Truthful and Stable One-sided Matching on Networks

Extended Abstract

Tianyi Yang
ShanghaiTech University
Shanghai, China
tianyi.yang12@gmail.com

Yuxiang Zhai
ShanghaiTech University
Shanghai, China
zhaiyx@shanghaitech.edu.cn

Dengji Zhao
ShanghaiTech University
Shanghai, China
dengji.zhao@gmail.com

Xinwei Song
ShanghaiTech University
Shanghai, China
songxw@shanghaitech.edu.cn

Miao Li
ShanghaiTech University
Shanghai, China
limiao@shanghaitech.edu.cn

ABSTRACT

Diffusion one-sided matching aims at incentivizing more participants to match so as to improve overall matching result. Existing works have tried to add constraints on Top Trading Cycles to obtain the incentive, but it only works in trees. In this paper, we first propose a mechanism named Swap With Neighbors (SWN), which can work in any graph structure and intuitively satisfy incentive compatibility and the tightest stability (first defined here) in the new setting. Then we find a natural improvement of SWN called Leave and Share which not only reaches the same properties as SWN but also provides an obvious efficiency difference.

KEYWORDS

Mechanism Design; Stability; Invitation Incentive

ACM Reference Format:
Tianyi Yang, Yuxiang Zhai, Dengji Zhao, Xinwei Song, and Miao Li. 2024. Truthful and Stable One-sided Matching on Networks: Extended Abstract. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 3 pages.

1 INTRODUCTION

Mechanism design over social networks incentivizes participants to invite their friends to form a larger market so that participants can receive better outcomes[2, 3, 6]. However, inviting more participants is not always beneficial to the inviters. In one-sided matching, the well-known Top Trading Cycle (TTC) mechanism gives a unique truthful, stable, and optimal solution [4, 5], but TTC failed to incentivize the participants to invite others because an invitee might compete with her inviters for the same match [1].

To design the incentive, Kawasaki et al. [1] provide strategy-proof solutions by restricting each agent's match domain and requiring the network to be trees. Our work will further improve the matching by relaxing the restrictions to general social networks and giving agents more matching choices. The challenge for designing a satisfying mechanism is that we cannot tell whether an agent's invitation will harm herself or those who invited her without fixing a matching mechanism. It seems that a natural way to find a better mechanism is through trial and error. However, this is not feasible in practice, since the space and description of the matching mechanism are exponential in the number of participants.

In this paper, we propose two mechanisms to incentivize agents to enlarge the game for a better result. The first mechanism is called Swap With Neighbors (SWN), which only allows each agent to match with her neighbors. To further improve the matching result, we design another mechanism called Leave and Share (LS), which allows matched agents to share their unmatched neighbors with others. By doing so, the later matched agents will have more choices, which improves their matching. Both SWN and LS incentivize invitation and give the most stable solution under networks.

2 THE MODEL

We consider a diffusion one-sided matching problem denoted by an undirected graph $G = (N, E)$, which contains n agents $N = \{1, \dots, n\}$. Each agent $i \in N$ is endowed with an indivisible item h_i and $H = \{h_1, \dots, h_n\}$ is the set of all items. We define agent i as j's neighbor if there is an edge $e \in E$ between agent i and j, and let $r_i \subseteq N$ be i's neighbor set. Each agent $i \in N$ has a strict preference $\succ_i$ over H, and we use $\succeq_i$ to represent the weak preference. Denote agent i's private type as $\theta_i = (\succ_i, r_i)$ and $\theta = (\theta_1, \cdots, \theta_n)$ as the type profile of all agents. Let θ_{-i} be the type profile of all agents except for agent i. Let Θ be the type profile space of all agents.

In a matching mechanism, each agent is required to report her type. We denote agent i's reported type as $\theta'_i = (\succ'_i, r'_i)$ (reporting neighbor set is treated as inviting neighbors in practice). For a given θ', we generate a directed graph $G(\theta') = (N(\theta'), E(\theta'))$, where edge $\langle i, j\rangle \in E(\theta')$ if and only if $j \in r'_i$. Different from the traditional setting, we assume only a subset of the agents $N_0 \subseteq N$ is initially in the game, and the others need the existing participants' invitation to join the game. Under θ', we say agent i is qualified if and only if there is a path (a chain of invitation) from an agent in N_0 to i in $G(\theta')$. Let $Q(\theta')$ be the qualified agent set under θ'.

