Chapter 11

Mechanism Design Without Money

Chapter 10 talks about mechanism design with money — we have an allocation rule and we wish to implement it in a strategic environment by incentivizing players suitably. We now discuss mechanism design without money. We begin with one of the most successful application of mechanism design without money which is stable matching.

11.1 Stable Matching

In the stable matching problem, we have two sets $\mathcal{A}$ and $\mathcal{B}$ of n vertices each. Each vertex in $\mathcal{A}$ has a strict preference order (which is a complete order) over $\mathcal{B}$ and each vertex in $\mathcal{B}$ has a strict preference order over $\mathcal{A}$. We are interested in a matching $\mathcal{M}$ of $\mathcal{A}$ with $\mathcal{B}$ which is "stable" in the following sense. Intuitively speaking, a matching $\mathcal{M}$ of $\mathcal{A}$ and $\mathcal{B}$ is "not stable" is there exist two vertices $a \in \mathcal{A}$ and $b \in \mathcal{B}$ which are not matched together and both of them prefer to be matched together compared to their current states. Such a pair (a, b) of vertices is called a "blocking pair" of the matching $\mathcal{M}$. In an unstable matching, a blocking pair can match themselves with one another thereby breaking the matching $\mathcal{M}$ justifying the instability of $\mathcal{M}$. The framework of stable matching has a lot of applications, for example, matching men with women in a marriage portal, doctors with hospitals, applicants with jobs. We now formally define the stable matching problem.

Blocking Pair

Definition 11.1. Suppose we have two sets $\mathcal{A}$ and $\mathcal{B}$ of n vertices each. Every vertex $a \in \mathcal{A}$ has a preference order $\succ_a$ over $\mathcal{B}$ and every vertex $b \in \mathcal{B}$ has a preference order $\succ_b$ over $\mathcal{A}$. Given a matching $\mathcal{M}$ between $\mathcal{A}$ and $\mathcal{B}$, a pair $a \in \mathcal{A}$ and $b \in \mathcal{B}$ is called a blocking pair for the matching $\mathcal{M}$ if one of the following holds.

(i) Both a and b are not matched to anyone in $\mathcal{M}$.

(ii) The vertex a is unmatched, b is matched with some $\mathcal{M}(b) \in \mathcal{A}$ in $\mathcal{M}$, and $a \succ_b \mathcal{M}(b)$.

(iii) The vertex b is unmatched, a is matched with some $\mathcal{M}(a) \in \mathcal{B}$ in $\mathcal{M}$, and $b \succ_a \mathcal{M}(a)$.

(iv) Both a and b are matched to $\mathcal{M}(a)$ and $\mathcal{M}(b)$ respectively in $\mathcal{M}$, and we have $a \succ_b \mathcal{M}(b)$ and

$b \succ_a \mathcal{M}(a)$.

A matching is called stable if there is no blocking pair.

Stable Matching

Definition 11.2. Suppose we have two sets $\mathcal{A}$ and $\mathcal{B}$ of n vertices each. Every vertex $a \in \mathcal{A}$ has a preference order $\succ_a$ over $\mathcal{B}$ and every vertex $b \in \mathcal{B}$ has a preference order $\succ_b$ over $\mathcal{A}$. A matching $\mathcal{M}$ is called stable if there is no blocking pair.

It follows from Definition 11.2 that every stable matching is a perfect matching. We now show the main result of stable matching theory – every instance of stable matching problem has a stable matching. Moreover, such a stable matching can be computed in polynomial time. This is the celebrated Theorem by Gale and Shapley.

Gale-Shapley Theorem

Theorem 11.1. Every stable matching instance has a stable matching. Moreover it can be computed in polynomial time.

Proof. Let us call the sets $\mathcal{A}$ and $\mathcal{B}$ as the sets of men and women respectively. We now describe an algorithm called "proposal algorithm by men" or "men-proposing deferred acceptance algorithm." Initially everyone is unmatched. We pick any arbitrary unmatched man and let him propose his most preferred woman whom he has not yet proposed (irrespective of whether she is currently matched or not). Whenever an unmatched woman receives a proposal she accepts the proposal and gets matched. When a matched woman receives a proposal, she accepts her proposal if she prefers the proposer more than her current matched partner; in this case, her previous matched partner becomes unmatched again. The algorithm terminates when all men (and thus all women) are matched.

