Platform Choice in Matching Markets

Varun Bansal [1] Dinko Dimitrov [2]

[1]Indian Statistical Institute, Delhi, India

[2]Saarland University, Saarbruecken, Germany

20th November, 2024

Background: The Two-Sided Matching Problem

- The two-sided matching problem is a well-established and extensively studied theoretical concept.
- Examples: Matching men to women, matching doctor-hospital matching, the school choice problem.
- Gale and Shapley (1962) provided the deferred acceptance algorithm (DAA) which delivered a stable matching for each two-sided matching problem.
- The DAA is also a strategy proof mechanism for the proposing side.

Platform Choice: Motivation

- What if the preferences of both sides are not clearly well defined?
- In college choice, students might have a well-defined ranking over colleges, while schools will have to rely on students' performances and their prior colleges.
- Job listers have to place listings on platforms and have to rely on some platform-level signals on the job seeker's qualifications.
- The academic job market relies on the job seeker's letters of recommendation which are taken to be signals of the job seeker's ability.

A Platform Choice Game

- We allow for platforms to act as signaling devices for the job seeker's ability.
- Each job is connected to a few platforms, and the ranking of each job over agents is dependent upon the platform choice of each agent, and the job's valuation of each platform it is connected to.
- The job seekers are the strategic agents who select the platforms to maximize their potential match given other job seeker's choices.

Model

- There are a finite number of job seekers denoted by the set A (hereafter referred to as agents) and job listers denoted by the set J, we assume that $|A| = |J| = n$.

- Each agent $a \in A$ has a strict ranking over job listers denoted by $\succ_a$.

- There exists a set of platforms denoted by P such that each platform is connected to a subset of jobs.

- The set of jobs connected to platform p is denoted by J_p, the set of platforms connected to job j is denoted by P_j.

- (Full Connectivity): $\cup_{p \in P} J_p = J$ and $\cup_{j \in J} P_j = P$.

Model Continued

- Each platform has a strict ranking over all agents in A denoted by $\succ_p$ which is known to the agents.
- These can be interpreted as the individual platform's signals should an agent join the platform.
- Each job j has a strict ranking over the platforms in P_j denoted by $\succ_j$ which is known to the agents.
- This represents the value each job attaches to the platforms where the job is listed.

An Illustrative Example

Suppose there are three agents and three jobs: $A = \{a_1, a_2, a_3\}$ and $J = \{j_1, j_2, j_3\}$ and two platforms $P = \{p_1, p_2\}$. Connections are as follows: $P_{j_1} = \{p_1, p_2\}$, $P_{j_2} = \{p_1\}$ and $P_{j_3} = \{p_1, p_2\}$. The rankings as discussed are depicted below:

$\mathbf{a}_1$	$\mathbf{a}_2$	$\mathbf{a}_3$	$\mathbf{p}_1$	$\mathbf{p}_2$	$\mathbf{j}_1$	$\mathbf{j}_2$	$\mathbf{j}_3$
j_1	j_1	j_3	a_2	a_3	p_2	p_1	p_1
j_2	j_3	j_2	a_1	a_1	p_1		p_2
j_3	j_2	j_1	a_3	a_2			

The above information will be enough to completely describe a platform choice game. We will next describe how a platform choice game is played.

The Platform Choice Game

- The job seekers are the only strategic agents in a platform choice game.
- The game is played in two stages.
- In stage one, each agent picks a platform. An arbitrary strategy profile in stage one is denoted by $(p_a)_{a \in A}$.
- In stage two, a deferred acceptance algorithm is run such that agents can only propose to the connected jobs of the platform chosen in stage one.

The Platform Choice Game

- Note that it is a dominant strategy for each agent to report their true preferences in stage two, so without loss of generality, the strategy profile in stage one $(p_a)_{a \in A}$ is equivalent to a (subgame perfect) strategy profile of the complete game.
- To complete the description of the game, we need to discuss what the job listers do when proposed to by multiple agents since they do not have preferences over agents.
- We denote by $DAA((p_a)_{a \in A})$ the matching induced by a strategy profile $(p_a)_{a \in A}$.