Definition 2.1. **A diffusion one-sided matching mechanism** is an allocation policy $\pi = (\pi_i)_{i \in N}$, where $\pi_i : \Theta \to H$ satisfies:

(1) for all agents $i \in N$, $\pi_i(\theta') \in H$, and $\pi_i(\theta') \neq \pi_j(\theta')$ if $i \neq j$.
(2) for all qualified agents $i \in Q(\theta')$, $\pi_i(\theta')$ is independent of the reports of all unqualified agents.

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand. © 2024 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

(3) for all unqualified agents $i \notin Q(\theta')$, $\pi_i(\theta') = h_i$.

Next, we define properties for a desirable matching mechanism.

Definition 2.2. A diffusion one-sided matching mechanism π is individually rational (**IR**) if for all $i \in N$, all $\theta_i \in \Theta_i$, and all $\theta'_{-i} \in \Theta_{-i}$, we have $\pi_i(\theta_i, \theta'_{-i}) \succeq_i h_i$.

Definition 2.3. A diffusion one-sided matching mechanism π is incentive compatible (**IC**) if for all $i \in N$, all $\theta'_{-i} \in \Theta_{-i}$ and all $\theta_i, \theta'_i \in \Theta_i$, we have $\pi_i(\theta_i, \theta'_{-i}) \succeq_i \pi_i(\theta'_i, \theta'_{-i})$.

For stability, we define Stable-CC which requires the blocking coalition to be a complete component.

Definition 2.4. Given an allocation $\pi(\theta)$, we say a set of agents $S \subseteq N$ (with item set $H_S \subseteq H$) is a **blocking coalition under complete components** for $\pi(\theta)$ if S forms a complete component in $G(\theta)$ and there is an allocation $z(\theta)$ such that for all $i \in S, z_i(\theta) \in H_S$ and $z_i(\theta) \succeq_i \pi_i(\theta)$ with at least one $j \in S, z_j(\theta) \succ_j \pi_j(\theta)$.

Definition 2.5. We say a mechanism π is **stable under complete components (Stable-CC)** if for all type profiles θ, there is no blocking coalition under complete components for $\pi(\theta)$.

3 THE MECHANISMS

We first propose an intuitive mechanism named Swap With Neighbors, which only allows agents to swap with their neighbors.

Swap With Neighbors (SWN)

(1) Initialize $N_{out} = \emptyset$ and $R_i = r'_i \cup \{i\}$.
(2) For a given $G(\theta')$, do the below steps until $N_{out} = N$:
 (a) Let each agent $i \in (N \setminus N_{out})$ point to her favorite item among herself and her neighbors remaining in the matching $R_i \cap (N \setminus N_{out})$.
 (b) For each cycle C, allocate the item to the agent who points to it and add C to N_{out}.

In SWN, agents' allocation is determined by trading cycles, which makes it strategy-proof for the preference report. Since each agent can only get allocated a house from her neighbors, misreporting on one's neighbor set is not beneficial. SWN also satisfies Stable-CC because a trading cycle within neighbors can always be allocated.

Theorem 3.1. *SWN is IR, IC, and Stable-cc.*

However, SWN achieves IC and Stable-CC at the cost of efficiency, as it limits the matching options for agents. To combat this, we propose another mechanism called Leave and Share. Leave and Share uses SWN as a base and adds a natural sharing process to enlarge agents' selection space, trying to provide a better allocation.

We introduce two notations to simplify our description.

Definition 3.2. Given a set $A \subseteq N$, we say $f_i(A) = j \in A$ is i's favorite agent in A if for any agent $k \in A, h_j \succeq'_i h_k$.

Definition 3.3. An ordering of agents is a one-to-one function $\mathcal{P} : \mathbb{N}^+ \to N$, where agent $\mathcal{P}(i)$ is the i^{th} agent in the ordering. Agents in $\mathcal{P}$ are sorted in ascending order by the length of the shortest path from agent set N_0 to them. Especially, for any agent $i \in N_0$, its shortest path length is 0. When multiple agents have the same length of the shortest path, we use a random tie-breaking.

