Graph Theory Exam 2 Study Guide

Ethan Beaird

March 12, 2023

0.1 Euler Tours

A walk is a *trail* if it crosses every edge at most once. Recall that an Euler trail is a trail that crosses every edge (necessarily once).

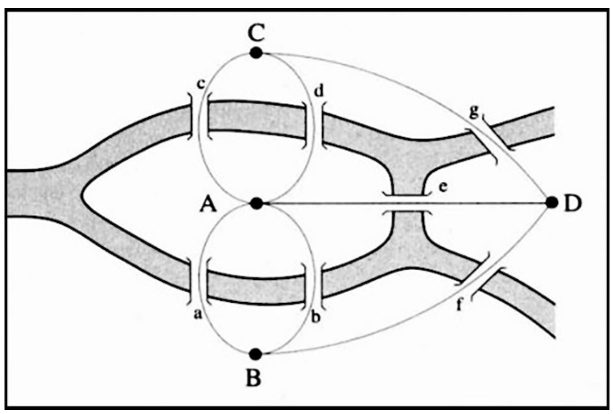

Figure 1: 7 Bridges of Konigsberg Problem

An *Euler tour* is a closed Euler trail. A graph G is *Eulerian* if it contains an Euler tour.

Theorem 4.1: A Graph G is Eulerian iff it is connected and every vertex has even degree.

Lemma: If H is a nontrivial connected graph where each vertex has even degree, then for all vertices in H, H has a closed trail (of positive length) starting and ending at v.

Corollary 4.1: A graph contains an Euler Trail iff it has 0 or 2 vertices of odd degree in G.

0.2 Hamilton Cycles

A *Hamilton Path* is a path which contains all vertices of G.
A *Hamilton Cycle* is a cycle which contains all vertices of G.

Unfortunately, checking whether or not a graph is Hamiltonian is Hell for computers to check.

We will see some necessary conditions for being Hamiltonian (i.e. it holds if it is Hamiltonian) and sufficient conditions (i.e. if it holds then G is Hamiltonian).

Theorem 4.2: If G is Hamiltonian, then for each nonempty proper subset $S \subset V(G)$, $\omega(G - S) \leq |S|$.

Theorem 4.3: If G is simple with $\nu \geq 3$ and $\delta \geq \frac{\nu}{2}$, then G is Hamiltonian.

Lemma: Suppose G is simple and u and v are non-adjacent vertices such that $d(u) + d(v) \geq \nu(G)$. Then G is Hamiltonian iff G + uv is Hamiltonian.

0.3 Closure

The *closure* of G is the graph obtained from G by recursively joining pairs of nonadjacent vertices u,v, satisfying $d(u) + d(v) \geq \nu(G)$ until all are gone. The graph created we call $C(G)$.

Lemma 4.4.2: $C(G)$ is well defined- no matter what choices you make, you get the same result.

0.3.1 Chinese Postman Problem

Edges = Streets (edges have positive weight). Vertices = intersections. Find a shortest closed walk that crosses all edges If G is Eulerian (every degree is even) then ever tour is optimal.

0.3.2 Traveling Salesman Problem

Given graph G with positive edge weights, find a Hamilton cycle with the least weight.

Easier Problem: Find a minimum weight Hamilton cycle in a weighted complete graph. There exists an algorithm which approximates an optimal solution.

A simplified version of an algorithm: Start with some Hamilton cycle in a complete graph

$$C = v_1 v_2 ... v_n v_1$$

Look at all i, j such that $1 < i+1 < j < \nu$ and compare $w(v_i v_{i+1}) + w(v_j v_{j+1})$ to $w(v_i v_j) + w(v_{i+1} v_{j+1})$. If the latter is smaller for some i, j, then do the following: C becomes $C_2 = v_i ... v_j v_{j-1} ... v_{i+1} v_{j+1} v_\nu v_i$ and the new weight of C_2 is $w(C) + w(v_i v_j) + w(v_{i+1} v_{j+1}) - w(v_i v_{i+1}) - w(v_j v_{j+1})$, which is strictly less than $w(C)$. Repeat this process until you can no longer do so.

