COL866: Special Topics in Algorithms **Fall 2023**

Assignment 1

Total points: 100 Deadline: Aug 21 (Monday)

1. a) [**5 points**] Show that under the men-proposing DA algorithm, there is always at least one woman who receives exactly one proposal.

> By way of contradiction, suppose every woman receives at least two proposals. Consider the round in which the *last* proposal is made under the DA algorithm. The woman receiving this proposal must either be previously unmatched (in which case she receives two or more proposals simultaneously in that round) or be tentatively matched already. In both cases, some man will be rejected by this woman, prompting a future proposal. This contradicts the fact that the proposal under consideration was the last one.

b) [**5 points**] Suppose there are n men and n women. As a function of n, what is the maximum number of proposals that can be made during the DA algorithm? (Hint: Use the above result.)

> By the above result, there is at least one woman who receives exactly one proposal. Each of the other $(n-1)$ women can receive at most n proposals each. Thus, the maximum number of proposals can be $n(n-1)+1 = n^2 - n + 1$.

c) [**5 points**] For an arbitrary n, construct an instance where the number of proposals made by men under the DA algorithm matches the bound shown by you above.

> The stable matching instance given below results in $n^2 - n + 1$ proposals.

$$
\begin{array}{lllllll}
m_1: & w_1 & w_2 & w_3 & \dots & w_{n-1} & w_n \\
m_2: & w_2 & w_3 & \dots & w_{n-1} & w_1 & w_n \\
m_3: & w_3 & w_4 & \dots & w_1 & w_2 & w_n \\
 & & & \vdots & & & \\
m_{n-1}: & w_{n-1} & w_1 & w_2 & \dots & w_{n-1} & w_n \\
m_n: & w_1 & w_2 & w_3 & \dots & w_{n-1} & w_n
\end{array}
\qquad
\begin{array}{lllllll}
w_1: & m_2 & m_3 & \dots & m_{n-1} & m_n & m_1 \\
w_2: & m_3 & m_4 & \dots & m_n & m_1 & m_2 \\
w_3: & m_4 & m_5 & \dots & m_1 & m_2 & m_3 \\
 & & & \vdots & & & \\
w_{n-1}: & m_n & m_1 & \dots & m_{n-3} & m_{n-2} & m_{n-1} \\
w_n: & m_1 & m_2 & m_3 & \dots & m_{n-1} & m_n
\end{array}
$$

2. [**10 points**] In this exercise, we will show that the strategy of repeatedly fixing blocking pairs may not give a stable matching. Consider the matching instance given below:

$$
\begin{array}{llll}
m_1: & w_2 & w_1 & w_3 \\
m_2: & w_1 & w_2 & w_3 \\
m_3: & w_1 & w_2 & w_3
\end{array}
\qquad
\begin{array}{llll}
w_1: & m_1 & m_3 & m_2 \\
w_2: & m_3 & m_1 & m_2 \\
w_3: & m_3 & m_1 & m_2
\end{array}
$$

Show that there is a cyclic sequence of unstable matchings in the above instance such that each matching in sequence can be obtained from its predecessor by "fixing" a blocking pair (i.e., by matching the blocking agents with each other, and also matching their previously assigned partners with each other).

Start with the matching $\mu_1 = \{(m_1, w_1), (m_2, w_2), (m_3, w_3)\}$.

- Fix the blocking pair (m_1, w_2) to get the matching $\mu_2 = \{(m_1, w_2), (m_2, w_1), (m_3, w_3)\}$.
- Fix the blocking pair (m_3, w_2) in μ_2 to get the matching $\mu_3 = \{(m_1, w_3), (m_2, w_1), (m_3, w_2)\}$.
- Fix the blocking pair (m_3, w_1) in μ_3 to get the matching $\mu_4 = \{(m_1, w_3), (m_2, w_2), (m_3, w_1)\}$.
- Fix the blocking pair (m_1, w_1) in μ_4 to get back μ_1.

This example is due to Knuth (1997); see Example 2 in Chapter 1.

3. Define the *cost* of a stable matching as sum of ranks of matched partners of all agents (men and women). For example, for the stable matching given by the underlined outcomes in the instance below, the cost is 1+1+2+1+4+1+2+3 = 15.

m_1:	$\underline{w_3}$	w_2	w_1	w_4	w_1:	m_4	m_3	m_1	$\underline{m_2}$
m_2:	$\underline{w_1}$	w_4	w_2	w_3	w_2:	$\underline{m_4}$	m_3	m_2	m_1
m_3:	w_2	$\underline{w_4}$	w_1	w_3	w_3:	m_3	$\underline{m_1}$	m_2	m_4
m_4:	$\underline{w_2}$	w_1	w_3	w_4	w_4:	m_2	m_1	$\underline{m_3}$	m_4

a) [**5 points**] For a general n, construct an instance with n men and n women and a stable matching for that instance with cost $n(n+1)$.

