Improved Bandits in Many-to-one Matching Markets with Incentive Compatibility

Fang Kong, Shuai Li*

John Hopcroft Center for Computer Science, Shanghai Jiao Tong University
{fangkong, shuaili8}@sjtu.edu.cn

Abstract

Two-sided matching markets have been widely studied in the literature due to their rich applications. Since participants are usually uncertain about their preferences, online algorithms have recently been adopted to learn them through iterative interactions. **?** initiate the study of this problem in a many-to-one setting with *responsiveness*. However, their results are far from optimal and lack guarantees of incentive compatibility. An extension of **?** to this more general setting achieves a near-optimal bound for player-optimal regret. Nevertheless, due to the substantial requirement for collaboration, a single player's deviation could lead to a huge increase in its own cumulative rewards and an $O(T)$ regret for others. In this paper, we aim to enhance the regret bound in many-to-one markets while ensuring incentive compatibility. We first propose the adaptively explore-then-deferred-acceptance (AETDA) algorithm for responsiveness setting and derive an $O(N\min\{N,K\}C\log T/\Delta^2)$ upper bound for player-optimal stable regret while demonstrating its guarantee of incentive compatibility, where N represents the number of players, K is the number of arms, T denotes the time horizon, C is arms' total capacities and Δ signifies the minimum preference gap among players. This result is a significant improvement over **?**. And to the best of our knowledge, it constitutes the first player-optimal guarantee in matching markets that offers such robust assurances. We also consider broader *substitutable* preferences, one of the most general conditions to ensure the existence of a stable matching and cover responsiveness. We devise an online DA (ODA) algorithm and establish an $O(NK\log T/\Delta^2)$ player-pessimal stable regret bound for this setting. Compared with **?**, this algorithm not only achieves a better result but also applies to more general markets.

1 Introduction

The problem of two-sided matching markets has been studied for a long history due to its wide range of applications in real life including the labor market and college admission (**??????**). There are two sides of market participants, e.g., employers and workers in the labor market, and each side has a preference ranking over the other side. The matching reflects the bilateral nature of exchange in the market. For example, a worker works for an employer and the employer employs this worker. Stability is a key concept describing the equilibrium of a matching, which ensures the current bilateral exchange cannot be easily broken. A rich line of works study how to find a stable matching in the market (**?????**). However, all of them assume the preferences of market participants are known *a priori*, which may not be satisfied in practice. For example in labor markets, workers usually have unknown preferences over employers since they do not know whether they like the task type or the employer. With the emergence of online marketplaces such as online labor market Upwork and crowdsourcing platform Amazon Mechanical Turk where employers have numerous similar tasks to delegate, workers are able to learn the uncertain preferences during the iterative matching process with employers through these tasks.

Multi-armed bandit (MAB) is a core problem that characterizes the learning process during iterative interactions when faced with uncertainty (**?**). There are also two sides of agents: a player on one side and K arms on the other side. The player has unknown preferences over arms. At each time, it selects an arm and receives a reward. The player's objective is to maximize the cumulative reward over a specified horizon. To better measure the performance of the player's strategy, an equivalent objective of minimizing the cumulative regret is widely studied, which is defined as the cumulative difference between the reward of the optimal arm and that of the selected arms.

Recently, a rich line of works study the bandit learning problem in matching markets where more than one player and arms exist. These works study the case where players have unknown preferences over arms and arms can determine their preferences over players based on some known utilities such as the profile of workers in online labor markets. To characterize the stability of the learned matching, the objective of stable regret is adopted and studied (**????????**). Previous works mainly focus on two types of objectives: the player-optimal stable regret and the player-pessimal stable regret. The former is defined as the cumulative difference between the reward of the arm in the players' most preferred stable matching and the accumulated reward by the player. The latter is defined compared with the reward of the arm in the players' least preferred stable matching. **?** first study the centralized version where a central platform assigns an allocation of arms to players in each round and provide theoretical guarantees. Since such a platform may not always

*Corresponding author.

exist in real applications, the following works mainly focus on the decentralized setting where each player makes her own decision (**???????**). Most of these works achieve guarantees on the player-pessimal stable regret and until recently, **?** and **?** independently propose algorithms that can reach player-optimal stable matching.

All of the above works study the one-to-one matching markets where each player proposes to one arm at a time and each arm could accept at most one player. The many-to-one setting is more general and common in real life such as in labor markets where an employer usually has a certain quota and can recruit a group of workers (**????**). **?** initialize the study in many-to-one markets by considering that arms have responsive preferences. However, their algorithm is only able to achieve player-pessimal stable matching and lacks guarantees on incentive compatibility. Incentive compatibility is a crucial property in multi-player systems as it ensures players are incentivized to act in ways that align with desired system outcomes, thereby promoting cooperation and efficiency rather than encouraging competitive or destructive behaviors. Deriving algorithms that can achieve better regret and enjoy guarantees on this property is important in matching markets.

In this paper, we aim to provide algorithms with improved regret guarantee and incentive compatibility for many-to-one markets. For the sake of the generality, we also study the decentralized setting. We propose an adaptive explore-then-DA (AETDA) algorithm for markets with responsive preferences and derive $O(N \min\{N, K\} C \log T/\Delta^2)$ upper bound for the player-optimal stable regret as well as a guarantee of incentive compatibility, where N is the number of players, K is the number of arms, C is arms' total capacities, T is the horizon, and Δ is the players' minimum preference gap. To the best of our knowledge, it is the first guarantee for the player-optimal regret in decentralized many-to-one markets and is also the first that simultaneously enjoys such robust assurance in one-to-one markets. Since arms preferences may possess a combinatorial structure which may not be well characterized by responsiveness, we also consider a more general setting with *substitutability* (**?**), one of the most generally known conditions to ensure the existence of a stable matching and naturally holds under responsiveness (**??**). We design an online deferred acceptance (ODA) algorithm for this more general setting and prove that the regret against the player-pessimal stable matching is bounded by $O(NK \log T/\Delta^2)$. As compared in Table 1, this result not only works under a more general setting but also achieves a great advantage over **?**.

2 Related Work

The matching market model characterizes many applications such as labor market (**?**), house allocation (**?**), college admission and marriage problems (**?**), among which the many-to-one setting is very common and widely studied (**?**). Responsiveness and substitutability are most generally known conditions to guarantee the existence of a stable matching (**????**) and the deferred acceptance (DA) algorithm is a classical offline algorithm to find a stable matching under this property (**??**).

For simplicity, we refer to the setting where one-side participants (players) have unknown preferences as the online setting. This line of works relies on the technique of MAB, a classical online learning framework with a single player and K arms (**?**). The explore-then-commit (ETC) (**?**), upper confidence bound (UCB) (**?**), Thompson sampling (TS) (**?**) and elimination (**?**) algorithms are common strategies to obtain $O(K \log T/\Delta)$ regret where Δ is the minimum suboptimality gap among arms.