0.4 Matchings

Definition: A subset $M \subset E(G)$ is a matching if $E(G)$ has no loops and no 2 edges in M are adjacent. *(When we consider matchings, assume G is always simple!)*

A matching M *saturates* v or v is *M-saturated* if v is an end of some edge in M.

M is a *perfect matching* if every vertex is M-saturated. M is a *maximum matching* if its a matching of biggest size.

An $M-$ *alternating path* is a path where edges alternate between M and $E - M$. An M-alternating path is an $M-$ *augmenting path* if the starting and ending vertices are M-unsaturated.

Theorem 5.1: A matching M in G is a maximum matching iff G contains no M-augmenting path.

Definition: For $S \subset V(G)$, the neighbor set of S, $N_G(S)$, is the set of vertices adjacent to at least 1 element of S.

Theorem 5.2: Let G be a bipartite graph. Then G contains a matching that saturates X iff $|N_G(S)| \geq |S|$.

Corollary 5.2: If G is k-regular and bipartite with $k > 0$, then G has a perfect matching.

0.5 Coverings

Definition: A *cover* of a graph G is a set $K \subset V(G)$ such that every edge is incident to some vertex in K.

Observation: By the nature of a cover K and a matching M, $|M| \leq |K|$. Why?

Every $e \in M$ is incident to at least one vertex $v \in K$, but no other $e' \in M$ is incident to this same vertex. For maximum matching M^* and minimum cover K', $M^* \leq K'$.

Theorem 5.3: In a bipartite graph, $|M^*| = |K'|$.

Lemma 5.3: If M is a matching, K is a cover, and $|M| = |K|$, then M is a maximum matching and K is a minimum cover.

Definition: A component of a graph is called *odd* if it has an odd number of vertices. Otherwise it is called *even*.

Theorem 5.4 - Tutte's Theorem: Let $o(G)$ be the number of odd components in G. G has a perfect matching iff $o(G - S) \leq |S|\ \forall S \subset V(G)$.

Corollary 5.3: Every 3-regular graph without a cut vertex has a perfect matching.

Algorithm to find a perfect matching in a bipartite graph:
With n employees and n tasks, each employee can do some subset of tasks. The *Hungarian Algorithm* presents the following idea: Given some non-perfect matching M, there is some M-unsaturated $u \in X$, and we construct an M-alternating tree rooted at u to find an M-augmenting path or $|S| \subset X > |N(S)|$.

Definition: For a bipartite G, $H \subset G$ is an M-alternating tree rooted at u if its a tree and every (u-v)-path is M-alternating.

Observe that in an M-alternating tree rooted at u, every path from u to $x \in X$ has last edge in M.

M-alternating tree subroutine: Given M and $u \in X$ unsaturated:

1. Start with H = $\{u\}$.
2. Look for $y \in Y - V(H)$ adjacent to some $x_0 \in V(H) \cap X$ via an edge e. If no such y exists, then $S = X \cap V(H)$ has the property that $|S| > |N(S)|$, so no perfect matching. Else, add y and e to H.
3. If y is M-unsaturated, then $u - y$ path in H is augmenting. Output the path.
4. If y is M-saturated and is paired by some $e' \in M$ with some $x \in X - V(H)$. Add x, e' to H ans return to step 2.

Things to Check:

- H is always M-alternating tree rooted at u. We just want if H starts as such then it remains so as we do steps 2-4. In step 2, we add an edge from $x_0 \in V(H) \cap X$ to some $y \in Y - V(H)$ and add y to H.

- In the original H, the $(u-x_0)$-path has last edge in M and previous vertex was in H, so the new y is not paired to x by an M-edge. So e is not in M, and our $(u-y)$-path is alternating. So H has $\epsilon = \nu - 1$ and we increment ϵ and ν by one each time, so it holds, H is still a tree.
- Notice that before step 2, for every x in V(H), x is M-saturated and the edge it is incident to in M is an edge in H. As we go through step 4, the new y vertex is incident to a new x vertex via an M-edge not in H. So still a tree, and since the (u-y) path ended in a non-M-edge, the (u-x) path will be M-alternating.
- H starts with 1 x-vertex. After running through 2-4, we add 1 x and 1 y vertex to H. When we get to the beginning of step 2, the number of X vertices in H = the number of Y + 1. If we can't find an $y \in Y - V(H)$ adjacent to a vertex to a vertex in $X \cap V(H)$, then the neighbors of $X \cap V(H)$ is already in $H \subset V(H) \cap Y$.
- We found an $S \subset X$ with $N(S) \subset V(H) \cap Y$ and $|N(S)| \leq V(H) - Y| < |S|$.