Consider any instance where, if man m ranks a woman w at k^{th} position, then w ranks m at $(n+1-k)^{\text{th}}$ position. The sum of ranks for any man-woman pair is $(n+1)$, and any stable matching will contain n such man-woman pairs, giving the desired cost.

b) [**15 points**] Prove that for any given preferences of n men and n women, the cost of *any* stable matching is at most $n(n+1)$.

Suppose, for contradiction, that there is a stable matching μ with cost strictly greater than $n(n+1)$.

Let M and W denote the set of all men and women, respectively.

For any man m, let $S_m := \{w \in W : w \succ_m \mu(m)\}$ denote the set of all women that m strictly prefers over his μ-partner. Likewise, for any woman w, let $S_w := \{m \in M : m \succ_w \mu(w)\}$ denote the set of all men that w strictly prefers over her μ-partner. Further, let us call a man-woman pair $(m, w) \in M \times W$ *man-improving* if $w \in S_m$, and *woman-improving* if $m \in S_w$. Note that by stability of μ, no man-woman pair can be simultaneously man-improving and woman-improving.

Observe that by definition of cost, we have

$$\begin{aligned}\sum_{m \in M} |S_m| + \sum_{w \in W} |S_w| &= \text{cost of } \mu - 2n \\ &> n(n+1) - 2n \\ &= n^2 - n.\end{aligned}$$

That is, the total number of man-improving and woman-improving pairs is strictly greater than $n^2 - n$.

Since there are n men and n women, there can be n^2 distinct man-woman pairs overall. Out of these, n pairs are contained in the stable matching μ. Among the remaining $n^2 - n$ pairs, only a subset can be either man-improving or woman-improving. Thus, the number of such pairs cannot exceed $n^2 - n$, giving us the desired contradiction.

Proof adapted from Tanya Khovanova's Math Blog.

4. [**10 points**] Recall from Lecture 3 that when a woman manipulates *optimally* under the men-proposing DA algorithm, the resulting matching is guaranteed to be stable with respect to the true preferences.

   Define *suboptimal* manipulation by a woman as a misreport where she gets a partner who, according to her true list, is better than her true match but worse than her optimally manipulated match (that is, a suboptimal manipulation is better than telling the truth, but not as good as optimal manipulation). Provide an instance where the DA matching after suboptimal manipulation is *not* stable with respect to the true preferences.

Consider the stable matching instance given below (Vaish and Garg, 2017, Example 1) where the DA outcomes are underlined:

$$\begin{array}{lllll} m_1\colon & \underline{w_2} & w_1 & w_3 & w_4 \\ m_2\colon & \underline{w_3} & w_1 & w_2 & w_4 \\ m_3\colon & \underline{w_1} & w_2 & w_3 & w_4 \\ m_4\colon & w_1 & \underline{w_4} & w_2 & w_3 \end{array} \qquad \begin{array}{lllll} w_1\colon & m_1 & m_2 & \underline{m_3} & m_4 \\ w_2\colon & m_3 & \underline{m_1} & m_2 & m_4 \\ w_3\colon & m_1 & \underline{m_2} & m_3 & m_4 \\ w_4\colon & m_1 & m_2 & m_3 & \underline{m_4} \end{array}$$

The woman w_1 can optimally manipulate by reporting $m_1 \succ m_4 \succ m_2 \succ m_3$ to get matched with man m_1, her top choice according to her true preferences. Observe that the optimal misreport is obtained in an "inconspicuous" manner by promoting m_4 in the true list of w_1.

There also exists a suboptimal misreport $m_2 \succ m_4 \succ m_3 \succ m_1$ which gets w_1 matched with man m_2. This is an improvement over her true match m_3 but is strictly worse than her optimal match m_1. The resulting matching $\{(m_1, w_3), (m_2, w_1), (m_3, w_2), (m_4, w_4)\}$ is not stable with respect to the true preferences as the pair (m_1, w_1) blocks it.