Aggregation

- Jobs value signals from different platforms differently, and each platform ranks the agents who joins it.
- A natural way for jobs to aggregate this information is the **lexicographic aggregation**.
- If $p \succ_j p'$, then if both (a, p) and (a', p') propose to job j, then (a', p') is rejected.
- If both (a, p) and (a', p) propose to job j, then (a', p) is rejected if $a \succ_p a'$.

Structure of the Talk

- Existence of a Nash equilibrium in the platform choice game.
- Structure of the set of NE in the platform choice game.
- A discussion on Comparability of NE.
- A general way of aggregating preferences by the job listers.
- A discussion on many-to-one matching platform choice games.

Result 1: Full Employment

Theorem

For every platform choice game, all agents in a Nash equilibrium are employed.

- For a strategy profile $(p_a)_{a \in A}$. If $(a', \emptyset) \in DAA((p_a)_{a \in A})$, then there exists a unilateral deviation such that in the deviated strategy profile $(p_{-a'}, p')$ such that $p' \neq p_{a'}$: $(a', j) \in DAA((p_{-a'}, p'))$ for some $j \in J$.

- This result is crucial and deceptively non-trivial.

Back to the Example

In the example discussed above:

$\mathbf{a}_1$	$\mathbf{a}_2$	$\mathbf{a}_3$	$\mathbf{p}_1$	$\mathbf{p}_2$	$\mathbf{j}_1$	$\mathbf{j}_2$	$\mathbf{j}_3$
j_1	j_1	j_3	a_2	a_3	p_2	p_1	p_1
j_2	j_3	j_2	a_1	a_1	p_1		p_2
j_3	j_2	j_1	a_3	a_2			

The strategy profile $S = \{p_2, p_1, p_2\}$ induces the matching $M = (a_1, \emptyset), (a_2, j_3), (a_3, j_1))$. Result 1 states that such a strategy profile cannot be a NE. The deviated strategy profile $S' = \{p_1, p_1, p_2\}$ induces the matching $M' = \{(a_1, j_2), (a_2, j_1), (a_3, j_3)\}$.

Back to the Example

In the example discussed above:

$\mathbf{a}_1$	$\mathbf{a}_2$	$\mathbf{a}_3$	$\mathbf{p}_1$	$\mathbf{p}_2$	$\mathbf{j}_1$	$\mathbf{j}_2$	$\mathbf{j}_3$
j_1	j_1	j_3	a_2	a_3	p_2	p_1	p_1
j_2	j_3	j_2	a_1	a_1	p_1		p_2
j_3	j_2	j_1	a_3	a_2			

The strategy profile $S = \{p_2, p_1, p_2\}$ induces the matching $M = (a_1, \emptyset), (a_2, j_3), (a_3, j_1))$. Result 1 states that such a strategy profile cannot be a NE. The deviated strategy profile $S' = \{p_1, p_1, p_2\}$ induces the matching $M' = \{(a_1, j_2), (a_2, j_1), (a_3, j_3)\}$.

In fact, S' is a Nash equilibrium of this game. What can we say about the existence of a Nash equilibrium in a platform choice game?

Result 2: Existence

Theorem

A Nash equilibrium exists in every platform choice game.

If we have lexicographic aggregation, a NE exists in every platform choice game. The proof of this result is indirect, as we describe a crucial aspect of the structure of the set of Nash equilibria.

The Induced Two-Sided Matching Problem

- We map each platform choice game to a two-sided matching problem.
- Each platform choice game is mapped to a *unique* two-sided matching problem.
- This map will provide us with a result on the *existence* and *structure* of the set of NE of a platform choice game.
- The map is as follows: For a given platform choice game the relevant two-sided matching problem is one where the preferences of the agents are the same, and those of the jobs coincide with the preferences of its highest ranked platform.

Back to the Example

For the illustrative example:

$\mathbf{a}_1$	$\mathbf{a}_2$	$\mathbf{a}_3$	$\mathbf{p}_1$	$\mathbf{p}_2$	$\mathbf{j}_1$	$\mathbf{j}_2$	$\mathbf{j}_3$
j_1	j_1	j_3	a_2	a_3	p_2	p_1	p_1
j_2	j_3	j_2	a_1	a_1	p_1		p_2
j_3	j_2	j_1	a_3	a_2			

The relevant two-sided matching problem is the following:

$\mathbf{a}_1$	$\mathbf{a}_2$	$\mathbf{a}_3$	$\mathbf{j}_1$	$\mathbf{j}_2$	$\mathbf{j}_3$
j_1	j_1	j_3	a_3	a_2	a_2
j_2	j_3	j_2	a_1	a_1	a_1
j_3	j_2	j_1	a_2	a_3	a_3

Result 3: Implementation of Stable Matchings

Theorem

Fix a platform choice problem and let M be a stable matching for the induced two-sided matching problem. Then there exists a Nash equilibrium $(p_a)_{a \in A}$ in the platform choice game such that $DAA((p_a)_{a \in A}) = M$.