Observe that, once a woman gets matched, she never becomes unmatched although she can change her partner. Also, every woman who has received at least on proposal is matched. Since there are n men and each has a preference list over n women, after at most n^2 iterations, every woman receives at least one proposal and thus all the women (and thus all the men) are matched. Hence, the algorithm terminates after at most n^2 iterations. What we show next is that the matching $\mathcal{M}$ computed by the algorithm is a stable matching. For that, we take an arbitrary pair $a \in \mathcal{A}$ and $b \in \mathcal{B}$ of man and woman who are not matched together by $\mathcal{M}$. We will show that (a, b) does not form a blocking pair. If a has never proposed to b during the run of the algorithm, then a prefers his matching partner in $\mathcal{M}$ than b and thus (a, b) cannot form a blocking pair. On the other hand, if a has proposed to b and still b is not matched with a in $\mathcal{M}$, then b prefers her partner in $\mathcal{M}$ than a and thus (a, b) cannot form a blocking pair. □

We observe that the proposal algorithm described in Theorem 11.1 does not specify which unmatched man is picked to propose. Can different order of unmatched men result into different stable matching to be output by the algorithm? The answer is no! To prove this, we will actually prove something stronger. For a man $a \in \mathcal{A}$, let us define $h(a)$ to be the most preferred woman (according to $\succ_a$) that a can be matched in any stable matching (as there could be multiple stable matchings). The following result shows that, in the matching output by the Gale-Shapley algorithm, every man a is matched with $h(a)$. This matching is called men-optimal stable matching.

Men-Optimal Stable Matching

Theorem 11.2. In any matching $\mathcal{M}$ computed by the proposal algorithm by men, every man $a \in \mathcal{A}$ is matched with $h(a)$ in $\mathcal{M}$.

Proof. Consider any fixed execution of the Gale-Shapley algorithm by men. Let us define $\mathcal{R}_i = \{(a, b) \in \mathcal{A} \times \mathcal{B} : b \text{ has rejected } a \text{ in the first } i \text{ iterations}\}$ and $\mathcal{R} = \cup_{i \in [n^2]} \mathcal{R}_i$. To prove the statement, it is enough to show that, for any pair $(a, b) \in \mathcal{R}$, there is no stable matching which matches a with b. We show this claim by induction on the number of iterations i.

For $i = 0$, $\mathcal{R}_0$ is an empty set and thus the statement is vacuously true. Hence the induction starts. Let us assume the statement for i and prove for $i+1$. If no woman has rejected any matched man in the $(i+1)$-th iteration, then we have $\mathcal{R}_{i+1} = \mathcal{R}_i$ and the claim follows from induction hypothesis. So let us assume that some woman $b \in \mathcal{B}$ received a proposal from a man $a \in \mathcal{A}$ in the $(i+1)$-th iterations and b has rejected her previous partner, say $a' \in \mathcal{A}$. That is, we have $\mathcal{R}_{i+1} = \mathcal{R}_i \cup \{(a', b)\}$. Then we obviously have $a \succ_b a'$. By induction hypothesis, the man a is not matched with any woman whom he prefers more than b in any stable matching. For the sake of finding a contradiction, let us assume that there is a stable matching $\mathcal{M}'$ which matches the woman b with a'. Since the man a is not matched with any woman whom he prefers more than b in $\mathcal{M}'$, the pair (a, b) forms a blocking pair for $\mathcal{M}'$ which contradicts our assumption that $\mathcal{M}'$ is a stable matching. $\square$

Along the line of men-optimal stable matching, one can define notion of men-pessimal, women-optimal, and women-pessimal stable matchings. It can be shown using similar argument as the proof of Theorem 11.2 that the stable matching computed by the Gale-Shapley proposal algorithm by men is actually a women-pessimal stable matching.

The notion of stable matching safe-guards against every pair of man and woman deviating from a matching – in a stable matching there is no pair of man and woman so that they deviate from the matching under consideration and both become happier by getting matched with one another. One can think that we could be more demanding that we wish to safe-guard against any coalition of men and women. In a stable matching, can it happen that there is a coalition of men and women who can deviate from the matching under consideration and get matched among themselves in a way that no one in the coalition is unhappier than before and at least one of them is happier than previous matching? The next result shows that it cannot happen – it is enough to safe-guard against a pair of man and woman. In the context of the stable matching problem, a matching is called a core solution if no coalition of men and women get matched among themselves in a way so that no one is the coalition is worse off and at least one of them is better off. More abstractly, a core allocation is defined as follows.