5. Consider the stable matching instance given below:

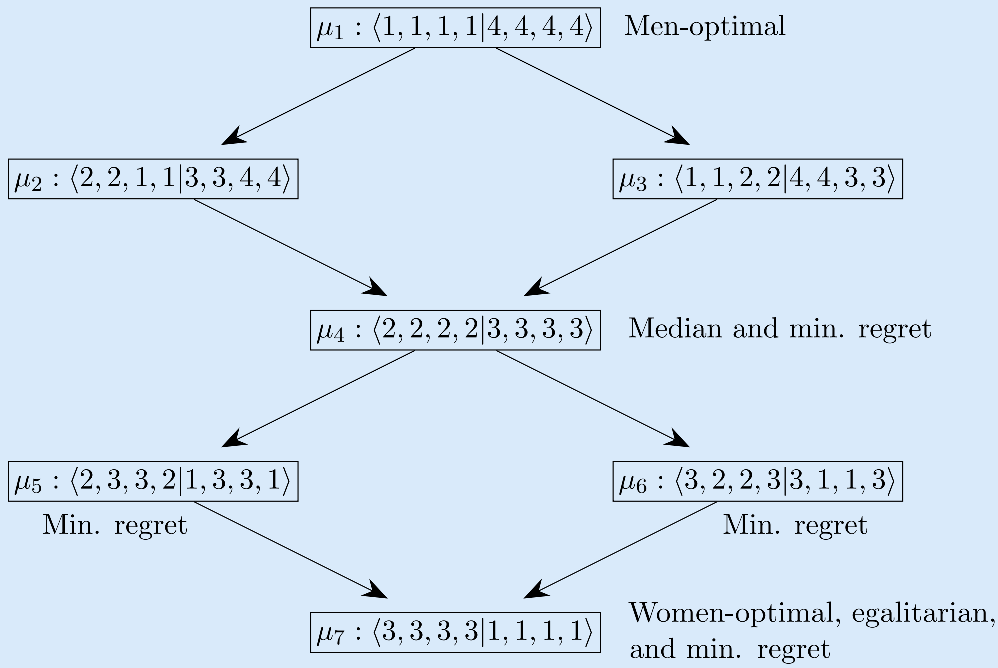
$$\begin{array}{lllll} m_1\colon & w_1 & w_2 & w_3 & w_4 \\ m_2\colon & w_2 & w_1 & w_4 & w_3 \\ m_3\colon & w_3 & w_4 & w_1 & w_2 \\ m_4\colon & w_4 & w_3 & w_2 & w_1 \end{array} \qquad \begin{array}{lllll} w_1\colon & m_3 & m_4 & m_2 & m_1 \\ w_2\colon & m_4 & m_3 & m_1 & m_2 \\ w_3\colon & m_1 & m_2 & m_4 & m_3 \\ w_4\colon & m_2 & m_1 & m_3 & m_4 \end{array}$$

a) [**10 points**] List all matchings that are stable for this instance.

The following seven matchings are stable for the given instance:

- $\mu_1 = \{(m_1, w_1), (m_2, w_2), (m_3, w_3), (m_4, w_4)\}$
- $\mu_2 = \{(m_1, w_2), (m_2, w_1), (m_3, w_3), (m_4, w_4)\}$
- $\mu_3 = \{(m_1, w_1), (m_2, w_2), (m_3, w_4), (m_4, w_3)\}$
- $\mu_4 = \{(m_1, w_2), (m_2, w_1), (m_3, w_4), (m_4, w_3)\}$
- $\mu_5 = \{(m_1, w_2), (m_2, w_4), (m_3, w_1), (m_4, w_3)\}$
- $\mu_6 = \{(m_1, w_3), (m_2, w_1), (m_3, w_4), (m_4, w_2)\}$
- $\mu_7 = \{(m_1, w_3), (m_2, w_4), (m_3, w_1), (m_4, w_2)\}$

b) [**5 points**] Illustrate the lattice of stable matchings for the above instance. (You may find it convenient to use the "vector of ranks of matched partners" notation discussed in Lecture 2.)



The arrow heads denote the direction of women's preferences (men's preferences are in the opposite direction).

c) [**5 points**] Identify the men-optimal, women-optimal, median, egalitarian, and minimum regret matchings. If multiple matchings satisfy a given criterion (e.g., median, egalitarian, etc.), identify all matchings that do.

Shown in the figure above.

6. [**10 points**] Prove or disprove the following statement for many-to-one stable matchings: It is impossible to have one stable matching where a hospital is matched only with its 1st and 4th choices, and another stable matching in which it is matched only with its 2nd and 3rd choices.

We will prove that it is impossible to have one stable matching where a hospital is matched only with its 1st and 4th choices and another stable matching in which it is matched only with its 2nd and 3rd choices.

Suppose, for contradiction, that there exists an instance where a hospital, say h, is matched with its 1st and 4th choice doctors, say d_1 and d_4, respectively, in one stable matching, say μ_1, and is matched with its 2nd and 3rd choice doctors, say d_2 and d_3, respectively, in another stable matching, say μ_2.

