Exercises after Lecture 10 (M7): Matching models.

Problem 1: Solve your own problem

This problem is meant to demonstrate the power of DA algorithm, which finds a stable matching in *any* marriage market.

Consider a market with four men and four women. Come up with arbitrary preferences for all players (i.e., a ranking for each player of all players on the other side of the market and the option to stay single).

1. Find a stable matching generated by men-proposing DA algorithm.
2. Find a stable matching generated by women-proposing DA algorithm.
3. Are there any other stable matchings?
4. Suppose a men-proposing DA algorithm is run. Is there a profitable deviation for any of the women – i.e., can any woman misreport her preferences to the mechanism to improve her matching? If yes, show it; if not, explain why.

   (*Hint: such a deviation exists if and only if you have more than one stable matching, which happens if and only if the outcomes of W-DA and M-DA algorithms are different.*)

Solution

We consider a marriage market with four men, denoted $M = \{m_1, m_2, m_3, m_4\}$, and four women, denoted $W = \{w_1, w_2, w_3, w_4\}$. We assume the following arbitrary ordinal prefererences of men over women (and the option of remaining single, denoted by the name of the player himself) and of women over men (and the same option):

$$
\begin{aligned}
&m_1 : w_1 \succ_{m_1} w_2 \succ_{m_1} m_1 \succ_{m_1} w_3 \succ_{m_1} w_4 \qquad && w_1 : m_4 \succ_{w_1} m_3 \succ_{w_1} w_1 \succ_{w_1} m_2 \succ_{w_1} m_1\\
&m_2 : w_4 \succ_{m_2} w_1 \succ_{m_2} w_2 \succ_{m_2} m_2 \succ_{m_2} w_3 && w_2 : m_3 \succ_{w_2} m_4 \succ_{w_2} w_2 \succ_{w_2} m_2 \succ_{w_2} m_1\\
&m_3 : w_3 \succ_{m_3} w_4 \succ_{m_3} w_1 \succ_{m_3} w_2 \succ_{m_3} m_3 && w_3 : m_1 \succ_{w_3} m_4 \succ_{w_3} m_3 \succ_{w_3} w_3 \succ_{w_3} m_2\\
&m_4 : m_4 \succ_{m_4} w_3 \succ_{m_4} w_4 \succ_{m_4} w_1 \succ_{m_4} w_2 && w_4 : m_4 \succ_{w_4} m_2 \succ_{w_4} m_3 \succ_{w_4} w_4 \succ_{w_4} m_1.
\end{aligned}
$$

(1) We proceed to find the matching $\mu_{MDA} : M \cup W \to M \cup W$ generated by the men-proposing deferred acceptance algorithm. At stage 0, all men propose to their most preferred partner (or simply opt to remain single). Thus, m_1 proposes to w_1, m_2 proposes to w_4, m_3 proposes to w_3 and m_4 opts to remain single. The woman w_1 has one offer, but prefers remaining single, so she rejects m_1, while w_3 and w_4 also have one offer, which they hold on to. At stage 1, all men have outstanding offers (or have retired from the marriage market) except m_1, who proposes to w_2. She also prefers remaining single to marrying m_1, so she rejects his offer. At stage 2, m_1 is still the only man without an outstanding offer. He prefers remaining single to marrying either of the remaining women. Therefore, matching is finalised at this stage. The resulting matching is

$$\mu_{MDA} = ((m_1, m_1), (m_2, w_4), (m_3, w_3), (m_4, m_4), (w_1, w_1), (w_2, w_2)).$$

(2) We now find the matching $\mu_{WDA} : M \cup W \to M \cup W$ generated by the women-proposing deferred acceptance algorithm. At stage 0, all women propose to their most preferred partner. Thus, w_1 proposes to m_4, w_2 proposes to m_3, w_3 proposes to m_1 and w_4 proposes to m_4. The men m_3 and m_1 have one offer each, however m_1 prefers remaining single so he rejects the offer from w_3, while m_3 holds on to the offer from w_2. The man m_4 has two offers. However, he prefers remaining single and rejects both. At stage 1, w_2 has an outstanding offer, so she does nothing. The woman w_1 proposes to m_3, w_3 proposes to m_4 and w_4 proposes to m_2. The latter now has one offer, which he holds on to, while m_4 still prefers remaining single, so that he rejects the offer from w_3. Finally, m_3 has two offers, one from w_1 and one from w_2. He prefers w_1 and holds on to her offer. Accordingly, he rejects w_2. At stage 2, w_1 and w_4 have outstanding offers, so they do nothing. The woman w_2 prefers remaining single over her other options. She therefore exits the marriage market. The woman w_3 proposes to m_3 who now has two offers. He prefers the offer from w_3 and rejects w_1. At stage 3, w_1 is the only woman without outstanding offers. She prefers remaining single to her remaining options, so no new offers are made and matching is finalised. The resulting matching is

$$\mu_{WDA} = \left(\left(w_1, w_1\right), \left(w_2, w_2\right), \left(w_3, m_3\right), \left(w_4, m_2\right), \left(m_1, m_1\right)\left(m_4, m_4\right)\right).$$

(3) First, we see that the set of singles is the same in both matchings (μ_{WDA} and μ_{MDA}), as expected. Then, we observe that the two matchings are in fact identical, i.e. $\mu_{WDA} = \mu_{MDA}$. Because of this, this is the unique stable matching.