Core Allocation

Definition 11.3. For every subset $\mathcal{S} \subset [n]$ of players, let $\mathcal{I}_{\mathcal{S}}$ be an index set and $\mathcal{K}_{\mathcal{S}} = \{(k_i^j)_{i\in\mathcal{S}} : j \in \mathcal{I}_{\mathcal{S}}\}$ the set of allocations when only players in $\mathcal{S}$ are present; define $\mathcal{K}_i = \{k_i : \exists \mathcal{S} \subseteq [n], (k_i, (k_j)_{j\in\mathcal{S}\setminus\{i\}}) \in \mathcal{K}_{\mathcal{S}}\}$ for $i \in [n]$. Every player $i \in [n]$ has a preference $\geqslant_i$ which is a partial order over $\mathcal{K}_i$. We say an allocation $(k_1, \ldots, k_n) \in \mathcal{K}_{[n]}$ is not a core allocation if there exists a subset $\mathcal{S} \subseteq [n]$ of players and an allocation $(k_i, (k_j)_{j\in\mathcal{S}\setminus\{i\}}) \in \mathcal{K}_{\mathcal{S}}$ such that

(i) $k'_i \geqslant_i k_i \;\; \forall\, i \in \mathcal{S}$

(ii) $k'_i >_i k_i$ for some $i \in \mathcal{S}$

Otherwise the allocation $(k_1, \ldots, k_n)$ is called a core allocation.

In case of the stable matching problem, the set $\mathcal{K}_{\mathcal{S}}$ of allocations for any $\mathcal{S} \subseteq [n]$ is

$$\mathcal{K}_{\mathcal{S}} = \left\{\left((\mathcal{M}(a))_{a\in\mathcal{A}\cap\mathcal{S}}, (\mathcal{M}(b))_{b\in\mathcal{B}\cap\mathcal{S}}\right) : \mathcal{M} \text{ is a perfect matching in } (\mathcal{A}\cup\mathcal{B})\cap\mathcal{S}\right\}$$

Typically, the set $\mathcal{K}$ of allocations would be clear from the context and we often skip defining it formally. Observe that, in general, we may not have any core allocation in some situations. However, in case of stable matching problem, core allocations are precisely the stable matchings.

Stable Matchings are Core Allocations

Theorem 11.3. In every instance of the stable matching problem, a matching is a core if and only if it is a stable matching.

Proof. By definition, every core matching is a stable matching. For the other direction, let $\mathcal{M}$ be a stable matching. If $\mathcal{M}$ is not a core matching, let $\mathcal{S}$ be a subset of men and women of smallest cardinality who can opt out of the matching $\mathcal{M}$ and get matched among themselves which makes, w.l.o.g, a man $a \in \mathcal{S}$ happier by getting matched with a woman $b \in \mathcal{S}$; that is $b \succ_a \mathcal{M}(a)$ where $\mathcal{M}(a)$ is the partner of a in $\mathcal{M}$. Observe that, $a \neq \mathcal{M}(b)$ and thus we have $a \succ_b \mathcal{M}(b)$. Then (a, b) forms a blocking pair for $\mathcal{M}$ contradicting our assumption that $\mathcal{M}$ is a stable matching. □

The last question on stable matching which we discuss here is regarding strategy-proofness. Is the men-proposing deferred acceptance algorithm strategy-proof for the players? The following example answers this question negatively. Consider a setting with 3 men m_1, m_2, m_3 and 3 women w_1, w_2, w_3 and the preferences be such that the men-proposing deferred acceptance algorithm outputs the matching $\{\{m_1, w_2\}, \{m_2, w_3\}, \{m_3, w_1\}\}$ and only woman w_1 receives 2 proposals whereas the other 2 women receive 1 proposal each. Suppose the woman w_1 received proposals from m_2 and m_3 (in this order), and rejects m_3. One such concrete example is the following and run of the men-proposal deferred acceptance algorithm with following proposal order of men: m_2, m_1, m_2, m_3.

Preference of m_1 :	$w_2 \succ w_1 \succ w_3$	Preference of w_1 :	$m_1 \succ m_3 \succ m_2$
Preference of m_2 :	$w_1 \succ w_3 \succ w_2$	Preference of w_2 :	$m_3 \succ m_1 \succ m_2$
Preference of m_3 :	$w_1 \succ w_2 \succ w_3$	Preference of w_3 :	$m_1 \succ m_3 \succ m_2$

Now suppose, the woman w_1 rejects m_3 instead of m_2 when she receives a proposal from m_2 – this happens if the woman w_1 misreports her preference to be $m_1 \succ m_2 \succ m_3$. Then the run of the men-proposal deferred acceptance algorithm with proposal order $m_2, m_1, m_1, m_3, \ldots$ of men results in matching $\{\{m_1, w_1\}, \{m_2, w_3\}, \{m_3, w_2\}\}$ which is better for the woman w_1 compared to the matching $\{\{m_1, w_2\}, \{m_2, w_3\}, \{m_3, w_1\}\}$.