Multiple-player MAB (MP-MAB) generalizes the standard MAB problem by considering more than one player in the environment. In this setting, each player selects an arm at each time and a player will receive nothing if it collides with others by selecting the same arm. The MP-MAB problem has been studied in both homogeneous and heterogeneous settings. The former assumes different players share the same preference over arms (**??**) and the latter allows players to have different preferences (**??**). Both settings aim to minimize the collective cumulative regret of all players.

The matching market problem is different from above MP-MAB framework by considering that each arm also has a preference ranking over players. When multiple players select one arm, the player preferred most by the arm would not be collided and would gain a reward. The objective in this setting is to learn a stable matching and minimize the stable regret for players. **?** first introduce the bandit learning problem in one-to-one matching markets and explore the empirical performances of the proposed algorithms in the market where all participants on each side have the same preferences. Recently, **?** study a variant of the problem and present the first theoretical guarantees in a centralized setting where a central platform assigns allocations to players in each round. Later, **?**, **?** and **?** successively study this setting in a decentralized manner where players make their own decisions without a central platform. These works additionally assume the preferences of participants satisfy some constraints to ensure the uniqueness of the stable matching. For a general decentralized one-to-one market, **?** and **?** propose UCB and TS-type algorithms with guarantees for player-pessimal stable regret, respectively. Until recently, the theoretical analysis for the stronger player-optimal stable regret objective has been derived (**??**).

Due to the generality when modeling real applications, **?** start to study the bandit problem in many-to-one settings. They assume arms have responsive preferences and derive algorithms both in centralized and decentralized settings. For the decentralized setting, they only guarantee the player-pessimal stable regret with the upper bound $O(N^5 K^2 \log^2 T/(\varepsilon^{N^4} \Delta^2))$ where $\varepsilon \in (0, 1)$ is a hyper-parameter. Please see Table 1 for a comprehensive comparison among these works.

3 Setting

The two-sided market consists of N players and K arms. Denote the player and the arm set as $\mathcal{N} = \{p_1, p_2, \ldots, p_N\}$ and $\mathcal{K} = \{a_1, a_2, \ldots, a_K\}$, respectively. Just as in common applications such as the online labor market, players have preferences over individual arms. The relative preference of

Table 1: Comparisons of settings and regret bounds with most related works. $*$ represents the player-optimal stable regret and bounds without labeling $*$ are for player-pessimal stable regret, $\#$ represents the centralized setting. $N, K, \Delta, C, \varepsilon, C'$ are the number of players and arms, the minimum preference gap among all players, the total capacities of all arms under responsiveness, the hyper-parameter of algorithms which can be very small, and the parameter related to the unique stable matching condition which can grow exponentially in N, respectively. 'Incentive' means that there is a guarantee for incentive compatibility.

	Regret bound	Setting
?	$O\left(K\log T/\Delta^2\right) * \#$ $O\left(NK\log T/\Delta^2\right) \#$	one-to-one, known Δ, incentive one-to-one, incentive
?	$O\left(\frac{N^5K^2\log^2 T}{\varepsilon^{N^4}\Delta^2}\right)$	one-to-one
?	$O\left(NK\log T/\Delta^2\right)$ $\Omega\left(N\log T/\Delta^2\right)$	one-to-one (serial dictatorship), incentive
?	$O\left(K\log^{1+\varepsilon} T + 2^{\left(\frac{1}{\Delta^2}\right)^{\frac{1}{\varepsilon}}}\right) *$ $O\left(NK\log T/\Delta^2\right)$	one-to-one one-to-one (uniqueness consistency)
?	$O\left(C'NK\log T/\Delta^2\right)$	one-to-one (α-reducible condition)
?	$O\left(\frac{N^5K^2\log^2 T}{\varepsilon^{N^4}\Delta^2}\right)$	one-to-one
?	$O\left(K\log T/\Delta^2\right) *$	one-to-one
?	$O\left(K\log T/\Delta^2\right) *$	one-to-one responsiveness **(our extension)**
?	$O\left(K\log T/\Delta^2\right) * \#$ $O\left(NK^3\log T/\Delta^2\right) \#$ $O\left(\frac{N^5K^2\log^2 T}{\varepsilon^{N^4}\Delta^2}\right)$	responsiveness, known Δ responsiveness responsiveness
Ours	$O\left(N\min\{N,K\}C\log T/\Delta^2\right) *$ $O\left(NK\log T/\Delta^2\right)$	responsiveness, incentive substitutability, incentive

player p_i for arm a_j can be quantified by a real value $\mu_{i,j} \in (0,1]$, which is unknown and needs to be learned during interactions with arms. For each player p_i, we assume $\mu_{i,j} \neq \mu_{i,j'}$ for distinct arms $a_j, a_{j'}$ as in previous works (**??????**). And $\mu_{i,j} > \mu_{i,j'}$ implies that player p_i prefers a_j to $a_{j'}$. For the other side of participants, arms are usually certain of their preferences for players based on some known utilities, e.g., the profiles of workers in the online labor markets scenario (**?????**). In many-to-one markets, when faced with a set $P \subseteq \mathcal{N}$ of players, the arm can determine which subset of P it prefers most. Denote $\mathtt{Ch}_j(P)$ as this choice of arm j when faced with P. Then for any $P' \subseteq P$, arm a_j prefers $\mathtt{Ch}_j(P)$ to P'.

At each round $t = 1, 2, \ldots$, each player $p_i \in \mathcal{N}$ proposes to an arm $A_i(t) \in \mathcal{K}$. Let $A_j^{-1}(t) = \{p_i : A_i(t) = a_j\}$ be the set of players who propose to a_j. When faced with the player set $A_j^{-1}(t)$, arm a_j only accepts its most preferred subset $\mathtt{Ch}_j(A_j^{-1}(t))$ and would reject others. Once p_i is successfully accepted by arm $A_i(t)$, it receives a utility gain $X_{i,A_i(t)}(t)$, which is a 1-subgaussian random variable with expectation $\mu_{i,A_i(t)}$. Otherwise, it receives $X_{i,A_i(t)}(t) = 0$. We further denote $\bar{A}_i(t)$ as p_i's matched arm at round t. Specifically, $\bar{A}_i(t) = A_i(t)$ if p_i is successfully matched and $\bar{A}_i(t) = \emptyset$ otherwise. Inspired by real applications such as labor market where workers usually update their working experience on their profiles, we also assume each player can observe the successfully matched players $\mathtt{Ch}_j(A_j^{-1}(t)) = \bar{A}_j^{-1}(t) = \left\{p_i : \bar{A}_i(t) = a_j\right\}$ with each arm $a_j \in \mathcal{K}$ as previous works (**?????**).