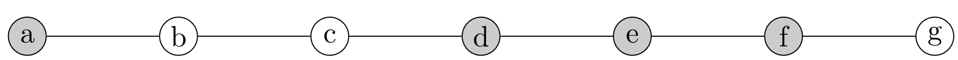
From the rural hospitals theorem, it follows that hospital h must be saturated (indeed, if it was unsaturated, then it must be matched with the same *set* of doctors in all stable matchings).

Consider the canonical one-to-one instance associated with such a many-to-one instance. Denote the copies of hospital h in the canonical instance by h^1 and h^2. Observe that under both matchings μ_1 and μ_2, the "more preferred" doctor must be matched with the "more preferred" copy of hospital h. That is, under μ_1, doctor d_1 is matched with h^1 and doctor d_4 is matched with h^2, while under μ_2, doctor d_2 is matched with h^1 and doctor d_3 is matched with h^2. If this is not the case, then under either μ_1 or μ_2, h^1 and the "better" doctor will constitute a blocking pair as the doctors have the same preferences over the copies of any hospital.

We will now show that under the matching μ_1, doctor d_2 must be matched with a hospital that is distinct from h. Indeed, under μ_1, doctor d_2 cannot be unmatched as it will then create a blocking pair with the copy h^2. Thus, d_2 must be a matched to a hospital, say h', that it strictly prefers over h, i.e., $h' \succ_{d_2} h$.

Let us now define the "min" mapping for the one-to-one matchings μ_1 and μ_2 wherein each doctor points to its less preferred partner while each copy of a hospital points to its more preferred partner. In the resulting mapping, d_2 will point to h^1 (with whom it is matched under μ_2) while h^1 points to d_1 (with whom it is matched under μ_1). This, however, shows that the induced mapping is not a matching, contradicting the lattice theorem for one-to-one stable matchings.

7. Consider a two-hospital kidney exchange setting with seven patient-donor pairs. The compatibility graph among these patient-donor pairs is as shown below. Each node in this graph represents a patient-donor pair and each edge denotes the possibility of a two-way exchange between adjacent nodes (we will assume that only two-way exchanges are allowed). The shaded (respectively, unshaded) nodes belong to hospital 1 (respectively, hospital 2).

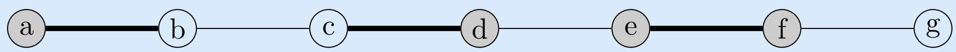


Suppose the hospitals are strategic agents who are only interested in getting as many of their own nodes matched as possible. Each hospital can choose to hide a subset of its patient-donor pairs from the centralized exchange (and match them internally if possible). Thus, hospital 1 (respectively, hospital 2) can hide any subset of the shaded (respectively, unshaded) nodes. The centralized exchange only sees the nodes revealed by the two hospitals as well as the edges (if any) between pairs of revealed nodes.

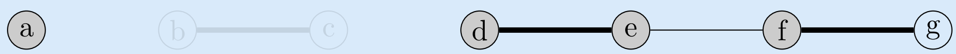
a) [**5 points**] Suppose the centralized exchange uses a deterministic maximum matching algorithm. Show that under any such algorithm, some hospital will have an incentive to not reveal all of its nodes. (In other words, show that any maximum matching algorithm fails to be strategyproof.)

Consider the instance shown in the figure above where both hospitals reveal all of their patient-donor pairs. The centralized exchange will pick one of the following four maximum matchings:

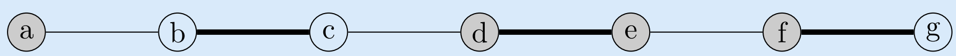
Case 1: $\{(a,b),(c,d),(e,f)\}$ leaving the node g unmatched



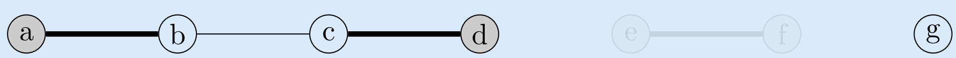
In this case, hospital 2 (which owns the unshaded nodes) can deviate by concealing the nodes b and c and matching them internally. The maximum matching on the revealed nodes will match the node g, leading to a strict improvement for hospital 2.