So the men-proposal deferred acceptance algorithm can be manipulated by women. Does there exist any algorithm for stable matching which is strategy-proof? The answer turns out to be negative [Rot08]. However, the following result shows that any man does not have any incentive to misreport in the men-proposal deferred acceptance algorithm. That is, the men-proposal deferred acceptance algorithm is strategy-proof for men.

Strategy-Proofness of Men-Proposal Deferred Acceptance Algorithm for Men

Theorem 11.4. The men-proposal deferred acceptance algorithm is strategy proof for men.

Proof. Suppose not, then there exists a man, say $m \in \mathcal{A}$ who can get better partner by misreporting his preference to be $\succ'_m$ instead of $\succ_m$. Let $\mathcal{M}$ and $\mathcal{M}'$ be the stable matchings output by the men-proposal deferred acceptance algorithm on true preference profile $(\succ_m, \succ_{-m})$ and the profile $(\succ'_m, \succ_{-m})$. Let $w \in \mathcal{B}$ be the partner of m in $\mathcal{M}'$. By assumption, we have $w \succ_m \mathcal{M}(m)$. Let $m_2 \in \mathcal{A}$ be the partner of w in $\mathcal{M}$. Since (m, w) does not form a blocking pair for $\mathcal{M}$, we have $\mathcal{M}(w) = m_2 \succ_w m$. Let $\mathcal{M}'(m_2) = w_2$ and again we have $\mathcal{M}'(m_2) = w_2 \succ_{m_2} \mathcal{M}(m_2) = w$ otherwise (m_2, w) forms a blocking pair for $\mathcal{M}'$. Let us define $\mathcal{A}_1 = \{a \in \mathcal{A} : \mathcal{M}'(a) \succ_a \mathcal{M}(a)\}$. Then the above argument shows that the partner of any partner of any man in $\mathcal{A}_1$ also belongs to $\mathcal{A}_1$. Let us define $\mathcal{B}_1 = \{\mathcal{M}'(a) : a \in \mathcal{A}_1\} = \{\mathcal{M}(a) : a \in \mathcal{A}_1\}$.

Obviously we have $m \in \mathcal{A}_1$ and thus $\mathcal{A}_1, \mathcal{B}_1 \neq \emptyset$. Consider the run of the men-proposal deferred acceptance algorithm on input $(\succ_m, \succ_{-m})$. Suppose t be the last iteration where some man $m_1 \in \mathcal{A}_1$ makes a proposal to some woman w_1; by the choice of t, the woman w_1 must have accepted the proposal by the man m_1 in the t-th iteration. Since every man in $\mathcal{A}_1$ gets worse partner in $\mathcal{M}$ than $\mathcal{M}'$, every man $m' \in \mathcal{A}_1$ has made a proposal to the woman $\mathcal{M}'(m')$ within the first $(t-1)$ iterations and eventually gets rejected and thus every woman has received at least one proposal within the first $(t-1)$ iterations. By the choice of m_1, the woman w_1 rejects some man $m'' \in \mathcal{A} \setminus \mathcal{A}_1$. So we have $m_1 \succ_{w_1} m''$ and $m'' \succ_{w_1} \mathcal{M}'(w_1)$ since $\mathcal{M}'(w_1)$ also must have proposed w_1. On the other hand, since we have $m'' \notin \mathcal{A}_1$, it follows that $w_1 \succ_{m''} \mathcal{M}(m'') \succcurlyeq_{m''} \mathcal{M}'(m'')$. Then (m'', w_1) forms a blocking pair for $\mathcal{M}'$ which is a contradiction. $\square$

11.2 House Allocation

In the stable matching problem, both the entities, namely men and women, have preferences over other. However, in many applications of matching, for example, campus house allocation, school admission, etc. only one type of entity has a preference. These situations are modelled by house allocation problem. In the house allocation problem, we have n agents each occupying a house (so we have n houses in total). Each agent also has a preference (a complete order) over the set of n houses. The question that we are interested in is whether we can reallocate the houses which makes the agents better off. We consider the *Top Trading Cycle Algorithm (TTCA)* which was credited to Gale in [SS74].