Result 3: Implementation of Stable Matchings

Theorem

Fix a platform choice problem and let M be a stable matching for the induced two-sided matching problem. Then there exists a Nash equilibrium $(p_a)_{a \in A}$ in the platform choice game such that $DAA((p_a)_{a \in A}) = M$.

Given Gale and Shapley (1962)'s result on the existence of a stable matching in a two sided-matching problem, this proves Result 2 as a corollary. It also proves that a NE of the platform choice game need not be unique.

Result 3: Implementation of Stable Matchings

Theorem

Fix a platform choice problem and let M be a stable matching for the induced two-sided matching problem. Then there exists a Nash equilibrium $(p_a)_{a \in A}$ in the platform choice game such that $DAA((p_a)_{a \in A}) = M$.

Given Gale and Shapley (1962)'s result on the existence of a stable matching in a two sided-matching problem, this proves Result 2 as a corollary. It also proves that a NE of the platform choice game need not be unique.

What about the converse?

Converse is not True

We look at the illustrative example again:

$\mathbf{a}_1$	$\mathbf{a}_2$	$\mathbf{a}_3$	$\mathbf{p}_1$	$\mathbf{p}_2$	$\mathbf{j}_1$	$\mathbf{j}_2$	$\mathbf{j}_3$
j_1	j_1	j_3	a_2	a_3	p_2	p_1	p_1
j_2	j_3	j_2	a_1	a_1	p_1		p_2
j_3	j_2	j_1	a_3	a_2			

Here the strategy profile discussed before $S = (p_1, p_1, p_2)$ is a NE and $DAA(S) = \{(a_1, j_2), (a_2, j_1), (a_3, j_3)\} = M$. However, as can be seen from the induced two-sided matching problem below, it is not stable. (a_1, j_1) blocks this matching.

$\mathbf{a}_1$	$\mathbf{a}_2$	$\mathbf{a}_3$	$\mathbf{j}_1$	$\mathbf{j}_2$	$\mathbf{j}_3$
j_1	j_1	j_3	a_3	a_2	a_2
j_2	j_3	j_2	a_1	a_1	a_1
j_3	j_2	j_1	a_2	a_3	a_3

Pareto Comparability

- In fact, the above strategy profile S implements a matching which is *pareto incomparable* (from the perspective of the agents) to the agent proposing DAA matching in the induced two-sided matching problem.

- The agent-proposing DAA gives us the matching $M' = \{(a_1, j_1), (a_2, j_3), (a_3, j_2)\}$.

- Comparing the two matchings: $M = \{(a_1, j_2), (a_2, j_1), (a_3, j_3)\}$ and $M' = \{(a_1, j_1), (a_2, j_3), (a_3, j_2)\}$. Agent 1 is better off in M', while agents 2 and 3 are better off in M.

- Note that by result 3, matching M' can also be implemented in NE. (the strategy profile $S = \{p_2, p_1, p_1\}$ implements it)

Result 4: When is the Converse True?

We say that platforms are *partitioning the set of jobs* in a platform choice problem if $J_p \cap J_{p'} = \emptyset$ for all $p, p' \in P$.

Result 4: When is the Converse True?

We say that platforms are *partitioning the set of jobs* in a platform choice problem if $J_p \cap J_{p'} = \emptyset$ for all $p, p' \in P$.

Theorem

Fix a platform choice problem with the platforms partitioning the set of jobs. Then any Nash equilibrium in the platform choice game generates a stable matching for the induced two-sided matching problem.

This result is quite robust, any relaxation to the limits to the intersection of platforms and it goes away.