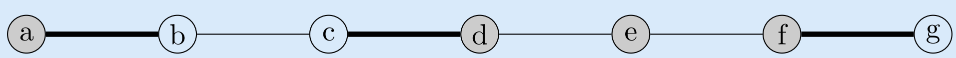
Case 2: $\{(b,c),(d,e),(f,g)\}$ leaving the node a unmatched



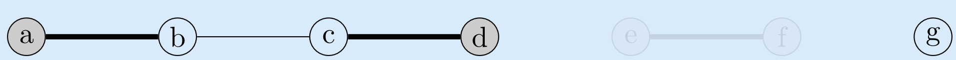
In this case, hospital 1 (which owns the shaded nodes) can deviate by concealing the nodes e and f and matching them internally. The maximum matching on the revealed nodes will match the node a, leading to a strict improvement for hospital 1.



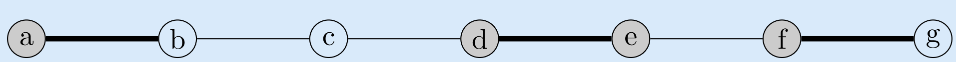
Case 3: $\{(a,b),(c,d),(f,g)\}$ leaving the node e unmatched



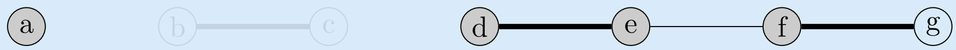
In this case, hospital 1 (which owns the shaded nodes) can deviate by concealing the nodes e and f and matching them internally. The maximum matching on the revealed nodes will match the nodes a and d, leading to a strict improvement for hospital 1.



Case 4: $\{(a,b),(d,e),(f,g)\}$ leaving the node c unmatched



In this case, hospital 2 (which owns the unshaded nodes) can deviate by concealing the nodes b and c and matching them internally. The maximum matching on the revealed nodes will match the node g, leading to a strict improvement for hospital 2.



Thus, regardless of which maximum matching the centralized exchange selects on the original instance, one of the hospitals can improve by concealing a subset of its nodes. This implies that any maximum matching algorithm fails strategyproofness.

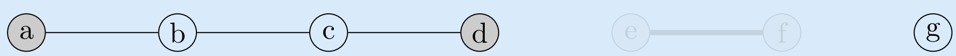
b) [**10 points**] Now suppose the centralized exchange uses a deterministic strategyproof algorithm.[1] Show that any such algorithm, on some input, will be forced to match at most half as many nodes as the maximum matching algorithm. (In other words, no deterministic strategyproof algorithm can guarantee more than half the size of a maximum matching.)

A strategyproof algorithm is presented in (Ashlagi et al., 2015). For the purpose of this exercise, we will simply assume that a strategyproof algorithm exists.

Consider once again the original instance where both hospitals reveal all of their patient-donor pairs. This instance has an odd number of nodes, and therefore at least one node will remain unmatched, resulting in the following two mutually exclusive and exhaustive cases.

Case 1: At least one of the shaded nodes is not matched in the original instance.

In this case, consider another input instance wherein hospital 1 deviates from the original instance by hiding the nodes e and f (and matches them internally), as shown below.

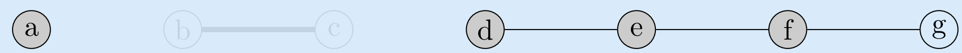


On this input, the centralized exchange *cannot* match both a and d as that would constitute a profitable deviation for hospital 1. Therefore, the centralized exchange can match at most one of the edges (a, b), (b, c) or (c, d). On the other hand, a maximum matching algorithm will necessarily select the edges (a, b) and (c, d). Therefore, the centralized exchange is forced to match at most half as many nodes as the maximum matching algorithm on this input.

Case 2: At least one of the unshaded nodes is not matched in the original instance.

By a similar argument as before, consider a deviation by hospital 2 from the original instance wherein it hides the nodes b and c (and matches them internally).

[1] Showing the existence of a strategyproof algorithm is non-trivial. For the purpose of this exercise, you can simply assume that there exists some strategyproof algorithm.



On this input, a maximum matching algorithm will necessarily select the edges (d, e) and (f, g). However, strategyproofness forces the centralized exchange to leave the node g unmatched.

References

Itai Ashlagi, Felix Fischer, Ian A Kash, and Ariel D Procaccia. Mix and Match: A Strategyproof Mechanism for Multi-Hospital Kidney Exchange. *Games and Economic Behavior*, 91:284–296, 2015. (Cited on page 8)

Donald Ervin Knuth. *Stable Marriage and its Relation to Other Combinatorial Problems: An Introduction to the Mathematical Analysis of Algorithms*, volume 10. American Mathematical Soc., 1997. (Cited on page 2)

Rohit Vaish and Dinesh Garg. Manipulating Gale-Shapley Algorithm: Preserving Stability and Remaining Inconspicuous. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 437–443, 2017. (Cited on page 4)