(4) We only have one stable matching, therefore if a men-proposing deferred acceptance algorithm is run, there is no profitable deviation for any of the women.

Problem 2: College admissions

This problem demonstrates how marriage model can be extended to allow many-to-one matchings, which turns it into a "college admissions model".

There is a market with four students $S = \{s_1, ..., s_4\}$ and three colleges $C = \{c_1, c_2, c_3\}$. College c_1 can admit two students (its *quota* is $q_1 = 2$); the remaining two colleges can admit one student each ($q_2 = q_3 = 1$). Players' preferences (ordinal rankings, written best to worst) are given by

$$\begin{aligned}
&\succ_{s_1}: c_3, c_1, c_2 && \succ_{c_1}: s_1, s_2, s_3, s_4 \\
&\succ_{s_2}: c_2, c_1, c_3 && \succ_{c_2}: s_1, s_2, s_3, s_4 \\
&\succ_{s_3}: c_1, c_3, c_2 && \succ_{c_3}: s_3, s_1, s_2, s_4 \\
&\succ_{s_4}: c_1, c_2, c_3 &&
\end{aligned}$$

Your goal is to find a stable matching in this problem. The only difference from the marriage model we considered in class is that college c_1 can admit *two* students. The trick is to represent the two available spots in c_1 as two independent players which have the same preferences over students and which rank equally against other colleges among the students.

In particular, consider instead a market with the same four students but now four colleges $C' = \{c_{1.1}, c_{1.2}, c_2, c_3\}$ (each with quota $q_i = 1$, as in the marriage model), and preferences are given by

$$\begin{aligned}
&\succ_{s_1}: c_3, c_{1.1}, c_{1.2}, c_2 && \succ_{c_{1.1}}: s_1, s_2, s_3, s_4 \\
&\succ_{s_2}: c_2, c_{1.1}, c_{1.2}, c_3 && \succ_{c_{1.2}}: s_1, s_2, s_3, s_4 \\
&\succ_{s_3}: c_{1.1}, c_{1.2}, c_3, c_2 && \succ_{c_2}: s_1, s_2, s_3, s_4 \\
&\succ_{s_4}: c_{1.1}, c_{1.2}, c_2, c_3 && \succ_{c_3}: s_3, s_1, s_2, s_4
\end{aligned}$$

1. Use the college-proposing DA algorithm to find a stable matching.
2. Matching μ generated by the C-DA algorithm is C'-optimal. However, there is another matching

$\mu' = \{(c_1, s_2, s_4), (c_2, s_1), (c_3, s_3)\}$ that is strictly preferred to μ by all colleges in C. How can you explain this contradiction?

Solution

(1) We want to find the matching μ generated by the college-proposing deferred acceptance algorithm, where the two available slots in c_1 are represented as two separate colleges $c_{1,1}$ and $c_{1,2}$.

- At stage 0, all colleges make an offer of admission to their preferred students: $c_{1,1}$, $c_{1,2}$ and c_2 all make offers to s_1 while c_3 makes an offer to s_3. The latter has got one offer, and he holds on to it. The student s_1 prefers the offer from $c_{1,1}$, so he rejects the offers from $c_{1,2}$ and c_2.
- At stage 1, c_3 and $c_{1,1}$ have outstanding offers, so they do nothing, while $c_{1,2}$ and c_2 both make offers to s_2, who prefers the offer from c_2. In consequence, she rejects the offer from $c_{1,2}$.
- At stage 2, $c_{1,1}$, c_2 and c_3 all have outstanding offers, so they do nothing. The college $c_{1,2}$ makes and offer to s_3 who prefers this offer to the one from c_3. In consequence, he rejects the latter college and holds on to the offer from $c_{1,2}$.
- A stage 3, c_3 is the only college without outstanding offers. It makes an offer to s_1, who prefers this to the offer from $c_{1,1}$.
- At stage 4, the latter is the only college without outstanding offers. It makes an offer of admission to s_2, who prefers what she already had and rejects the offer. A stage 4, $c_{1,1}$ makes and offer to s_3 who prefers this to the offer from $c_{1,2}$. He therefore rejects the latter offer and accepts the one from $c_{1,1}$.
- Now, at stage 5, $c_{1,2}$ is the only college without outstanding offers. It makes and offer to s_4, who now has one offer. She holds on to it.
- At stage 6, all colleges have outstnading offers, so no new offers are made. Therefore, matching is finalised. The resulting matching is

$$\mu = ((c_{1,1}, s_3), (c_{1,2}, s_4), (c_2, s_2), (c_3, s_1)).$$

(2) C-DA matching μ is strongly Pareto-optimal for colleges among **stable** matchings. The suggested matching μ' is not stable.