A Notion of Comparing different NE

Consider two Nash equilibria $(p_a)_{a \in A}$ (NE1) and $(p^*_a)_{a \in A}$ (NE2). Result 1 states that each agent is employed at each of these two Nash equilibria. For $j \in J$, let $a_j, a^*_j \in A$ be s.t. $\left(a_j, p_{a_j}, j\right) \in DAA((p_a)_{a \in A})$ and $\left(a^*_j, p_{a^*_j}, j\right) \in DAA((p_a)_{a \in A})$ for $p_{a_j}, p_{a^*_j} \in P_j$.

A Notion of Comparing different NE

Consider two Nash equilibria $(p_a)_{a \in A}$ (NE1) and $(p^*_a)_{a \in A}$ (NE2). Result 1 states that each agent is employed at each of these two Nash equilibria. For $j \in J$, let $a_j, a^*_j \in A$ be s.t. $\left(a_j, p_{a_j}, j\right) \in DAA((p_a)_{a \in A})$ and $\left(a^*_j, p_{a^*_j}, j\right) \in DAA((p_a)_{a \in A})$ for $p_{a_j}, p_{a^*_j} \in P_j$.

We say that NE1 is *better* than NE2 if for all jobs $j \in J$ we have that either $p_{a_j} \succ_j p_{a^*_j}$ or $p_{a_j} = p_{a^*_j}$.

This gives us a partial ranking of the set of NE, where the “best” NE are the ones where each job is matched to an agent who has chosen its favorite platform.

Properties of the Comparison of NE

- All stable matchings of the induced two-sided matching problem can be implemented by a *maximal* Nash equilibrium.

- In fact, any maximal NE will only implement a stable matching of the induced two-sided matching problem.

- We can have two NE where one is *better* than the other but both implement the same matching.

- We can have two NE, say NE1 and NE2 such that NE1 is *better* NE2, but NE2 implements a matching which is a Pareto improvement over the matching implemented by NE1 (from the agent's perspective).

Generalizing Aggregation

We look at a more generalized version of aggregations, including but not limited to lexicographic aggregation. We call it **platform monotonic aggregation**.

Definition

Denote by $r_p(a) \in \{1, ..., n\}$ the position agent $a \in A$ has in $\succ_p$. The induced aggregate ranking of job j, denoted by $>_j$ over A_j is *platform monotonic* if:

- For $p \in P_j$ and $a', a'' \in A_j^p : a' \succ_p a''$ implies $a' >_j a''$;
- For $p', p'' \in P_j$, $a' \in A_j^{p'}$ and $a'' \in A_j^{p''}$: $p' \succ_j p''$ and $r_{p'}(a') \leq r_{p''}(a'')$ implies $a' >_j a''$.

The aim is to prove results similar to those discussed for the generalized aggregation.

Result 1: Full Employment

Result 1 goes through seamlessly for the generalized aggregation.

Theorem

For every platform choice game where all jobs aggregate preferences in a platform-monotonic manner, all agents in a Nash equilibrium are employed.

Many-to-One Matching

- What if each job "lister" can offer multiple jobs and has preferences on the subset of agents who occupy those jobs?
- Agents now have preferences over job "listers" and not individual jobs.
- We aim to see how the platform choice game in a many-to-one matching setting generalizes.
- What are possible natural restrictions on the preferences of jobs?

Responsive-Monotonic Preferences

We define induced preferences of job listers differently. We say that job listers now have preferences over *specialized* subsets of agent-platform pairs.

Definition

We say that preferences of job listers are responsive-monotonic if:

- Preferences of the jobs over singleton pairs, for example, $\{(a_1, p_k)\}$ satisfies platform monotonicity.
- Preferences over sets S, S' such that $S = S' \cup \{(a, p)\} \setminus \{(a', p')\}$ for some $a \neq a'$, $a, a' \in A$ and $p, p' \in P$ is defined by: S is preferred to S' if and only if $\{(a, p)\}$ is preferred to $\{(a', p')\}$.

This definition is based on the definition of responsive preferences in Roth (1985).

What Next?

- We aim to generalize the platform choice game to the case of platform monotonic preferences as well as to the many-to-one matching problem.
- In the standard many-to-one matching, the DAA outputs a stable matching if the preferences are as loose as *substitutable* (responsive are special cases).
- We conjecture that a coherent theory of the platform choice game can be constructed for what would be the equivalent of substitutable preferences (Kelso and Crawford (1982)) in our model.

Thank You!

It's been a pleasure!