Deep Learning for Two-Sided Matchings - Summary

Aditya Kumar Shaw
ksaditya@iisc.ac.in

Arun Kumar
arunkumar12@iisc.ac.in

Prabhav Verma
prabhavverma@iisc.ac.in

Shishir Madhab Roy
shishirm@iisc.ac.in

Abstract

We investigate the integration of deep learning methods into the development of two-sided matching mechanisms, exploring the trade-off between strategy-proofness and stability. This study introduces differentiable surrogates to quantify ordinal strategy-proofness and stability, enabling neural networks to map preference profiles to valid matchings. Experimental results demonstrate precise control over the trade-off through parameter tuning, surpassing traditional approaches. By leveraging recommended papers and relevant literature, we examine the potential of machine learning pipelines in market design, aiming to replicate and enhance existing results.

1 Introduction

The *two-sided matching problem* involves pairing two sets of agents, each with a strict preference order over the other set, aiming to maximize overall stability. The applications include matching students and colleges, workers and firms, marriageable men and women and many more.

Stability in two-sided matching refers to a situation where there are no two agents, one from each side, who prefer to match with each other over their current matches. *Strategy Proofness (SP)* in two-sided matching ensures that no agent can manipulate their preferences to improve their outcome without worsening someone else's. *Individual Rationality (IR)* in two-sided matching ensures every agent is assigned a partner they prefer over being unmatched, preventing regrets for participating.

It is proved that stability and SP cannot be simultaneously achieved in two-sided matching Dubins and Freedman [1981], Roth [1982]. Thus, a trade-off must exist between stability and SP. However, this trade-off is poorly understood beyond the existing solutions and Ravindranath et al. [2023] successfully tried to get a better understanding of this trade-off using deep learning.

2 Related Work

Historically, the *automated mechanism design (AMD)* approach framed problems (including auctions & two-sided matching) as a linear program. However, this formulation suffers from exponential growth in complexity with the increase in number of agents and items.

Recent works, Dütting et al. [2022], Curry et al. [2020], Shen et al. [2021], Rahme et al. [2021], overcame this limitation by using deep neural networks for solving economic design problems, in particular they explored the *design of auctions* using deep learning frameworks.

Other related work includes using SVMs to search for stable matching without considering SP, and stable matching in context of bandid problems where agents' preferences are unknown a priori.

Preprint. Under review.

3 Existing Methodology

For notations, refer to the appendix A

3.1 Deferred Acceptance (DA)

Gale and Shapley [1962] introduced a simple matching mechanism for two-sided matching problem - DA - which is stable (for both sides) but not SP (for one side).

1. $\forall w \in W : w$ creates a list of its acceptable firms called remaining firms.
2. $\forall w \in W : w$ proposes to its most preferred acceptable remaining firm.
3. $\forall f \in F : f$ tentatively accepts its best proposal, if any, and rejects other proposals.
4. $\forall w \in W : w$ if rejected by firm f, removes f from its remaining firms list.
5. Repeat steps $[2, 3, 4, 5]$ until no changes occur in matchings.

This is the *worker-proposing DA algorithm*, firm-proposing DA algorithm can be defined analogously.

3.2 Random Serial Dictatorship (RSD)

Abdulkadiroglu et al. [2006] introduced another matching mechanism - RSD - which when used to solve two-sided matching problem is SP but not stable.

1. Sample a priority order, π[1], on the set $W \cup F$ uniformly at random.
2. For k = $\{1, ..., |\pi|\}$:
   (a) If π_k is not yet matched, then match π_k to its most preferred unmatched agent or to $\perp$ if all acceptable agents are already matched to some other agent.

3.3 Top trading cycles (TTC)

The TTC mechanism introduced in Shapley and Scarf [1974] when used to solve two-sided matching problem is neither SP (SP for one side) nor stable. Following is the *worker-proposing TTC algorithm*, firm-proposing TTC algorithm can be defined analogously.

1. $\forall a \in W \cup F : a$ creates a list of its acceptable agents called remaining agents.
2. Create a directed graph G with each unmatched agent a pointing to its most preferred agent. The agent a can point to itself if its remaining agent list is empty.
3. Every worker w that is part of a cycle in G is matched to a firm (or itself) it points to.
4. All unmatched agents remove from their remaining agents list the agents that got matched in this round.
5. Repeat steps $[2, 3, 4, 5]$ until all workers are matched.

3.4 Randomized Matching Mechanism

Randomized Matching Mechanism, g, maps preference profiles, $\succ$, to distributions on matching, $g(\succ) \in \Delta(\mathcal{B})$. $r \in [0,1]^{|\overline{W}|\times|\overline{F}|}$ is the marginal probability, r_{wf}, with which the worker w is matched with firm f, for each $w \in \overline{W}, f \in \overline{F}$.