The matching $\bar{A}(t)$ at round t is the set of all pairs $(p_i, \bar{A}_i(t))$. Stability of matchings is a key concept that describes the state in which any player or arm has no incentive to abandon the current partner (**??**). Formally, a matching is stable if it cannot be improved by any arm or player-arm pair. Specifically, an arm a_j improves $\bar{A}(t)$ if $\mathtt{Ch}_j(\bar{A}_j^{-1}(t)) \neq \bar{A}_j^{-1}(t)$. That's to say, arm a_j would not accept all members in $\bar{A}_j^{-1}(t)$ when faced with this set. A pair (p_i, a_j) improves the matching $\bar{A}(t)$ if p_i prefers a_j to $\bar{A}_i(t)$ and a_j would accept p_i when faced with $\bar{A}_j^{-1}(t) \cup \{p_i\}$, i.e., $p_i \in \mathtt{Ch}_j(\bar{A}_j^{-1}(t) \cup \{p_i\})$. That's to say, p_i prefers arm a_j than its current partner and a_j would also accept p_i if p_i apply for a_j together with a_j's current partners (**???**).

Responsive preferences are widely studied in many-to-one markets which guarantee the existence of a stable matching (**??**). Under this setting, each arm a_j has a preference ranking over individual players and a capacity $C_j > 0$. When a set of players propose to a_j, it accepts C_j of them with the highest preference ranking. This case recovers the one-to-one matching when $C_j = 1$. For convenience, define $C = \sum_{j \in [K]} C_j$ as the total capacities of all arms. Apart from responsiveness, we also consider a more general substitutability setting in Section 6.

In this paper, we study the bandit problem in many-to-one matching markets with responsive and substitutable preferences. Under both properties, the set M^* of stable matchings between $\mathcal{N}$ and $\mathcal{K}$ is non-empty (**??**). For each player p_i, let $\overline{m}_i \in [K]$ and $\underline{m}_i \in [K]$ be the index of p_i's most and least favorite arm among all arms that can be matched with p_i in a stable matching, respectively. The objective of each player p_i is to minimize the cumulative stable regret defined as the cumulative difference between the reward of the stable arm and that the player receives during the horizon. The player-optimal and pessimal stable regret are defined as

$$\overline{R}_i(T) = \mathbb{E}\left[\sum_{t=1}^{T} \mu_{i,\overline{m}_i} - X_{i,A_i(t)}(t)\right] , \tag{1}$$

$$\underline{R}_i(T) = \mathbb{E}\left[\sum_{t=1}^{T} \mu_{i,\underline{m}_i} - X_{i,A_i(t)}(t)\right] , \tag{2}$$

respectively (**??????**). The expectation is taken over by the randomness in reward gains and the players' policies.

For convenience, we define the corresponding gaps to measure the hardness of the problem.

Definition 1. *For each player p_i and arm $a_j \neq a_{j'}$, define $\Delta_{i,j,j'} = |\mu_{i,j} - \mu_{i,j'}|$ as the preference gap of p_i between a_j and $a_{j'}$. Let $\Delta = \min_{i,j,j':j \neq j'} \Delta_{i,j,j'}$ be the minimum preference gap among all players and arms, which is non-zero since players have distinct preferences.*

4 An Extension of ?

Recall that **?** provide a near-optimal bound $O(K \log T/\Delta^2)$ for player-optimal stable regret in one-to-one markets. We first provide an extension of their algorithm, explore-then-deferred-acceptance (ETDA), for many-to-one markets with responsiveness and $N \leq K \cdot \min_{j \in [K]} C_j$.

The deferred acceptance (DA) algorithm is designed to find a stable matching when both sides of participants have known preferences. The algorithm proceeds in multiple steps. At the first step, all players propose to their most preferred arm and each arm rejects all but their favorite subset of players among those who propose to it. Such a process continues until no rejection happens. It has been shown that the final matching is the player-optimal stable matching under responsiveness (**??**).

Since players are uncertain about their preferences, the ETDA algorithm lets players first explore to learn this knowledge and then follow DA to find a stable matching. Specifically, each player first estimates an index in the first N rounds (phase 1); and then explores its unknown preferences in a round-robin way based on its index (phase 2). After estimating a good preference ranking, it will follow DA to find the player-optimal stable matching (phase 3). Compared with **?**, the difference mainly lies in the first phase of estimating indices for players where multiple players can share the same index in many-to-one markets. For completeness, we provide the detailed algorithm in Appendix A and the theoretical guarantees below.

Theorem 1. *Following ETDA, the player-optimal stable regret of each player p_i satisfies*

$$\overline{R}_i(T) \leq O\left(K \log T/\Delta^2\right) . \tag{3}$$

Due to the space limit, the proof of Theorem 1 is deferred to Appendix A.2. Under the same decentralized setting, this player-optimal stable regret bound is even $O(N^5 K \log T/\varepsilon^{N^4})$ better than the weaker player-pessimal stable regret bound in **?**. Such a result also achieves the same order as the state-of-the-art analysis in the reduced one-to-one setting (**?**).

Though achieving better regret bound, the ETDA algorithm is not incentive compatible. We can consider the market where the player-optimal stable arm of a player p_i is its least preferred arm. If p_i always reports that it does not estimate the preference ranking well, then the stopping condition of phase 2 is never satisfied. In this case, all of the other players fail to find a stable matching and suffer $O(T)$ regret, while this player is always matched with more preferred arms than that in the stable matching during phase 2, resulting in $O(T)$ improvement in the cumulative rewards. Thus player p_i lacks the incentive to always act as the algorithm requires. To improve the algorithm in terms of incentive compatibility, we further propose a novel algorithm in the next section.

5 Adaptively ETDA (AETDA) Algorithm

In this section, we propose a new algorithm adaptively ETDA (AETDA) for many-to-one markets with responsive preferences which is incentive compatible. To ensure each player has a chance to be matched, we simply assume $N \leq C$ as existing works in many-to-one and one-to-one markets (**?????**), which relaxes the requirement of ETDA in the previous section.

For simplicity, we present the main algorithm in a centralized manner in Algorithm 1, i.e., a central platform coordinates players' selections in each round. The discussion on how to extend it to a decentralized setting is provided later.