Leave and Share (LS)

(1) Initialize $N_{out} = \emptyset$ and an empty stack S. Define the top and bottom of S as S_{top} and S_{bottom} respectively, and let $R_i = r'_i \cup \{S_{bottom}, i\}$.
(2) While $N_{out} \neq N$:
 (a) Find the minimum t such that $\mathcal{P}(t) \notin N_{out}$. Push $\mathcal{P}(t)$ into S.
 (b) While S is not empty:
  (i) While $f_{S_{top}}(R_{S_{top}}) \notin S$, push $f_{S_{top}}(R_{S_{top}})$ into S.
  (ii) Pop off all agents from S_{top} to $f_{S_{top}}(R_{S_{top}})$, who already formed a trading cycle C following their favorite agents. Allocate each agent $i \in C$ the item $h_{f_i(R_i)}$. Add C to N^t_{out}.
  (iii) Update the neighbor set of C's remaining neighbors by removing C, i.e., for all $j \in \bigcup_{i \in C} r'_i \setminus N^t_{out}$, set $r'_j = r'_j \setminus C$.
 (c) Add N^t_{out} to N_{out}. Let all remaining neighbors of N^t_{out} connect with each other, i.e., they become neighbors of each other. That is, let $X = \bigcup_{i \in N^t_{out}} r'_i \setminus N^t_{out}$ and for all $j \in X$, set $r'_j = r'_j \cup X$.

Theorem 3.4. *For any ordering $\mathcal{P}$, LS is IR, IC, and Stable-CC.*

4 OPTIMALITY ANALYSIS

In this section, we run experiments in random graphs to show the performance of our mechanisms. We define $\succ_i (j)$ as the j^{th} favorite item of i. Assuming that h_i is $\succ_i (j)$ and $\pi_i(\theta)$ is $\succ_i (k)$, we define the ascension of i as $d_i = j - k$. The average ascension of all agents is defined as $D = \frac{\sum_{i \in N} d_i}{n}$. We use D to measure agents' satisfaction in a one-sided matching mechanism.

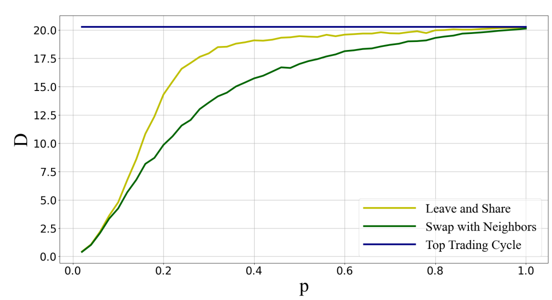

Figure 1: 100 graphs generated for each p to see how D changes accordingly. The minimum scale for p is 0.02.

To generate random networks, we define the probability of an edge between any two nodes as p. In Figure 1, we generate 100 graphs of 50 nodes with fixed but randomly generated preferences and adjust p to see how D changes. When p is close to 1, the performances of LS and SWN are close, and they are the same as TTC when $p = 1$. Due to the sharing process, LS converges to TTC faster than SWN. When p goes to 0, both LS and SWN have a poorer performance, because there are fewer neighbors to swap or share.

REFERENCES

[1] Takehiro Kawasaki, Ryoji Wada, Taiki Todo, and Makoto Yokoo. 2021. Mechanism Design for Housing Markets over Social Networks. In *AAMAS '21: 20th International Conference on Autonomous Agents and Multiagent Systems, Virtual Event, United Kingdom, May 3-7, 2021*. 692–700.

[2] Bin Li, Dong Hao, Hui Gao, and Dengji Zhao. 2022. Diffusion auction design. *Artificial Intelligence* 303 (2022), 103631.

[3] Bin Li, Dong Hao, Dengji Zhao, and Tao Zhou. 2017. Mechanism Design in Social Networks. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*. 586–592.

[4] Jinpeng Ma. 1994. Strategy-proofness and the strict core in a market with indivisibilities. *International Journal of Game Theory* 23, 1 (1994), 75–83.

[5] Lloyd Shapley and Herbert Scarf. 1974. On cores and indivisibility. *Journal of mathematical economics* 1, 1 (1974), 23–37.

[6] Yao Zhang and Dengji Zhao. 2022. Incentives to Invite Others to Form Larger Coalitions. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*. 1509–1517.