edges have odd degrees. A similar argument shows that a graph can only have an Euler circuit if all the vertices have even degrees. It is also clear that in order to have either an Euler walk or Euler circuit, the graph needs to be connected.

Theorem 6.2 (Theorem 5.4. in [KT17])**.** *A graph G has an Euler circuit if and only if it is connected and every vertex has even degree.*

The proof is a bit subtle, and we have to make sure we are doing a rigorous job, but the idea is simple. Since all vertices have an even degree, if we start a walk and traverse edges in some order, we can never get "stuck", since whenever we enter a vertex (hence using up one of the edges incident to it) we can always "leave" (using another edge). So the only place where we can end up without more edges to traverse is the starting vertex. Our path may not be long enough, but the key idea here is that the if we remove the edges we traversed from the graph, the remaining graph still satisfy the property that all degrees are even, so we can repeat the procedure.

Example 6.3. *Before seeing the proof, let's see an example of this idea in action. Consider the graph in figure 2. At first we take an arbitrary path, starting at vertex 1, like in figure 3.*

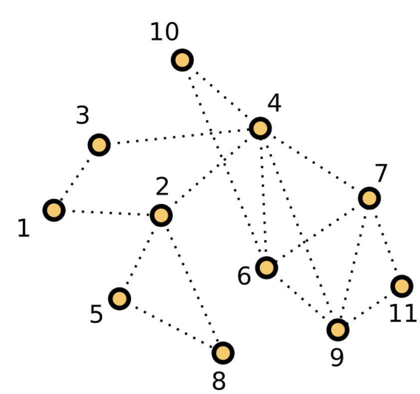
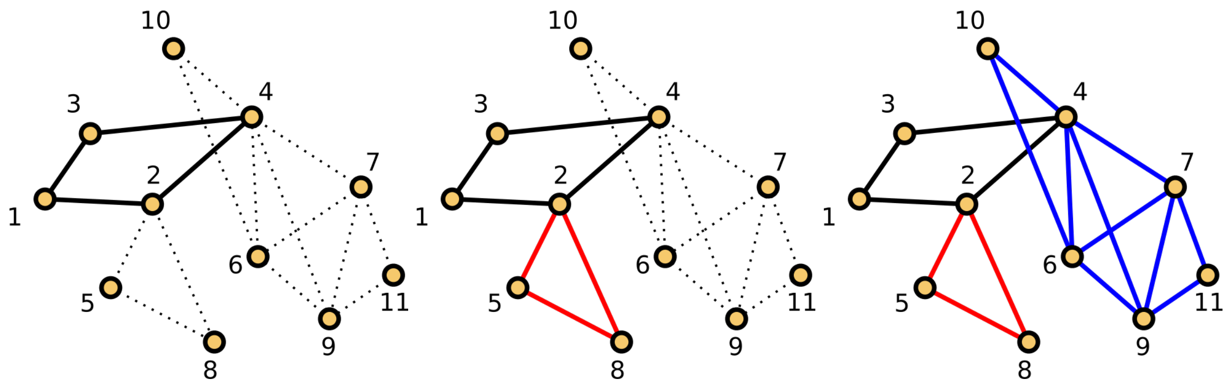


Figure 2: An Eulerian graph

Proof of Theorem 6.2. We already know that the conditions on degrees and connectedness are necessary, so we proceed to prove the converse. Assume that $G = (V, E)$ is a connected graph with every vertex having even degree. We will proceed by induction on the number of edges $n = |E|$. As a base case, we have the graph with one vertex and no edges, this graph has an Euler circuit (the empty walk). Assume for induction that all connected graphs on at most $n - 1$ vertices with vertices having even degrees have an Euler circuit. Consider the case where $|E| = n$. Start at any vertex v_0 and traverse edges in any order. Since every vertex has even degree, we can always leave any



(a) Tracing the first cycle (b) Adding the second cycle (c) Finishing with the third cycle