Intuitively, AETDA integrates the learning process into each step of DA instead of estimating the full preference ranking before following DA like the ETDA algorithm. More specifically, each player explores arms in a round-robin manner in each step to learn its most preferred arm and then focuses on this arm before being rejected in the corresponding step of DA. For each player p_i, the algorithm maintains S_i to represent the available arm set that has not rejected p_i in previous steps and E_i to represent the exploration status. Specifically, $E_i = \text{True}$ means that p_i still needs to explore arms in a round-robin manner to find its most preferred arm in S_i, and $E_i = \text{False}$ means that p_i now focuses on its most preferred available arm. At the beginning of the algorithm,

Algorithm 1: centralized adaptively explore-then-deferred-acceptance (AETDA, from the view of the central platform)

1: Initialize: $S_i = \mathcal{K}, E_i = \text{True}$ for each player $p_i \in \mathcal{N}$
2: **for** round $t = 1, 2, ...,$ **do**
3:    Allocate $A_i(t) \in S_i$ to each player p_i with $E_i = \text{True}$ in a round-robin manner; Allocate $A_i(t) = \text{opt}_i$ to each player p_i with $E_i = \text{False}$
4:    Receive the estimation status opt_i from each p_i
5:    **for** each player $p_i \in \mathcal{N}$ with $\text{opt}_i \neq -1$ **do**
6:       $E_i = \text{False}$
7:    **end for**
8:    **for** each player $p_i \in \mathcal{N}$ and $a_j \in S_i$ with $p_i \notin \texttt{Ch}_j(\{p_{i'} : \text{opt}_{i'} = a_j\} \cup \{p_i\})$ **do**
9:       $S_i = S_i \setminus \{a_j\}$
10:       **if** $E_i = \text{False}$ and $a_j = \text{opt}_i$ **then**
11:          $E_i = \text{True}$
12:       **end if**
13:    **end for**
14: **end for**

S_i is initialized as the full arm set $\mathcal{K}$ and E_i is initialized as True (Line 1).

For players with $E_i = \text{True}$, the central platform would allocate the arm $A_i(t) \in S_i$ in a round-robin manner. And for those players with $E_i = \text{False}$, they can just focus on the determined optimal arm opt_i (Line 3). After being matched in each round, each player p_i would update its empirical mean $\hat{\mu}_{i,A_i(t)}$ and the number of observed times $T_{i,A_i(t)}$ on arm $A_i(t)$ as $\hat{\mu}_{i,A_i(t)} = (\hat{\mu}_{i,A_i(t)} \cdot T_{i,A_i(t)} + X_{i,A_i(t)}(t))/(T_{i,A_i(t)} + 1)$, $T_{i,A_i(t)} = T_{i,A_i(t)} + 1$. For the preference value $\mu_{i,j}$ towards each arm a_j, p_i also maintains a confidence interval at t with the upper bound $\text{UCB}_{i,j} := \hat{\mu}_{i,j} + \sqrt{6 \log T / T_{i,j}}$ and lower bound $\text{LCB}_{i,j} := \hat{\mu}_{i,j} - \sqrt{6 \log T / T_{i,j}}$. If $T_{i,j} = 0$, $\text{UCB}_{i,j}$ and $\text{LCB}_{i,j}$ are set as ∞ and $-\infty$, respectively. When the UCB of a_j is even lower than the LCB of other available arms, a_j is considered to be less preferred. Based on the estimations, p_i needs to determine whether an arm can be considered as optimal in S_i and submit this status to the platform (Line 4). Specifically, if there exists an arm $a_j \in S_i$ such that $\text{LCB}_{i,j} > \max_{a_{j'} \in S_i \setminus \{a_j\}} \text{UCB}_{i,j'}$, then a_j is regarded as optimal and player p_i would submit $\text{opt}_i = a_j$ to the platform. Otherwise, no arm can be regarded as optimal, and p_i would submit $\text{opt}_i = -1$. For players who have learned their most preferred arm, the platform would mark their exploration status as False (Line 6).

To avoid conflict when players with $E_i = \text{True}$ explore arms in a round-robin manner, we introduce a detection procedure to detect whether an arm in S_i is occupied by its more preferred players (Line 8-13). Specifically, if an arm a_j does not accept player p_i when faced with the player set who regards a_j as the optimal one (Line 8), then p_i can be regarded to be rejected by a_j when exploring this arm. In this case, no matter whether this arm is the most preferred one, p_i has no chance of being matched with it. So p_i directly deletes a_j from its available arm set S_i (Line 9). And if this arm is just the estimated optimal arm of p_i, then this case is equivalent in offline DA to that p_i is rejected when proposing to its most preferred arm (Line 10). In this case, p_i needs to explore to learn its next preferred arm and update E_i as True (Line 11).

For the arrangement of round-robin exploration, without loss of generality, we can convert the original set of K arms with total capacity C into a set of C new arms, each with a capacity 1. When N players explore these C new arms: the platform let p_1 follow the ordering $1, 2, ..., C-1, C, 1, ...$; p_2 follow $2, 3, ..., C, 1, 2, ...$; and so on. If an arm a_j is unavailable for a player p_i, p_i simply forgo the opportunity to select in the corresponding rounds. This pre-arranged ordering ensures that, in the worst case, each player can match with each available new arm, and so as to the available original arm, at least once in every C rounds.

Extension to the decentralized setting. In the decentralized setting without a central platform, each player maintains and updates their own S_i and E_i. We can define a phase version of Algorithm 1. Specifically, each phase contains a number of rounds and the size of phases grows exponentially, i.e., $2, 2^2, 2^3, \cdots$. Within each phase, each player p_i would explore arms in S_i in a round-robin manner if $E_i = \text{True}$ as discussed above and focus on arm opt_i otherwise. Players only update the status of opt_i (Line 4), E_i (Line 6), and S_i (Line 8-13) at the end of the phase based on the communication with other players and arms. If L observations on arms are enough to learn the optimal one in the centralized version, then the stopping condition (Line 4) would be satisfied at the end of the phase guaranteeing the number of observations in this decentralized version and the total number of selecting times would be at most $2L$ due to the exponentially increasing phase length. So the regret in this decentralized version is at most two times as that suffered in the centralized version. And the number of communications is at most $O(\log T)$ which is of the same order as the ETDA algorithm and also **?** for the one-to-one setting.

5.1 Theoretical Analysis

Algorithm 1 presents a new perspective that integrates the learning process into each step of the DA algorithm to find a player-optimal stable matching, which is more adaptive compared with existing explore-then-DA strategy (**??**). In the following, we will show that such a design simultaneously enjoys guarantees of player-optimal stable regret and incentive compatibility.

Theorem 2. *Following Algorithm 1, the player-optimal stable regret of each player p_i satisfies*

$$\overline{R}_i(T) \leq O\left(N \cdot \min\{N, K\} C \log T / \Delta^2\right) .$$

The following theorem further discusses the incentive compatibility of Algorithm 1.