Problem 3: Book giveaway

Djul has defended his Ph.D. and found a job. He looks back at the small library of books that he has assembled during his studies and decides that he does not need them as much any more. Therefore, he decides to give the books away to fellow Ph.D. students. Suppose there are $b \in \{1, ..., B\}$ books and $i \in \{1, ..., N\}$ interested students. Since $N > B$, Djul decides that it would be fair to limit the giveaway to one book per person. Let $\theta_{i,b}$ denote the valuation of student i for book b (privately known by student i). Assume that all students are economists who act in pure self-interest.

1. Given that Ph.D. students are poor,[1] and Djul himself now has a well-paying job, he would prefer to give the books away for free. Propose a mechanism that Djul could use to allocate the books among fellow students for free and in a way that would be Pareto optimal.
2. Suppose now that $N = 6$, $B = 4$, and the realized valuations are as given in Table 1. Calculate the allocation produced by your mechanism from part 1.

[1]The story is taking place in the U.S.

$\theta_{i,b}$	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$
$b=1$	4	4	1	8	9	9
$b=2$	0	2	4	9	5	3
$b=3$	9	5	5	2	6	4
$b=4$	7	6	0	7	2	6

Table 1: Preferences for the Book Giveaway problem

3. Does there exist a mechanism that allocates the books without transfers efficiently (i.e., in a welfare-maximizing way)? If yes: present a mechanism. If not: explain why.
4. Djul has run your mechanism from part 1 and messaged people regarding who got which book, but lost his phone with all the notes and messages before actually giving any books away. He thus cannot remember which book was promised to which student. Each student, however, knows which book they were promised. How can Djul recover the promised allocation without running the whole mechanism again? (Propose a mechanism that relies on students' reports of the books they were promised and explain why it works.)

Solution

1. There are a few alternatives. Djul could use the deferred acceptance algorithm with students proposing in a random order (analogous to random serial dictatorship in social choice). This could effectively be implemented via a "first come-first serve" rule. Since in this situation every student gets their most preferred book (out of those that were not most preferred by preceding students), there is no scope for Pareto-improving exchanges, hence the resulting book allocation is Pareto-optimal. Note, however, that it need not be welfare maximizing. E.g., let there be two students, two books, $\theta_{1,b} = (10, 9)$ and $\theta_{2,b} = (10, 1)$, and student $i = 1$ gets to choose first. Then student 1 gets book 1 and student 2 gets book 2, which yields welfare 11, but they could *trade*, rather than just exchange, books (with student 2 paying student 1 any amount in $[1, 9]$), to arrive at an allocation that yields welfare 19.

   The same issue arises if we try to use the Top Trading Cycles algorithm with any arbitrary initial allocation – the resulting allocation would be Pareto-efficient for the same reason, but not necessarily welfare-maximizing for the same reason.

2. Take the DA algorithm, in which students select books in the order of their indices. Then student 1 picks book 3, student 2 picks book 4, student 3 picks book 2 (since 3 was taken), and student 4 picks book 1.

3. The arguments in part 1 suggest that the standard matching algorithms are not efficient. While we would typically resort to VCG to implement an efficient allocation, it is not an option in this case since the goal is to avoid payments. Using non-monetary transfers like time or effort would, strictly speaking, fulfill the goal (hence VCG would be an acceptable answer if a non-monetary implementation of transfers is specified), but it defeats the spirit of the problem, since the intent is to not impose extra burden on the students.

   At the same time, without transfers the allocation can not be implemented, which is easy to see from the IC conditions. The example given in part 1 with realized types $\theta_{1,b} = (10, 9)$ and $\theta_{2,b} = (10, 1)$ shows that DRM $(k^*, t = 0)$ is not DSIC, since under this realized type profile k^* prescribes that student 1 gets book 2, but they would prefer to misreport their valuation vector as, e.g., $\hat{\theta}_{1,b} = (11, 1)$ in order to get book 1, which they prefer more. The fact that DRM $(k^*, t = 0)$ is not BIC follows from the same example if we assume that $\theta_{2,b} = (10, 1)$ is the only type possible for player 2.

4. Consider the following mechanism: if every book is claimed by exactly one student, implement the re-

ported allocation, otherwise burn all books in a book-burning van. This mechanism has an equilibrium in which all reports are truthful. To see this, note that no student i has an incentive to report the book they like less than the one they were assigned (or report no book), since this cannot result in i getting a better book. On the other hand, if i reports a more preferred book b than the one they were assigned, Pareto-optimality implies that this book is claimed by some other student j, who reports truthfully in equilibrium – hence book b is claimed by both i and j, which triggers the burn clause in the mechanism, and neither of them gets any book. This outcome is worse for i than getting the initially allocated book, hence this deviation is not profitable either. So none of the available deviations is profitable for i, and i was arbitary, hence truthtelling is an equilibrium of the game induced by this mechanism.