Definition 1 (Ex ante justified envy) *A randomized matching r causes ex ante justified envy if -*

- *some worker w prefers f over some fractionally matched firm f' (including $f' = \perp$) and firm f prefers w over some fractionally matched worker w' (including $w' = \perp$).*
- *some worker w finds a fractionally matched $f' \in F$ unacceptable i.e. $r_{wf'} > 0, \perp \succ_w f'$ or some firm f finds a fractionally matched $w' \in W$ unacceptable i.e. $r_{w'f} > 0, \perp \succ_f w'$.*

[1] π is a permutation of the agents $a \in W \cup F$ denoting the priority of agents

Definition 2 (Ex ante stable) *A randomized matching, r, is ex ante stable if and only if it does not cause any ex ante justified envy.*

Definition 3 (Utility Function) *For worker w we define, $u_w : \overline{F} \rightarrow \Re$, a $\succ_w$-utility function where $u_w(f) > u_w(f') \iff f \succ_w f', \forall f, f' \in \overline{F}$. Analogously, we define $\succ_f$-utility function for firm f.*

Definition 4 (Ordinal SP) *A randomized matching mechanism, g, satisfies ordinal SP if and only if $\forall a \in W \cup F, \forall \succ \in P, \forall \succ_a$-utility function, and for all reports $\succ'_a$ we have -*

$$\mathbb{E}_{\mu \sim g(\succ_a, \succ_{-a})}[u_i(\mu(i))] \geq \mathbb{E}_{\mu \sim g(\succ'_a, \succ_{-a})}[u_i(\mu(i))] \tag{1}$$

Definition 5 (First order Stochastic Dominance (FOSD)) *A randomized matching mechanism, g, satisfies FOSD if and only if $\forall w \in W, \forall f' \in \overline{F}$ such that $f' \succ_w \perp, \forall \succ_{-w}$, we have -*

$$\sum_{f \in F : f \succ_w f'} g_{wf}(\succ_w, \succ_{-w}) \geq \sum_{f \in F : f \succ_w f'} g_{wf}(\succ'_w, \succ_{-w}) \tag{2}$$

4 Proposed Methodology

Ravindranath et al. [2023] explored randomized matching mechanisms, for which the strongest SP concept is Ordinal SP, equivalent to FOSD. They formulated the two-sided matching problem as a deep learning task. The Neural Network (NN) function, $g^\theta : P \rightarrow \Delta(\mathcal{B})$, parameterized by θ, maps preference profile $\succ$ to matching distribution r.

The paper represented the ordinal preference order as a vector with constant offset utility since NN can't process ordinal preferences directly. Furthermore, they identified suitable, differentiable surrogates for approximate SP and stability. The NN's final output, $r \in [0, 1]^{n \times m}$, represents marginal probabilities in a randomized matching for the input profile. Notably, $r_{wf} = 0$ only when the match is unacceptable and this is realized using a boolean mask β.

The *Stability Violation (STV())* acts as a differentiable surrogate for stability and it is given by -

$$STV(g^\theta) = \mathbb{E}_{\succ}[stv(g^\theta, \succ)]B \tag{3}$$

Theorem 1 *A randomized mechanism, g^θ, is ex ante stable up to zero-measure events if and only if $STV(g^\theta) = 0$.*

The *Ordinal SP Violation (RGT())* acts as a differentiable surrogate for ordinal SP and it is given by -

$$RGT(g^\theta) = \mathbb{E}_{\succ}[regret(g^\theta, \succ)]C \tag{4}$$

Theorem 2 *A randomized mechanism, g^θ, is ordinal SP up to zero-measure events if and only if $RGT(g^\theta) = 0$.*

Following is the *loss function* proposed, where $\lambda \in [0, 1]$ is the trade-off between stability and SP -

$$\min_\theta \lambda * STV(g^\theta) + (1 - \lambda) * RGT(g^\theta) \tag{5}$$

5 Experimental Results

For $\lambda = 0$, the learnt mechanism's performance closely resembles RSD with very low regret but poor stability. As λ increases, the learnt mechanism approximates DA. At intermediate λ values, it outperforms convex combinations of DA, RSD, and TTC.

6 Problem Statement

The objective of this project is to study, analyze and reproduce the results of the papers under review and try to implement and explore other Deep Learning architectures for the problem, assessing the trade-offs between strategy-proofness and stability they provide.

References

Atila Abdulkadiroglu, Parag Pathak, Alvin E Roth, and Tayfun Sonmez. Changing the boston school choice mechanism. Working Paper 11965, National Bureau of Economic Research, January 2006.

Michael J. Curry, Ping-Yeh Chiang, Tom Goldstein, and John Dickerson. Certifying strategyproof auction networks, 2020.

L E Dubins and D A Freedman. Machiavelli and the gale-shapley algorithm. *Am. Math. Mon.*, 88(7): 485, August 1981.