Theorem 3. *(Incentive Compatibility) Given that all of the other players follow Algorithm 1, no single player p_i can improve its final matched arm by misreporting its opt_i in some rounds.*

Compared with **?**, our result not only achieves an $O(N^4 K \log T / (C \varepsilon^{N^4}))$ improvement over their weaker player-pessimal stable regret objective but also enjoys guarantees of incentive compatibility. Compared with the state-of-the-art result in one-to-one settings, our algorithm is more robust to players' deviation only with the cost of $O(NC)$ worse regret bound (**??**). To the best of our knowledge, it is the first algorithm that simultaneously achieves guarantees of polynomial player-optimal stable regret and incentive compatibility in both many-to-one markets and previously widely studied one-to-one markets without knowing the value of Δ.

Due to the space limit, the proofs of two theorems are deferred to Appendix B.

6 Online DA Algorithm for Substitutability

In many-to-one markets, arms may have combinatorial preferences over groups of players, which may not be well characterized by responsiveness. In this setting, we consider the markets with substitutability, which is one of the most common and general conditions that ensure the existence of a stable matching and is defined below.

Definition 2. *(Substitutability) The preference of arm a_j satisfy substitutability if for any player set $P \subseteq \mathcal{N}$ that contains p_i and $p_{i'}$, $p_i \in \mathtt{Ch}_j(P \setminus \{p_{i'}\})$when $p_i \in \mathtt{Ch}_j(P)$.*

The above property states that arm a_j keeps accepting player p_i when other players become unavailable. This is the sense that a_j regards players in a team as substitutes rather than complementary individuals (in which case the arm may give up accepting the player when others become unavailable). Such a phenomenon appears in many real applications and covers responsiveness as proved below.

Remark 1. *Select a player set $P \subseteq \mathcal{N}$ which contains p_i and $p_{i'}$. Suppose $p_i \in \mathtt{Ch}_j(P)$, i.e., p_i is one of the C_j highest-ranked players in P. Then when the available set becomes $P \setminus \{p_{i'}\}$, p_i is still one of the C_j highest-ranked players, i.e., $p_i \in \mathtt{Ch}_j(P \setminus \{p_{i'}\})$.*

The substitutability property is more general than responsiveness as arms' preferences can have combinatorial structures. The following is an example that satisfies substitutability but not responsiveness (**?**).

Example 1. *There are 3 players and 2 arms, i.e., $\mathcal{N} = \{p_1, p_2, p_3\}, \mathcal{K} = \{a_1, a_2\}$. The arms' preference rankings over subsets of players are*

- $a_1 : \{p_1, p_2\}, \{p_1, p_3\}, \{p_2, p_3\}, \{p_3\}, \{p_2\}, \{p_1\}$.
- $a_2 : \{p_3\}, \emptyset$.

That is to say, $\mathtt{Ch}_j(P)$ is the subset that ranks highest among all subsets listed above that only contain players in P. Taking the preferences of a_2 as an example, when $p_3 \in P$, then $\mathtt{Ch}_j(P) = \{p_3\}$; otherwise, $\mathtt{Ch}_j(P) = \emptyset$.

For many-to-one markets with substitutable preferences, we propose an online deferred acceptance (ODA) algorithm (presented in Algorithm 2). ODA is inspired by the idea of the DA algorithm with the arm side proposing, which finds a player-pessimal stable matching when players know their preferences. Specifically, the DA algorithm with the arm proposing proceeds in several steps. In the first step, each arm proposes to its most preferred subset among all players. Each player would reject all but the most preferred arm among those who propose it. In the following each step, each arm still proposes to its most preferred subset of players among those who have not rejected it and each player rejects all but the most preferred one among those who propose to it. This process stops when no rejection happens and the final matching is the player-pessimal stable matching (**??**).

Algorithm 2: online deferred acceptance (from view of p_i)

1: Input: player set $\mathcal{N}$, arm set $\mathcal{K}$
2: Initialize: $P_{i,j} = \mathcal{N}, \hat{\mu}_{i,j} = 0, T_{i,j} = 0$ for each $j \in [K]$; $S_i(1) = \{a_j \in \mathcal{K} : p_i \in \mathtt{Ch}_j(P_{i,j})\}$
3: **for** each round $t = 1, 2, \cdots$ **do**
4: Select $A_i(t) \in S_i(t)$ in a round-robin way
5: Update $\hat{\mu}_{i,\bar{A}_i(t)}$ and $T_{i,\bar{A}_i(t)}$ if $\bar{A}_i(t) = A_i(t) \neq \emptyset$
6: $S_i(t+1) = S_i(t)$
7: **for** $a_j \in S_i(t)$ and $\mathrm{UCB}_{i,j}(t) < \max_{a_{j'} \in S_i(t)} \mathrm{LCB}_{i,j'}(t)$ **do**
8: $S_i(t+1) = S_i(t+1) \setminus \{a_j\}$
9: **end for**
10: **if** $t \geq 2$ and $\forall p_{i'} \in \mathcal{N} : \bar{A}_{i'}(t) = \bar{A}_{i'}(t-1)$ **then**
11: $\forall j \in [K], P_{i,j} = P_{i,j} \setminus \{p_{i'} : \bar{A}_{i'}(t) \neq j, \exists t' < t-1 \text{ s.t. } \bar{A}_{i'}(t') = j\}$
12: $S_i(t+1) = \{a_j : p_i \in \mathtt{Ch}_j(P_{i,j})\}$
13: **end if**
14: **end for**

The ODA algorithm is designed with the guidance of this procedure but players decide which arm to select in each round. Specifically, each player p_i needs to record the available player set $P_{i,j}$ for each arm a_j, which consists of players who have not rejected arm a_j and is initialized as the full player set $\mathcal{N}$. Then if a player p_i is in the choice set of a_j when the set $P_{i,j}$ of players is available, i.e., $p_i \in \mathtt{Ch}_j(P_{i,j})$, p_i would be accepted if it proposes to a_j together with other players in $P_{i,j}$. The main purpose of the algorithm is to let players wait for this opportunity to choose arms that will successfully accept them.

Each player p_i can further construct the plausible set S_i to contain those arms that may successfully accept it, i.e., $S_i = \{a_j : p_i \in \mathtt{Ch}_j(P_{i,j})\}$. Here for simplicity, we additionally assume each player p_i knows whether $p_i \in \mathtt{Ch}_j(P)$ for each possible $P \subseteq \mathcal{N}$. This assumption is only used for clean analysis and the algorithm can also be generalized to the case where this information is unavailable by letting players in $P_{i,j}$ pull a_j and observe whether it is accepted. Since arms know their own preferences and conflicts are deterministically resolved, at most 2^N rounds are needed to obtain this information. Apart from $P_{i,j}$ and S_i, each player p_i also maintains $\hat{\mu}_{i,j}$ and $T_{i,j}$ to record the estimated value for $\mu_{i,j}$ and the number of its observations. At the beginning, both values are initialized to 0.