Figure 3: Finding an Euler circuit

vertex we entered, so at some point our walk will get us back to v_0. If we traversed all edges, we have found an Euler circuit and we are done. Otherwise, let $W = (v_0, v_1, \ldots v_k = v_0)$ be the walk G' be the graph G with the edges from the walk W removed. The graph G' may be disconnected, let its connected components be $G'_1, G'_2, \ldots, G'_m$. Because G is connected, at least one vertex in each of the G'_is appears in W. Let $w_{i,1}$ be a vertex that appears in W and is contained in G'_i. Any component G'_i still only has vertices with even degrees, as for any vertex of G we have removed an even number of edges. Therefore, by the induction hypothesis, G_i has an Euler circuit. Note that if a graph has an Euler circuit starting at some vertex, it will have an Euler circuit starting at any vertex (we can cyclically shift the circuit). Therefore we may assume that G'_i has an Euler circuit of the form

$$(w_{i,1}, w_{i,2}, \ldots, w_{i,k_i} = w_{i,k_i}.$$

Now we will patch these Euler circuits together with W. After possibly reordering the components, we may assume that W contains the starting vertices of the G'_is in the order $w_{1,1}, w_{2,1}, w_{3,1}, \ldots, w_{m,1}$. Define a new walk as follows:

$$\begin{aligned}(v_0, v_1, \ldots v_{i_1} &= w_{1,1}, w_{1,2}, \ldots, w_{1,k_1} = v_{i_1}, v_{i_1+1}, \ldots\\ \ldots, v_{i_2} &= w_{2,1}, w_{2,2}, \ldots, w_{2,k_2} = v_{i_2}, v_{i_2+1}, \ldots\\ \ldots, v_{i_m} &= w_{m,1}, w_{m,2}, \ldots, w_{m,k_m} = v_{i_m}, v_{i_m+1}, \ldots, v_k = v_0.\end{aligned}$$

Our walk now uses every edge in the graph exactly once.

Q.E.D.

Exercise 6.4. *Under what conditions does a graph have an Euler walk (not necessarily a circuit)?*

7 Hamiltonian Graphs (Chapter 5.3 in [KT17])

Definition 7.1. *A **Hamilton path** is a path in the graph that traverses every vertex exactly once. If $v_0, \ldots, v_k$ is a Hamilton path and $v_k v_0$ is also an edge in the graph, then we call this Hamilton path a **Hamilton cycle**. A graph that has a Hamilton cycle is said to be **Hamiltonian**.*

Exercise 7.2. *Explain why the graph in Figure 4 has no Hamilton path.*

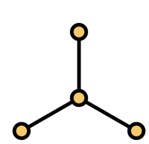


Figure 4: A graph with no Hamilton path

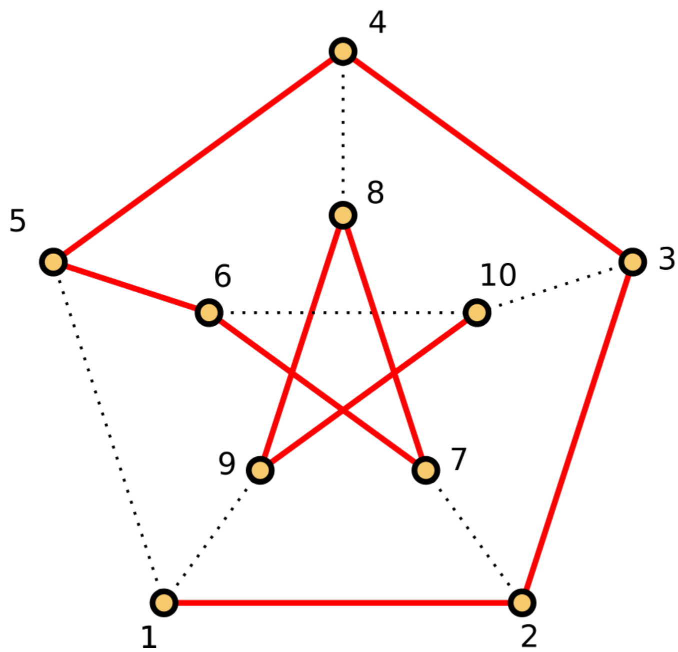


Figure 5: The Petersen graph

Figure 5 shows the **Petersen graph**, a graph that provides many counterexamples, and a Hamilton path in it.

Exercise 7.3. *Prove that the Petersen graph does not have a Hamilton cycle. (this is not easy!)*

Theorem 6.2 tells us that it is very easy to tell when a graph has an Euler circuit (and an Euler walk). For Hamilton paths and cycles, there is no known easy way of answering the question in general. Finding a Hamilton path on an n-vertex graph is also very difficult, the brute-force check takes $O(n!)$ time if the basic operation is "given a sequence of vertices, check if it is a path in the graph", but the best known algorithm still takes $O(n^2 2^n)$ time.

