CS 70 Spring 2022 | Discrete Mathematics and Probability Theory — Koushik Sen and Satish Rao | HW 02

Due: Saturday, 2/5, 4:00 PM
Grace period until Saturday, 2/5, 6:00 PM

Sundry

Before you start writing your final homework submission, state briefly how you worked on it. Who else did you work with? List names and email addresses. (In case of homework party, you can just describe the group.)

1 Pairing Up

Prove that for every even $n \geq 2$, there exists an instance of the stable matching problem with n jobs and n candidates such that the instance has at least $2^{n/2}$ distinct stable matchings.

2 Nothing Can Be Better Than Something

In the stable matching problem, suppose that some jobs and candidates have hard requirements and might not be able to just settle for anything. In other words, each job/candidate prefers being unmatched rather than be matched with those below a certain point in their preference list. The term entity to refer to a candidate/job. A matching could ultimately have to be partial, i.e., some entities would and should remain unmatched.

Consequently, the notion of stability here should be adjusted a little bit to capture the autonomy of both jobs to unilaterally fire employees and/or employees to just walk away. A matching is stable if

- there is no matched entity who prefers being unmatched over being with their current partner;
- there is no matched/filled job and unmatched candidate that would both prefer to be matched with each other over their current status;
- there is no matched job and matched candidate that would both prefer to be matched with each other over their current partners; and
- similarly, there is no unmatched job and matched candidate that would both prefer to be matched with each other over their current status;

- there is no unmatched job and unmatched candidate that would both prefer to be with each other over being unmatched.

(a) Prove that a stable pairing still exists in the case where we allow unmatched entities.

(HINT: You can approach this by introducing imaginary/virtual entities that jobs/candidates "match" if they are unmatched. How should you adjust the preference lists of jobs/candidates, including those of the newly introduced imaginary ones for this to work?)

(b) As you saw in the lecture, we may have different stable matchings. But interestingly, if an entity remains unmatched in one stable matching, it/she must remain unmatched in any other stable matching as well. Prove this fact by contradiction.

3 A Better Stable Pairing

In this problem we examine a simple way to *merge* two different solutions to a stable matching problem. Let R, R' be two distinct stable pairings. Define the new pairing $R \wedge R'$ as follows:

> For every job j, j's partner in $R \wedge R'$ is whichever is better (according to j's preference list) of their partners in R and R'.

Also, we will say that a job/candidate *prefers* a pairing R to a pairing R' if they prefers their partner in R to their partner in R'. We will use the following example:

jobs	preferences	candidates	preferences
A	1>2>3>4	1	D>C>B>A
B	2>1>4>3	2	C>D>A>B
C	3>4>1>2	3	B>A>D>C
D	4>3>2>1	4	A>B>D>C

(a) $R = \{(A,4),(B,3),(C,1),(D,2)\}$ and $R' = \{(A,3),(B,4),(C,2),(D,1)\}$ are stable pairings for the example given above. Calculate $R \wedge R'$ and show that it is also stable.

(b) Prove that, for any pairings R, R', no job prefers R or R' to $R \wedge R'$.

(c) Prove that, for any stable pairings R, R' where j and c are partners in R but not in R', one of the following holds:

- j prefers R to R' and c prefers R' to R; or
- j prefers R' to R and c prefers R to R'.

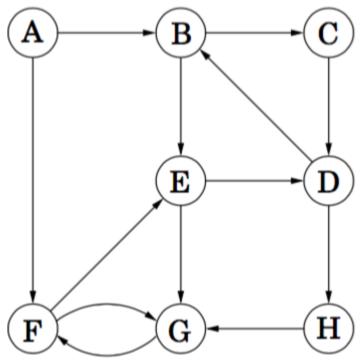
[*Hint*: Let J and C denote the sets of jobs and candidates respectively that prefer R to R', and J' and C' the sets of jobs and candidates that prefer R' to R. Note that $|J| + |J'| = |C| + |C'|$. (Why is this?) Show that $|J| \leq |C'|$ and that $|J'| \leq |C|$. Deduce that $|J'| = |C|$ and $|J| = |C'|$. The claim should now follow quite easily.]

(You may assume this result in the next part even if you don't prove it here.)