In each round t, each player p_i proposes to the arm a_j in the plausible set $S_i(t)$ in a round-robin way (Line 4). If they are successfully matched with each other (Line 5), p_i would update the corresponding $\hat{\mu}_{i,j}, T_{i,j}$ as Section 5. When the UCB of a_j is even lower than the LCB of other plausible arms, a_j is considered to be less preferred. In this case, the final stable arm of player p_i must be more preferred than a_j

and thus there is no need to select a_j anymore (Line 8).

Recall that the plausible sets of players are constructed based on the available sets for arms. To ensure each player successfully be accepted by arms in their own plausible set, all players need to keep the available sets for arms updated in sync. With the awareness that players always select plausible arms in a round-robin way, once p_i observes that all players focus on the same arm in the recent two rounds, it believes all players have determined the most preferred one. In this way, p_i would update the available set $P_{i,j}$ for each arm a_j by deleting players who do not consider a_j as stable arms anymore (Line 11). Since all players have the same observations, the update times of $P_{i,j}$ would be the same. Such a stage in which all players determine the most preferred arm in the plausible set can just be regarded as a step of the offline DA algorithm (with the arm side proposing) where each player rejects all but the most preferred one among those who propose to it. Thus the update times of $P_{i,j}$ just divide the total horizon into several stages with each corresponding to a step of DA.

6.1 Theoretical Analysis

We first provide the regret bound for Algorithm 2.

Theorem 4. *Following Algorithm 2, the player-pessimal stable regret of each player p_i satisfies*

$$\underline{R}_i(T) \leq O(NK \log T/\Delta^2)\,. \tag{4}$$

Apart from the regret guarantee, we also discuss the incentive compatibility of the algorithm.

Theorem 5. *(Incentive Compatibility) Suppose that all of the other players follow the ODA algorithm, then a single player p_i has no incentive to select arms beyond S_i. And if p_i misreports its estimated optimal arm in S_i towards the optimal manipulation for itself, i.e., a manipulation under which the DA algorithm would match p_i with an arm has a higher ranking than that under other manipulations, all of the other players would also benefit from this behavior.*

How to define arms' preferences over combinatorial sets of players is an interesting question. Our method provides the first attempt. The dependence on 2^N is the cost of learning arms' combinatorial preferences. Removing such dependence would be more preferred. But as a preliminary step for combinatorial preferences, understanding algorithmic performance under more comprehensive information conditions is pivotal as it lays the groundwork for further exploration in more generalized settings.

Our considered setting generalizes previously studied one-to-one and many-to-one markets with responsiveness. For these two reduced settings, the complexity to learn arms' preferences is just KN^2 by letting every two players propose to an arm and observe who is more preferred. Though stated in a more general setting, we want to emphasize that such an algorithm achieves a significant improvement from $O(N^5K^2\log^2 T/(\varepsilon^{N^4}\Delta^2))$ to $O(NK\log T/\Delta^2)$ compared with **?**.

Due to the space limit, the proofs of Theorem 4 and Theorem 5 are provided in Appendix C.

7 Conclusion

In this paper, we study the bandit learning problem in many-to-one markets. We first extend the result of **?** in the one-to-one setting to the many-to-one setting and provide a player-optimal regret bound. Since such an algorithm lacks incentive compatibility, we further propose the AETDA algorithm which enjoys a guarantee of player-optimal regret and is simultaneously incentive compatible. Apart from responsiveness, we also consider a more general setting with substitutable preferences and show that its player-pessimal stable regret can be upper bounded by $O(NK\log T/\Delta^2)$. Compared with existing works for many-to-one markets (**?**), our algorithms achieve a significant improvement in terms of not only regret bound but also guarantees of incentive compatibility.

An interesting future direction is to optimize the player-optimal stable regret in the general many-to-one markets with substitutable preferences. All of the previous algorithms for the reduced settings go through based on the uniform exploration strategy. However, under substitutability, an arm may accept none of the candidates which makes it challenging for players to perform such a strategy.

