

# CSCI 570 - Fall 2019 - HW 1

Junzhe Liu, 2270250947

September 2, 2019

## 1 Graded Problems

- 1.1 State True/False: An instance of the stable marriage problem has a unique stable matching if and only if the version of the Gale-Shapely algorithm where the male proposes and the version where the female proposes both yield the exact same matching.**

True.

In male propose process, each man always gets their best valid partners while the woman gets their worst valid partners; However in the female propose process, each woman receives best valid partners and the man receives the worst. If both process yield the identical matching, then for each man / woman, their best and worst valid partners are the same person, in another word, there only exist one valid partner for each man or woman. Therefore, the stable marriage problem exist one unique matching result.

- 1.2 A stable roommate problem with 4 students a, b, c, d is defined as follows. Each student ranks the other three in strict order of preference. A match- ing is defined as the separation of the students into two disjoint pairs. A matching is stable if no two separated students prefer each other to their current roommates. Does a stable matching always exist? If yes, give a proof. Otherwise give an example roommate preference where no stable matching exists.**

Stable matchings may not exist, for example, the following image shows one possible situation where no perfect matching is stable:

### Stable roommate problem

Q. Do stable matchings always exist?

A. Not obvious a priori.

Stable roommate problem.

- $2n$  people; each person ranks others from 1 to  $2n - 1$ .
- Assign roommate pairs so that no unstable pairs.

	1 <sup>st</sup>	2 <sup>nd</sup>	3 <sup>rd</sup>
A	B	C	D
B	C	A	D
C	A	B	D
D	A	B	C

no perfect matching is stable

$A-B, C-D \Rightarrow B-C$  unstable

$A-C, B-D \Rightarrow A-B$  unstable

$A-D, B-C \Rightarrow A-C$  unstable

Observation. Stable matchings need not exist.

10

## 1.3 Kleinberg and Tardos, Chapter 1, Exercise 4

### Algorithm 1 Extended Gale-Shapley Algorithm

```

1: Initially all  $r \in R$  and  $h \in H$  are free
2: while some hospital  $h$  has a vacant position and  $h$  hasn't asked every student do
3:   Choose such a hospital  $h$ 
4:   Let  $s$  be the highest-ranked student in  $h$ 's preference list to whom  $h$  has not yet asked
5:   if  $s$  is free then then
6:     assign  $s$  provisionally to  $h$ 
7:   else  $s$  is currently assigned to  $h'$ 
8:     if  $s$  prefers  $h$  to  $h'$  then
9:       assign  $s$  provisionally to  $h$ 
10:      remove  $s$  from  $h'$  positions.
11:     end if
12:   end if
13: end while

```

On termination each undersubscribed hospital has assigned to it all of the students on its reduced list, and a fully assigned hospital with  $q$  places has assigned to it the first  $q$  students on its reduced list.

It can be shown that the matching defined by the provisional assignments on termination is uniquely defined for any given instance of the problem and that it is stable. If a hospital is undersubscribed, no other student is assigned to it in any stable matching. Also, each student is assigned in this matching to his worst valid partner.

#### 1.4 $N$ men and $N$ women were participating in a stable matching process in a small town named Walnut Grove.

Let's simplify this problem as a man-propose process.

First of all, Almazo will propose to Laura, if Laura decides to reject him and stay with her current fiancé (let's say Ben), then Almazo will just simply turn back to Nelly.

However, if Almazo ranks higher than Ben in Laura's preference list, then Laura will accept Almazo as fiancé, leaving Nelly and Ben back to single situation.

Since Nelly remains alone for now, those men who rank Nelly higher than their current partner will step out and propose to Nelly, there comes two situations: 1. If there's no one who prefers Nelly than their partner, then Nelly will still remain alone; 2. If it does exist such a group of men, then Nelly will choose the best one according to her preference. Let's say Nelly chooses Allen, and Allen breaks up with his girlfriend, Alice.