However, in some cases we can conclude that the graph has a Hamilton cycle. For example, any complete graph K_n has a Hamilton cycle. Notice that having more edges can never hinder the existence of a Hamilton cycle (unlike in the case of Euler cycles). So most of the Theorems that guarantee the existence of Hamilton cycles are about the graph having "sufficiently many edges".

Theorem 7.4 (Dirac, 1952, Theorem 5.5. in [KT17])**.** *If a simple graph G has n vertices with $n \geq 3$ and each vertex v has $\deg_G(v) \geq \lceil \frac{n}{2} \rceil$ then G is Hamiltonian.*

Proof. Suppose the Theorem fails, and let n be the smallest positive integer for which there is a graph with each vertex having degree at least $\lceil \frac{n}{2} \rceil$ and there is no Hamiltonian cycle in G. Since the only possible graph on 3 vertices with $\deg_G(v) \geq 2$ is the complete graph, and it has a Hamilton cycle, we may assume $n \geq 4$.

Let t be the largest integer for which G has a path $P = (x_1, \dots, x_t)$ on t vertices. Since the path begins with x_1 and ends with x_t, if any neighbor of x_1 or x_t is not already in P, we could attach them to the beginning or the end, resulting in a longer path. So we may assume that all neighbors of x_1 and all neighbors of x_t are all in P already. In particular, this shows that $\lceil \frac{n}{2} \rceil < t$

Set up $t - 1 \leq n - 1$ boxes and put the (at least $\lceil \frac{n}{2} \rceil$ many) edges of the form $x_1 x_{i+1}$ into box i, and put the (similarly at least $\lceil \frac{n}{2} \rceil$ many) edges of the form $x_i x_t$ into box i. We have at least n edges to put in at most $n - 1$ boxes, so there is an index i with $1 \leq i < t$ such that both $x_1 x_{i+1}$ and $x_i x_t$ are edges in G. Then we can reverse the end of the path and form

$$C = (x_1, x_2, \dots, x_i, x_t, x_{t-1}, \dots, x_{i+2}, x_{i+1}),$$

which is now a cycle of length t. Since G has no Hamilton cycle, we must have $t < n$, and combining it with our other estimate for t, we get $\lceil \frac{n}{2} \rceil < t < n$. If y is a vertex not contained in C, there must be an x_j adjacent to y, in which case we can form a path in G of length $t + 1$ by starting a path at y, then tracing the cycle from x_j. This contradicts our assumption that t was maximal.

Q.E.D.

A different proof of Theorem 7.4 leads to some other sufficient conditions for the existence of Hamilton cycles. For the proof, see [Mor17], Theorem 13.13 or [Gui18], Theorem 5.3.2.

8 Graph coloring (Chapter 5.4 in [KT17])

Definition 8.1. *If G is a graph and C is a set (the elements of C are often called* colors*), a* ***proper coloring*** *of G is a function $f : V(G) \to C$ such that if $xy \in E(G)$, we have $f(x) \neq f(y)$. The smallest integer t for which there is a proper coloring of G with $|C| = t$ is called the* ***chromatic number*** *$\chi(G)$ of G. In this case, we say that G is t-colorable.*

Figure 6 shows a 5-coloring of a graph

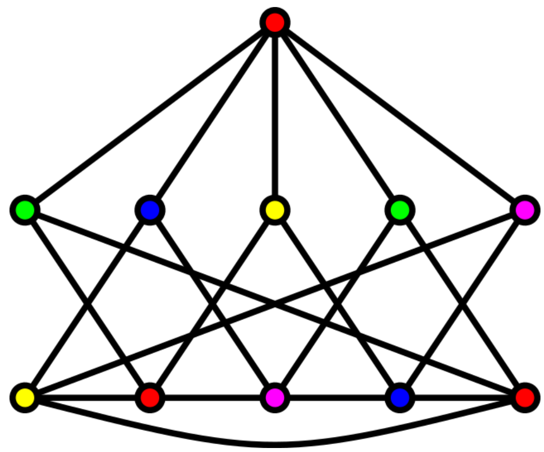


Figure 6: A 5-coloring of a graph

But is this the minimal number of colors we need?