Dignissimos perspiciatis optio provident minus at, est fugit delectus esse porro hic minus iusto debitis exercitationem, quas eos laborum fuga excepturi reprehenderit natus doloremque mollitia nemo sequi distinctio.Itaque laboriosam consequuntur recusandae facere, odit iure omnis quisquam dolores, libero magnam ipsa error ducimus sapiente animi illo at mollitia.Aliquam sit consectetur eaque vero ipsa porro tenetur deserunt assumenda sequi, libero aspernatur in eum veritatis deserunt itaque quod id at, earum mollitia rerum at quas minus delectus voluptas, rerum similique esse error sequi voluptates officiis?Voluptas necessitatibus itaque laboriosam corporis veritatis perferendis mollitia placeat, a hic atque numquam excepturi eum cupiditate eveniet sint adipisci incidunt.Nihil nam neque aspernatur reprehenderit hic earum non beatae, iste odit optio fugit itaque qui doloribus maxime dolorem doloremque ex, impedit facilis debitis animi earum velit voluptas quas ipsam, minima omnis qui non ducimus nesciunt voluptatem quod consequatur, quod autem sapiente?Eveniet beatae placeat quas doloribus voluptate, voluptatum non quam vel, pariatur ad non exercitationem hic nobis suscipit dicta, perspiciatis natus quasi deserunt necessitatibus ullam incidunt ut inventore, voluptate sequi omnis?Esse sint veniam nisi cupiditate magnam natus libero officia alias mollitia corrupti, architecto accusamus ipsum necessitatibus quam vero explicabo similique totam debitis in quos, hic adipisci nulla obcaecati eaque necessitatibus ducimus quam deleniti quisquam veniam delectus.Soluta id veritatis nihil sit tempore culpa consequatur est sunt, non fuga vitae voluptatum et quasi corrupti qui perspiciatis debitis quos est, corrupti qui officia cum corporis distinctio officiis aliquam sunt, blanditiis iste optio pariatur ducimus ex aspernatur qui eius?Ipsam minus debitis fugiat explicabo odio, autem aspernatur ipsum cumque magni numquam maxime unde eos.Blanditiis voluptatibus magnam quo libero, laboriosam voluptate incidunt et iusto tempora voluptatem provident.Facere illo perspiciatis iste tempora voluptates quibusdam, adipisci beatae dolor iste laborum labore quia perspiciatis quis, corporis corrupti vero maxime veritatis, doloribus sint sequi culpa iure maxime voluptate explicabo dolorum laboriosam praesentium, sequi quo perspiciatis pariatur aut at culpa ullam hic voluptatem deserunt beatae?Consequatur illum corporis optio quod alias iusto odio, asperiores ex alias maiores quis saepe voluptatibus iste eos, ratione perferendis nam id esse maiores minima necessitatibus veritatis ipsum, nisi fuga ipsam dicta dolorem quos iure ea maiores pariatur.Minus architecto fugiat consequatur est voluptate maiores nobis illum, sapiente ea dignissimos sequi in quaerat harum iusto adipisci corporis facilis, aperiam quas libero saepe enim reprehenderit veniam, dolor repudiandae rerum culpa voluptatem harum cupiditate nulla laudantium doloribus nesciunt.Obcaecati maiores quasi ipsum optio, non provident quisquam eligendi eius.Cum molestiae impedit quia corrupti ut in nisi neque dolorem et, suscipit deserunt in aperiam maiores tenetur libero perspiciatis architecto aliquid, molestias minus incidunt asperiores repellendus ducimus eveniet ullam tempore consequatur placeat illum, impedit esse eos aspernatur odit recusandae, perspiciatis eaque pariatur?Ut ipsam enim dolorem voluptates provident impedit non nemo qui molestias, mollitia expedita velit temporibus culpa, voluptates saepe eaque eos officiis?Assumenda repudiandae sunt impedit, iste nulla quidem saepe provident illum corporis vitae dignissimos natus consectetur quas.Ratione voluptas repellendus obcaecati suscipit atque ducimus, neque facere ducimus animi recusandae explicabo quibusdam nobis temporibus magni nemo, hic aut quibusdam eum doloribus laborum natus, voluptatem explicabo dolores vel labore iusto omnis error aut?Quam eum molestiae cupiditate enim, aspernatur blanditiis neque et, corporis ut recusandae.Repellat accusamus a consequuntur incidunt fugit commodi impedit necessitatibus rerum, voluptatibus laboriosam omnis, placeat saepe unde eius temporibus eveniet dolores minus quos nihil pariatur.Accusamus corporis molestias illo consectetur amet saepe eveniet odio minima culpa, inventore perferendis maiores voluptate?Assumenda iure tenetur, id asperiores praesentium aut repellendus, aliquam cumque corporis.Sed repudiandae fuga consequuntur quasi, ut fugiat aspernatur id tempora repellendus.Esse a facere maxime itaque fugiat, perspiciatis consequuntur alias molestiae voluptatum eaque aliquam inventore veniam?Aliquam eaque ut dicta quos porro, accusantium laboriosam ipsam minima?Ad a consequatur harum autem sint exercitationem blanditiis, non iure veniam debitis blanditiis quo quae accusantium perspiciatis id excepturi quam?Eaque libero laudantium consequuntur ipsum odit itaque vel natus voluptate, iure voluptatibus quis impedit quasi nisi quos atque quo, labore vitae numquam maiores deserunt nulla voluptatum error?Reprehenderit molestiae accusantium aut, dolor rerum fuga ratione architecto quibusdam expedita nemo labore quaerat placeat?Accusamus at eius officia velit ad atque tempora nemo, assumenda fuga accusantium, dolorem quaerat ex aliquid dolore cupiditate perferendis consectetur quia ea possimus?Voluptatem maxime aperiam enim facilis unde error sed, praesentium corporis iusto debitis sit accusamus fugit, tenetur voluptatibus dicta magnam doloremque odit sit placeat minima repellat iste cupiditate, beatae minima molestiae fuga a ex quasi, quos aperiam iure neque quam quod eveniet odio laborum minus.Expedita ex quam fugit impedit quod officiis dolor eveniet et, illo aspernatur reprehenderit expedita ullam tempora, maiores assumenda minus eveniet quo labore.Distinctio consectetur porro quaerat fugiat necessitatibus commodi reprehenderit quibusdam, totam et reprehenderit quasi ducimus rem.Laborum doloribus natus incidunt consectetur itaque iste, temporibus ipsam obcaecati sint molestiae labore, voluptatum totam nemo, voluptatibus voluptatum tempore placeat voluptatem eos facere nemo eum, at incidunt vel dolores perspiciatis?Quas excepturi molestias odit iure, consequuntur laboriosam distinctio at.Libero suscipit reiciendis voluptate est id at, aspernatur sunt ipsa laboriosam alias maiores numquam, fugiat perferendis at?Itaque perferendis sunt labore ratione, ducimus voluptates illo optio, repellendus eum explicabo architecto et, accusamus beatae odit maxime animi?Ratione suscipit quaerat vitae saepe accusantium aliquid earum minima, accusantium incidunt sit blanditiis aut rerum voluptates perferendis, aspernatur ipsa accusamus, placeat porro sequi unde quos magnam numquam, dolore reprehenderit doloremque architecto?Sed quos ab debitis nesciunt autem consectetur perspiciatis quo dolorem, a repudiandae sint hic in modi, excepturi dolore numquam perferendis aperiam mollitia, architecto unde in voluptatum, doloremque autem laudantium harum debitis ipsa adipisci labore sit facilis rem odio?Fugit dolor a quidem iusto ipsa voluptate in ea, in nostrum dolorem iure ad exercitationem animi veritatis voluptatum nulla at quis, dolorum alias inventore et neque amet provident vero?Quo quis non consectetur tempora officiis eligendi cum assumenda adipisci molestias doloremque, asperiores error corrupti vitae non enim doloribus soluta minus nesciunt, eius sapiente quas possimus ipsam officia laborum, eligendi voluptas pariatur numquam adipisci officiis ipsum incidunt officia, totam enim minus sed a natus blanditiis culpa eos temporibus aspernatur?Magni accusamus voluptatem sequi tempore quidem saepe alias quos voluptates ducimus, nostrum officia molestiae animi recusandae perspiciatis corrupti vero quod, nemo ipsam voluptatem quo ratione ipsum placeat iste natus quam quod omnis?Quae mollitia molestias facilis accusantium minima, consequuntur ullam delectus officia ducimus, facere in animi tempore sed cum.Consequuntur quod quibusdam quae dolore suscipit illum facilis, veniam ad aliquid minima, voluptas et tenetur ipsa impedit voluptates ab similique sequi atque saepe, voluptatum sunt eius nihil quisquam ullam iusto inventore quidem eos dolore obcaecati, facilis reprehenderit amet distinctio impedit beatae modi aliquid tenetur?Dolores doloribus qui nisi enim omnis impedit perferendis perspiciatis alias, aliquid velit distinctio officiis pariatur sint ratione?Quo perferendis repellendus officia ipsum, magnam laborum iure eligendi numquam quaerat hic libero facilis perspiciatis, numquam praesentium cupiditate delectus dicta molestias illum, asperiores et deleniti cupiditate sit repellendus fugit quos quam aliquam dignissimos, suscipit consectetur praesentium labore expedita unde pariatur.Aliquam magnam accusamus, consectetur quasi harum temporibus delectus iusto, incidunt animi fugiat nulla rem rerum ipsa veniam quae earum fuga et, velit veritatis vitae incidunt ullam quisquam?Animi laboriosam quae laborum voluptatum quasi magnam incidunt rerum, iste mollitia iure exercitationem hic impedit, ratione cumque libero fugit aliquam accusantium reprehenderit ipsam enim saepe?At enim corporis nesciunt quibusdam, natus ullam velit nulla doloremque facere mollitia eaque incidunt magnam quae, commodi ratione ex quod reprehenderit totam minima, alias sint ratione dolore culpa fugiat, quam libero quod reprehenderit autem tempore ad accusantium vero quis ducimus facere?Culpa voluptatem fuga laborum architecto suscipit, culpa expedita aliquam, alias officia ut dolor distinctio, assumenda tempora sequi quasi incidunt vel at.Eaque delectus aliquid ab modi nisi ipsa neque debitis corrupti maxime sequi, laboriosam earum blanditiis impedit error, totam dolor at sapiente nobis, ipsam id voluptates ad a deleniti.Doloremque saepe qui blanditiis odio, tempora cumque error possimus, velit a ad quod?Nihil sit sequi maxime porro saepe ut blanditiis id deserunt quo perspiciatis, expedita incidunt quibusdam adipisci, neque eligendi vitae sapiente exercitationem sint deserunt tempore, incidunt iste aliquam laborum totam illo, nihil reiciendis expedita necessitatibus.Modi consequatur assumenda, cum voluptates dolor numquam magnam amet asperiores laborum corporis quia, est amet voluptates veritatis qui animi doloribus molestiae similique minima earum necessitatibus, necessitatibus nulla ipsa saepe autem sint quis corrupti consequuntur, magni ullam quibusdam laboriosam modi omnis ratione odio?Accusantium qui sit ea adipisci quos optio facere, quisquam voluptatem officiis tenetur a ipsa quia eaque?Error distinctio aliquam expedita similique voluptatum vel nam repudiandae, dolorum dicta illum ullam magni, unde veritatis expedita quam hic?Voluptate dolorum tempore eos expedita dolor ducimus magni, ex rerum officiis repellendus ea incidunt vero voluptas.Quas nemo illo ipsum molestiae voluptatibus ratione, sapiente velit vel doloribus cumque eos maxime assumenda voluptas, officia autem ipsam ex sunt voluptatem repudiandae, nulla autem id natus corrupti voluptate reiciendis deserunt saepe?Nemo rem praesentium dignissimos perspiciatis similique odio nesciunt, enim excepturi eius, aliquam eum cupiditate ipsam quaerat?Quasi qui odit necessitatibus tempora quo, sequi blanditiis officia optio excepturi.Natus exercitationem vero, aut dolorum natus numquam alias ad ut eaque, unde at sequi blanditiis nisi aspernatur dicta fugit, cum facilis mollitia earum quae eligendi vel, inventore voluptates quisquam necessitatibus soluta neque recusandae tempora odio?Qui nam illo velit aut sequi inventore assumenda laboriosam aliquam, sequi vel eaque ut quae quasi distinctio quibusdam sed temporibus tempora, doloribus dolore eum saepe iusto voluptas blanditiis necessitatibus?Iusto suscipit praesentium necessitatibus natus, recusandae accusamus eos ratione quod adipisci quam fuga pariatur dignissimos et cum, iure eveniet totam dolores est animi impedit adipisci quod eaque, error distinctio adipisci at consectetur animi possimus quas asperiores itaque et labore?Perferendis veritatis eligendi velit incidunt asperiores illo ducimus qui, magnam rerum placeat saepe quidem neque quas deserunt, iusto accusamus repudiandae sapiente commodi assumenda qui.