Eventually, the single woman will face the first situation: that is no man who prefers her than his current partner. Let's just suppose Alice is the girl who confronts such awkward situation, what she can do now is to wait and some man to take her hand.

In the meantime, Ben will decide who he will propose to, is it the one who remains alone right now or propose to someone who has paired up. If the single woman ranks higher than Laura, then Ben can just pair up with her. However if she ranks lower than Laura, then he'll have to search all the way down his preference list. Notice that Ben can only pair with those ranks lower than Laura and higher than single woman according to his preference. Suppose Ben successfully paired up with Cindy, and made Bob a bachelor.

Now that the same situation will happen on Bob: to propose to single woman or the others. If he successfully proposed to another woman, then the bachelor identity will switch to another man; However if Alice happens to rank higher than Cindy (Bob's previous partner), then Bob will propose to Alice, the algorithm comes to an end.

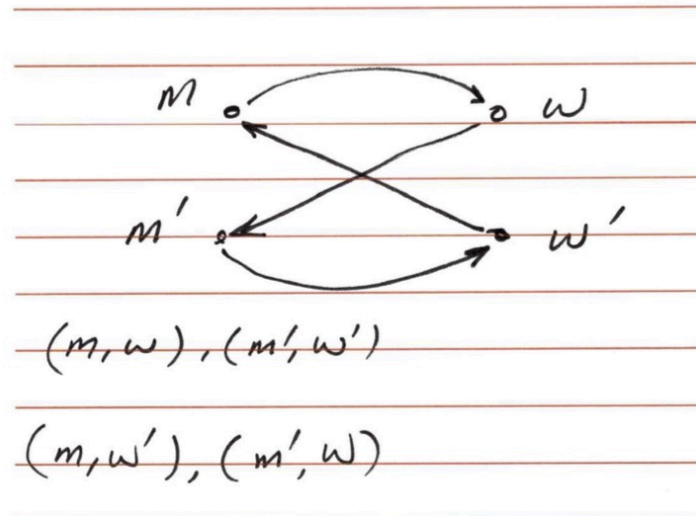
To conclude, each iteration of the algorithm takes 5 steps:

- 1. Each iteration begins with a single woman  $W$  and a single man  $M$
- 2. List out all the man who prefers  $W$  than his current partner, if  $M$  is inside this list, jump out step 3.  $W$  will choose the best one according to her preference. If the list is empty, continue.
- 3. According to  $M$ 's preference list, propose to the next woman.
- 4. If  $W$  pairs up with  $M$ , algorithm stops and quits.
- 5. Renew  $W$  and  $M$ .

## 2 Practice Problems

### 2.1 Kleinberg and Tardos, Chapter 1, Exercise 1.

False. Consider a following situation: the arrow indicates the preference (over another person):



Two possible perfect matching does not include the such pair satisfying the condition stated in the topic.

### 2.2 Kleinberg and Tardos, Chapter 1, Exercise 2.

True. Imagine if a stable matching  $S'$  does not include pair  $(m, w)$ . Suppose  $(m, w') \in S'$ ,  $(m', w) \in S'$ . Then  $(m, w)$  is an unstable pair,  $S'$  is not a stable matching. Therefore, for all possible stable matching,  $(m, w)$  is sure to be one of its element.

### 2.3 Kleinberg and Tardos, Chapter 1, Exercise 3.

There is not always a stable pair of schedules given a set of TV shows and ratings.

For instance, Suppose A has two shows  $(a_1, a_2)$  with ratings of 2 and 4, and B has two shows  $(b_1, b_2)$  with ratings of 1 and 3. Then there are two possible schedules:  $(a_1, b_1), (a_2, b_2)$  or  $(a_1, b_2), (a_2, b_1)$ . In the first situation, B will consider switch the two shows in order to win one slot; While in the second situation, A will switch two shows to win both slots rather than only one. Therefore, no matter what the situation is, there will always exist one network who wishes to change its schedule, this is an counterexample.