(d) Prove an interesting result: for any stable pairings R, R', (i) $R \wedge R'$ is a pairing [*Hint*: use the results from (c)], and (ii) it is also stable.

4 Graph Basics

In the first few parts, you will be answering questions on the following graph G.



(a) What are the vertex and edge sets V and E for graph G?

(b) Which vertex has the highest in-degree? Which vertex has the lowest in-degree? Which vertices have the same in-degree and out-degree?

(c) What are the paths from vertex B to F, assuming no vertex is visited twice? Which one is the shortest path?

(d) Which of the following are cycles in G?

i. $(B,C),(C,D),(D,B)$

ii. $(F,G),(G,F)$

iii. $(A,B),(B,C),(C,D),(D,B)$

iv. $(B,C),(C,D),(D,H),(H,G),(G,F),(F,E),(E,D),(D,B)$

(e) Which of the following are walks in G?

i. (E,G)

ii. $(E,G),(G,F)$

iii. $(F,G),(G,F)$

iv. $(A,B),(B,C),(C,D),(H,G)$

v. $(E,G),(G,F),(F,G),(G,C)$

vi. $(E,D),(D,B),(B,E),(E,D),(D,H),(H,G),(G,F)$

(f) Which of the following are tours in G?

i. (E,G)

ii. $(E,G),(G,F)$

iii. $(F,G),(G,F)$

iv. $(E,D),(D,B),(B,E),(E,D),(D,H),(H,G),(G,F)$

v. $(B,C),(C,D),(D,H),(H,G),(G,F),(F,E),(E,D),(D,B)$

In the following three parts, let's consider a general undirected graph G with n vertices ($n \geq 3$). If true, provide a short proof. If false, show a counterexample.

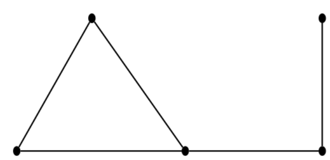
(g) True/False: If each vertex of G has degree at most 1, then G does not have a cycle.

(h) True/False: If each vertex of G has degree at least 2, then G has a cycle.

(i) True/False: If each vertex of G has degree at most 2, then G is not connected.

5 Degree Sequences

The *degree sequence* of a graph is the sequence of the degrees of the vertices, arranged in descending order, with repetitions as needed. For example, the degree sequence of the following graph is $(3,2,2,2,1)$.



For each of the parts below, determine if there exists a simple undirected graph G (i.e. a graph without self-loops and multiple-edges) having the given degree sequence. Justify your claim.

(a) $(3,3,2,2)$

(b) $(3,2,2,2,2,1,1)$

(c) $(6,2,2,2)$

(d) $(4,4,3,2,1)$

6 Proofs in Graphs

Please prove or disprove the following claims.

(a) On the axis from San Francisco traffic habits to Los Angeles traffic habits, Old California is more towards San Francisco: that is, civilized. In Old California, all roads were one way streets. Suppose Old California had n cities ($n \geq 2$) such that for every pair of cities X and Y, either X had a road to Y or Y had a road to X. Prove or disprove that there existed a city which was reachable from every other city by traveling through at most 2 roads.

[*Hint:* Induction]

(b) In lecture, we have shown that a connected undirected graph has an Eulerian tour if and only if every vertex has even degree.

Consider a connected graph G with n vertices which has exactly $2m$ vertices of odd degree, where $m > 0$. Prove or disprove that there are m walks that *together* cover all the edges of G (i.e., each edge of G occurs in exactly one of the m walks, and each of the walks should not contain any particular edge more than once).

7 Bipartite Graphs

An undirected graph is bipartite if its vertices can be partitioned into two disjoint sets L, R such that each edge connects a vertex in L to a vertex in R (so there does not exist an edge that connects two vertices in L or two vertices in R).

(a) Suppose that a graph G is bipartite, with L and R being a bipartite partition of the vertices. Prove that $\sum_{v \in L} \deg(v) = \sum_{v \in R} \deg(v)$.

(b) Suppose that a graph G is bipartite, with L and R being a bipartite partition of the vertices. Let s and t denote the average degree of vertices in L and R respectively. Prove that $s/t = |R|/|L|$.

(c) Prove that a graph is bipartite if and only if it can be 2-colored. (A graph can be 2-colored if every vertex can be assigned one of two colors such that no two adjacent vertices have the same color).