Full algorithm:

- A. Start with a matching.
- B. If M is perfect, stop.
- C. Else, run subroutine for some M-unsaturated vertex u in X.
- D. If subroutine stops at step 2, no perfect matching. STOP.
- E. Else the subroutine outputs an M-augmenting path P. Replace M with $M \triangle E(P)$ which is a larger matching.
- F. Go to (B).

0.6 Edge Colorings

Definition: A $k-$*edge coloring* of a loopless graph G is an assignment of "colors" $1, 2, 3, ..., k$ to the edges of G. I.e. label each edge with $\{1, 2, 3, ..., k\}$. The coloring is *proper* if no two adjacent edges have the same color.

Definition: G is $k-$*edge colorable* if G has a proper k-coloring. The *edge chromatic number* $\chi'(G)$ is the smallest k for which G is k-edge-colorable. Also, we could say a coloring C as a part. of $E(G)$ by $E_i = \{e \in E(G) | e \text{ color is i}\}$.

$$E(G) = E_1 \cup E_2 \cup ... \cup E_k$$

We denote C as $\{E_1, E_2, ..., E_k\}$. Observe that for C a proper coloring, $\forall e_1, e_2 \in E_i$ with $e_1 \neq e_2 \Rightarrow e_1, e_2$ are not adjacent. So E_i is a matching. Recall that

$\Delta(G)$ is the maximum degree of any vertex in G.

Observe that $\chi'(G) \geq \Delta(G)$ as there exist $|\Delta(G)|$-many edges incident to the same vertex.

Theorem 6.1: If G is bipartite, $\chi'(G) = \Delta(G)$.

Theorem - Vizing's Theorem: If G is simple, then its edge-chromatic-number is either $\Delta(G)$ or $\Delta(G) + 1$.

Time Tabling Problem: m teachers need to teach n classes P_{ij} times a week. Our goal is to teach all classes or sections in as few of periods as possible. Each period is a matching, so a time table is a coloring of a bipartite graph. One more consideration is the number of rooms available. If ϵ is the total number of classes, can you find a time table where in each period, at most the ceiling of $\frac{\epsilon}{p}$ classes are being taught? Yes!

Lemma 6.3: If M,N are disjoint matchings of a graph G with $|M| > |N|$, then there are disjoint matchings M', N' such that $|M'| = |M| - 1$, $|N'| = |N| + 1$ and $M \cup N = M' \cup N'$.

Theorem 6.3: Let G be bipartite and $p \leq \Delta(G)$. Then there exist p disjoint matchings $M_1, M_2, ..., M_p$ such that $E(G)$ =Union of the $M_i s$ and
$\lfloor \frac{\epsilon}{p} \rfloor \leq M_i \leq \lceil \frac{\epsilon}{p} \rceil$.

Independent Sets and Cliques

Recall that $S \subset V(G)$ is an *independent set* if no 2 vertices in S are adjacent in G. It is maximum if it is of maximum size.

Theorem 7.1: $S \subset V(G)$ is an independent set iff $V(G) - S$ is a covering.

$S \subset V(G)$ independent set means that all v are non-adjacent in S. $S \subset V(G)$ clique means that all v adjacent in S.

Let $\alpha(G)$ be the *independence number*, the maximum size of an independent set.
Let $\beta(G)$ be the *vertex covering number*, the maximum size of a vertex covering.

Corollary 7.1: $\alpha(G) + \beta(G) = \nu(G)$
$L \subset E(G)$ is an *edge covering* if every $v \in V(G)$ is incident to some edge in L. An edge covering exists iff $\delta(G) > 0$.

$\alpha'(G)$ is the *edge independence number*, the maximum size of a matching in G.
$\beta(G)$ is the *edge covering number*, the minimum size of an edge covering.

M matching does not imply M^C a covering.
L covering does not imply L^C a matching.

Theorem 7.2: If $\delta(G) > 0$, then $\alpha' + \beta' = \nu$.