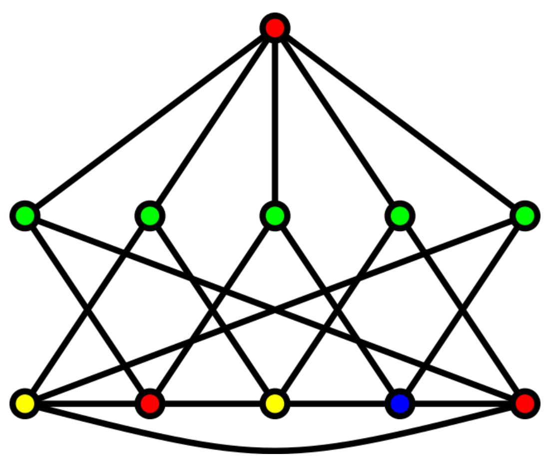


Figure 7: A 4-coloring of the same graph

But is *this* the minimal number of colors? It's not so easy to decide.

Deciding when a graph has a k-coloring is another example of a computationally difficult problem. Like many other difficult problems, in certain special cases it's quite easy.

Theorem 8.2. *A graph is 1-colorable if it has no edges.*

Example 8.3. *Prove that a complete graph K_n has $\chi(K_n) = n$.*

Theorem 8.4 (Theorem 5.7 in [KT17]). *A graph is 2-colorable if and only if it does not contain an odd-length cycle.*

Proof. Clearly the condition is necessary. It suffices to consider connected graphs. Pick a vertex x and define a map $f : E(G) \to [2]$ by the rule

$$f(y) = \begin{cases} 1 \text{ if the shortest path from } x \text{ to } y \text{ is odd} \\ 2 \text{ if the shortest path from } x \text{ to } y \text{ is even} \end{cases}$$

We claim this is a proper coloring. If there are adjacent vertices y and z both colored i (for $i = 1$ or 2), then consider shortest paths $(x = y_0, y_1, \dots, y_{k-1}, y_k = y)$ from x to y and $(x = z_0, z_1, \dots, z_l = z)$ from x to z. Note that if $y_i = z_j$ for some i, j then i must equal j as the paths must be minimal to the vertex $y_i = z_j$. Then $(y_i, y_{i+1}, \dots y_k = y, z = z_l, z_{l-1}, \dots, z_{i+1}, z_i)$ is an odd length cycle.

Q.E.D.

A 2-colorable graph is called a **bipartite** graph. This refers to the idea that we can partition the vertices into 2 subsets A and B such that there are no edges between vertices in A, and no edges between vertices in B. Equivalently, the induced subgraphs of G by A and B are independent, and $A \cup B = V(G)$. This also easily generalizes to n-colorable graphs.

9 Cliques and Chromatic number (Chapter 5.4.2 in [KT17])

Definition 9.1. *A **clique** in a graph G is a set $K \subseteq V(G)$ such that the subgraph induced by K is the complete graph $K_{|K|}$. The **maximum clique size** or **clique number** of a graph G, denoted $\omega(G)$ is the largest t for which there exists a clique K with $|K| = t$.*

Considering example 8.3, we see that

$$\chi(G) \geq \omega(G).$$

But this estimate is not very effective at computing the chromatic number:

Proposition 9.2 (Proposition 5.9 in [KT17]). *For every $t \geq 3$, there exists a graph G_t such that $\chi(G_t) = t$ and $\omega(G_t) = 2$.*

Moreover, finding the maximum clique size is also a computationally difficult problem. Both computing $\omega(G)$ and $\chi(G)$ are difficult problems but they both have *easy to verify certificates*. That is, given a proposed solution (a coloring of the vertices, or a proposed set of vertices that is supposed to be a clique) is easily verified to be a solution. In both cases, we just need to check that certain vertices are or are not adjacent (which is at most $\binom{n}{2}$ checks in the worst case). So these problems are in $\mathcal{NP}$ (see [KT17] Chapter 4.4.3 for a discussion of the class $\mathcal{NP}$).

It turns out that both the maximum clique problem and the graph coloring problems are *as difficult as any problem that is in $\mathcal{NP}$* (in particular, they are probably more difficult than the graph isomorphism problem, but this depends on $\mathcal{P} = \mathcal{NP}$). Problems in this complexity class are known as $\mathcal{NP}$-complete.

References

[Gui18] David Guichard. *Combinatorics and Graph Theory*. Open access, 2018. Available at https://www.whitman.edu/mathematics/cgt_online/book/. 5

[KT17] Mitchel T. Keller and William T. Trotter. *Applied Combinatorics*. Open access, 2017. Available at http://www.rellek.net/appcomb/. 1, 2, 3, 4, 5, 6

[Mor17] Joy Morris. *Combinatorics*. Open access, 2017. Available at http://www.cs.uleth.ca/~morris/Combinatorics/Combinatorics.html. 5