CS 5084 - Assignment 1

Kleinberg and Tardos - Chapter 01

Ehu Shubham Shaw

January 28, 2025

1 - Stable Matching Problem - Exercise 1.1 Analysis

Every stable matching solution has to include a man m and a woman w who rank each other first in their respective preference lists. After careful analysis, I have determined this statement to be false, and I will demonstrate this with a counterexample. Consider a simple Stable Matching Problem involving two men (m1, m2) and two women (w1, w2). Let's arrange the preference lists as follows:
For men, m1 prefers w1 over w2. m2 prefers w1 over w2.
For women, w1 likes m1 over m2, whereas w2 prefers m2.
In this case, we can see that m1 and w1 rank each other first in their preference lists, satisfying the problem's requirement. However, I'll show that there is a stable matching in which m1 and w1 are not paired together.

Consider the matching S = {(m2, w1), (m1, w2).}.

This matching is stable since:
First, we look at m1's possibility for forming a blocking pair. Although m1 prefers w1 above his current companion, w2, w1 is pleased with her match, m2. Thus, m1 cannot form a blocking pair with w1.
Second, consider m2's situation: He's already coupled with his first option w1, thus he has no reason to establish a blocking partnership with anyone else.Third, from the perspective of w1, she is paired with m2, and while she likes m1, m1 is unable to create a blocking pair with her, as previously established.
Finally, w2 prefers m2 over her existing partner m1, and m2 prefers w1 to w2, therefore no blocking pair can develop here either. This counterexample conclusively demonstrates that the original statement is untrue. Even if a man and a woman rank each other top, they are not always paired in stable matches. This exposes an intriguing aspect of the Stable Matching Problem: reciprocal first preferences do not always determine the final matching in all stable solutions.Through this analysis, we can conclude that stability in matchings is more complex than simply matching mutual first preferences.

2 - Exercise 1.3 - Television Network Scheduling Stability

The subject involves an interesting application of stability principles to television network scheduling. We must investigate whether stable scheduling pairings always exist between two competing networks and develop an algorithm to find such pairs if they exist.

I shall show that stable couples always exist by offering a constructive algorithm for creating them. I will then establish its correctness.
Proposed algorithm: "Opposite Rating Assignment"
First the Initial Setup Sort shows from both networks in descending order according to ratings.
Number of time slots from one to n. Secondly I Schedule Assignment. Of Network A: Assign shows from highest to lowest rating (slots 1–n) and Network B: Assign shows from lowest to highest rating (slots 1–n).

For Example: For n = 4, now we see:

Network A Shows	Network B. Shows
A1 rating: 100.	B1 rating: 95.
A2 rating: 90.	B2 rating: 85.
A3 rating: 80.	B3 rating: 75.
A4: 70 rating;	B4: 65 rating.

In the first time slot, Network A had A1 with a score of 100, while Network B had B4 with a score of 65. The winner of this match was Network A, specifically A1.
In the second time slot, Network A had A2 with a score of 90, while Network B had B3 with a score of 75. The winner was Network A, specifically A2.
In the third time slot, Network A had A3 with a score of 80, while Network B had B2 with a score of 85. The winner was Network B, specifically B2.
In the fourth time slot, Network A had A4 with a score of 70, while Network B had B1 with a score of 95. The winner was Network B, specifically B1.
The stability of the television network scheduling problem is proven by analyzing the current state and the impossibility of unilateral improvement. Network A wins the first half of the slots by airing higher-rated shows against Network B's lower-rated ones, while Network B wins the second half under the reverse scenario. Any attempt by Network A to rearrange its schedule would result in matching its higher-rated shows against B's strongest shows, leading to a net loss of currently won slots. Similarly, Network B cannot improve its position by shifting its highest-rated shows earlier, as they would then compete against even stronger shows from Network A. The correctness of the algorithm is ensured by its termination in O(n \log n) time due to sorting, its stability as proven through the inability of either network to gain an advantage through unilateral changes, and its completeness for any set of n shows with distinct ratings. By arranging schedules in opposite order of ratings, the algorithm guarantees a natural equilibrium where no network can improve its standing, thus always producing a stable scheduling pair

3. (25 points) Exercise 1.5: The Stable Matching Problem assumes

In this topic, we look at the stable matching problem, which allows men and women to have ties in their preference lists. This means they may rank two or more people equally rather than following a fixed sequence. We investigate two types of instability: strong and weak instability.

(a) Is it always possible to find a perfect match without strong instability?

When a man and a woman choose each other over their matched mates, there is a high level of instability. We need to see if we can always discover a match in which this never occurs.
The answer is yes; we can always discover something that matches. We can use the modified version of the Gale-Shapley method, which operates as follows.1. Men propose to their favorite woman (or any of their top choices).2. Women evaluate proposals and tentatively select the best one based on their preferences, breaking ties in a consistent fashion.3. Rejected men propose to their next pick, and the process continues until everyone is matched.
The Gale-Shapley algorithm ensures that no man and woman who favor each other are left unpaired, as well as the absence of strong instability. As a result, a perfect match without significant instability is always possible.