Paul Dütting, Zhe Feng, Harikrishna Narasimhan, David C. Parkes, and Sai Srivatsa Ravindranath. Optimal auctions through deep learning: Advances in differentiable economics, 2022.

D. Gale and L.S. Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962. doi: https://doi.org/10.2307/2312726.

Jad Rahme, Samy Jelassi, Joan Bruna, and S. Matthew Weinberg. A permutation-equivariant neural network architecture for auction design, 2021.

Sai Srivatsa Ravindranath, Zhe Feng, Shira Li, Jonathan Ma, Scott D. Kominers, and David C. Parkes. Deep learning for two-sided matching, 2023.

Alvin E Roth. The economics of matching: Stability and incentives. *Math. Oper. Res.*, 7(4):617–628, November 1982.

Lloyd Shapley and Herbert Scarf. On cores and indivisibility. *Journal of Mathematical Economics*, 1 (1):23–37, 1974. ISSN 0304-4068. doi: https://doi.org/10.1016/0304-4068(74)90033-0.

Weiran Shen, Pingzhong Tang, and Song Zuo. Automated mechanism design via neural networks, 2021.

A Notations

$W = \{w_i\}_{i=1}^n$ is the set of n workers and $F = \{f_j\}_{j=1}^m$ is the set of m firms.

(w_i, f_j) is a match i.e. worker w_i is matched with firm f_j and vice versa. Furthermore, if a worker or firm is unmatched, we say it is matched with $\perp$.

μ is a feasible matching, $\mu = \{(w_i, f_j)_{w_i \in \overline{W}, f_j \in \overline{F}}\}$, such that each worker and each firm occurs in a match exactly once and $\overline{W} = W \cup \{\perp\}, \overline{F} = F \cup \{\perp\}$.

If $(w_i, f_j) \in \mu$, then we write $\mu(w_i) = f_j, \mu(f_j) = w_i$.

$\mathcal{B} = \{\mu\}$ is the set of all possible matching.

Each worker (firm) has a strict preference order, $\succ_w$ $(\succ_f)$, over $\overline{W}(\overline{F})$.

If a worker (firm) prefers firm f over f' (worker w over w'), then we write $f \succ_w f'$ $(w \succ_f w')$.

Given a strict preference order, $\succ_w$ $(\succ_f)$, of worker w (firm m), all firms f (workers w) such that $f \succ_w \perp$ $(w \succ_f \perp)$ are called acceptable firms (workers).

$\succ = \{\succ_1, ..., \succ_n, \succ_{n+1}, ..., \succ_{n+m}\}$ is a preference profile consisting of the preference orders of n workers followed by the preference orders of m firms.

$P = \{\succ\}$ is the set of all preference profiles.

B Stability Violation (STV())

$stv(g^\theta, \succ)$ is given by -

$$stv(g^\theta, \succ) = \frac{1}{2}\left(\frac{1}{m} + \frac{1}{n}\right) \sum_{w=1}^{n} \sum_{f=1}^{m} stv_{wf}(g^\theta, \succ)$$

where $stv_{wf}(g^\theta, \succ)$ is given by -

$$stv_{wf}(g^\theta, \succ) = \left(\sum_{w' \in \overline{W}} g^\theta_{w'f} * max(q^\succ_{wf} - q^\succ_{w'f}, 0)\right) * \left(\sum_{f' \in \overline{F}} g^\theta_{wf'} * max(p^\succ_{wf} - p^\succ_{wf'}, 0)\right)$$

where $p^\succ_{wf}, p^\succ_{wf'}, q^\succ_{wf}, q^\succ_{w'f}$ are the input representations of the preference orders.

C Ordinal SP Violation (RGT())

$regret(g^\theta, \succ)$ is given by -

$$regret(g^\theta, \succ) = \frac{1}{2}\left(\frac{1}{m} * \sum_{w \in W} regret_w(g^\theta, \succ) + \frac{1}{n} * \sum_{f \in F} regret_f(g^\theta, \succ)\right)$$

where $regret_w(g^\theta, \succ)$ is given by -

$$regret_w(g^\theta, \succ) = \max_{\succ'_w \in P}\left(\max_{f' \succ_w \perp} \sum_{f \succ_w f'} \Delta_{wf}(g^\theta, \succ'_w, \succ)\right)$$

and $regret_f(g^\theta, \succ)$ is given by -

$$regret_f(g^\theta, \succ) = \max_{\succ'_f \in P}\left(\max_{w' \succ_f \perp} \sum_{w \succ_f w'} \Delta_{wf}(g^\theta, \succ'_f, \succ)\right)$$

where $\Delta_{wf}(g^\theta, \succ'_f, \succ)$ is given by -

$$\Delta_{wf}(g^\theta, \succ'_f, \succ) = g^\theta_{wf}(\succ'_w, \succ_{-w}) - g^\theta_{wf}(\succ_w, \succ_{-w})$$