(b) Is it always possible to find a perfect match without weak instability?

A weak instability occurs when a man and a woman either prefer each other to their partners, or when one favors the other while the other is indifferent. In this example, the answer is no; such a perfect match does not often occur. Here's an example featuring two males and two women: Women's preferences: - W_1 ranks m_1 and m_2 similarly.- W_2 evaluates m_1 and m_2 equally. - Men's preferences: - M_1 ranks w_1 and w_2 similarly - M_2 evaluates w_1 and w_2 equally.Everyone sees the two possibilities as equal, hence no matter how we pair them, there will always be at least one weakly unstable pair. For example, if we match (m_1, w_1) with (m_2, w_2), m_2 may prefer w_1, while w_1 is indifferent, resulting in weak instability. So The Gale-Shapley algorithm can identify a perfect match without strong instability, ties can result in weakly unstable pairs, making perfect matching impossible.

4. Exercise 1.8 - Analysis of Truthfulness in Stable Matching

I'll see if a lady can get a better match in the Gale-Shapley algorithm by misrepresenting her preferences. Specifically, I will show through a counterexample that a woman can enhance her match by rearranging the order of two men on her preference list, even if both are ranked low in her genuine preferences.

Consider the following scenario: three men (m1, m2, m3) and three women (w1, w2, and w3).

In this scenario, the men and women have ranked their preferences for potential partners. Man 1 has the highest preference for Woman 1, followed by Woman 2, and then Woman 3 as his least preferred choice. Man 2, on the other hand, prefers Woman 2 the most, with Woman 1 as his second choice, and Woman 3 as his least preferred. Similarly, Man 3 also prefers Woman 1 the most, followed by Woman 2, and lastly Woman 3.

On the women's side, Woman 1 has the strongest preference for Man 2, followed by Man 3, with Man 1 being her least preferred option. Woman 2 ranks Man 3 as her top choice, then Man 1, and finally Man 2. Lastly, Woman 3 prefers Man 1 above all, followed by Man 2, with Man 3 as her least preferred match.

On the women's side, Woman 1 has the strongest preference for Man 2, followed by Man 3, with Man 1 being her least preferred option. Woman 2 ranks Man 3 as her top choice, then Man 1, and finally Man 2. Lastly, Woman 3 prefers Man 1 above all, followed by Man 2, with Man 3 as her least preferred match.

When we execute the Gale-Shapley algorithm with the given real preferences, we get stable matching pairs: Man 1 with Woman 2, Man 2 with Woman 1, and Man 3 with Woman 3.

Assume Woman 1 falsely alters her preference list by swapping the final two options, resulting in her new ranking favoring Man 2 the most, followed by Man 1, and placing Man 3 last rather than Man 1. Running the Gale-Shapley method with the amended preference list yields a different result. This time, Man 1 is partnered with Woman 3, Man 2 stays with Woman 1, and Man 3 is coupled with Woman 2. This change in choice by a single person leads to a different final match.

On the women's side, Woman 1 has the strongest preference for Man 2, followed by Man 3, with Man 1 being her least preferred option. Woman 2 ranks Man 3 as her top choice, then Man 1, and finally Man 2. Lastly, Woman 3 prefers Man 1 above all, followed by Man 2, with Man 3 as her least preferred match.

In the second pairing, w1 is paired with m2, whom she actually favors over both m1 and m3. This illustrates that by faking her preferences for two less favored men, w1 was able to match with her true most preferred man, m2.

The essential discovery is that by making oneself appear more responsive to m1 (by ranking him higher than m3), w1 triggered a series of proposals that eventually led to her receiving a proposal from her most desired partner, m2. This works because the Gale-Shapley algorithm favors proposers (men in this example), and by deliberately modifying the rejection chain, a woman can occasionally manipulate the algorithm to her benefit.

On the women's side, Woman 1 has the strongest preference for Man 2, followed by Man 3, with Man 1 being her least preferred option. Woman 2 ranks Man 3 as her top choice, then Man 1, and finally Man 2. Lastly, Woman 3 prefers Man 1 above all, followed by Man 2, with Man 3 as her least preferred match.

This example conclusively proves component (b) of the question, which states that there are circumstances in which a woman can improve her match by rearranging the order of two males on her preference list. This finding has significant implications for the Gale-Shapley algorithm's practical implementation because it demonstrates that participants may be motivated to misrepresent their genuine preferences.

This study also reveals an intriguing asymmetry in the Gale-Shapley algorithm: while it is well known that males (proposers) cannot gain by lying about their preferences, we have demonstrated that women (receivers) can. This implies that the algorithm's ability to be strategy-proof only applies to one side of the matching.