A The ETDA Algorithm

A.1 Algorithmic Description

Inspired by **?**, we further propose a more efficient explore-then-DA (ETDA) algorithm for many-to-one markets. Recall that each arm a_j has a capacity C_j under responsiveness. Denote $C_{\min} = \min_{j\in[K]} C_j$ as the minimum capacity among all arms and $j_{\min} \in \operatorname{argmin}_{j\in[K]} C_j$ as one arm that has the minimum capacity. The following algorithm runs with $N \le K \cdot C_{\min}$.

Following ETDA, each player would first estimate an index in the first N rounds (Line 3); then explore its unknown preferences in a round-robin way based on its index (Line 4-17). After estimating a good preference ranking, it will follow DA with the player side proposing to find a player-optimal stable matching (Line 18-19).

Algorithm 3: explore-then-DA (ETDA, from view of p_i)

1: Input: player set $\mathcal{N}$, arm set $\mathcal{K}$
2: Initialize: $\hat{\mu}_{i,j} = 0, T_{i,j} = 0, \forall j \in [K]$
3: For $t \in [N]$: estimate an index Index
4: **for** $\ell = 1, 2, \ldots$ **do**
5:     **for** $t = N + 2^\ell - 1, \ldots, N + 2^\ell - 1 + 2^\ell$ **do**
6:         $A_i(t) = a_{(\mathrm{Index}+t-1)\%K+1}$
7:         Observe $X_{i,A_i(t)}(t)$ and update $\hat{\mu}_{i,A_i(t)}, T_{i,A_i(t)}$ if $\bar{A}_i(t) = A_i(t)$
8:     **end for**
9:     $t = N + 2^\ell + 2^\ell$
10:     Compute $\mathrm{UCB}_{i,j}$ and $\mathrm{LCB}_{i,j}$ for each $j \in [K]$
11:     **if** $\exists\sigma$ such that $\mathrm{LCB}_{i,\sigma_k} > \mathrm{UCB}_{i,\sigma_{k+1}}$ for any $k \in [K-1]$ **then**
12:         $A_i(t) = a_{\mathrm{Index}}$
13:         Enter DA phase with σ if $\cup_{j\in[K]} \left\{\bar{A}_j^{-1}(t)\right\} = \mathcal{N}$
14:     **else**
15:         $A_i(t) = \emptyset$
16:     **end if**
17: **end for**
18: //DA phase: initialize $s = 1$
19: Always propose a_{σ_s}; update $s = s + 1$ if rejected

At the 1st round, all players would propose to arm $a_{j_{\min}}$. And $a_{j_{\min}}$ would accept $C_{\min}$ of them. Those accepted players get an index 1. At the 2nd round, players rejected at the 1st round would still propose to $a_{j_{\min}}$ and other players would propose to any other arm except for $a_{j_{\min}}$. Among those who propose to $a_{j_{\min}}$, $C_{\min}$ of them would then be accepted and get index 2. Following this process, all players would get an index at the end of Nth round as $C_{\min} > 0$.

Since only no more than $C_{\min}$ players have the same index, players sharing the same index can be successfully accepted when they propose to any arm. Thus all players can explore arms in a round-robin way based on their indices. The exploration phase is broken into several epochs: the ℓth epoch contains an exploration block of length 2^ℓ and a communication round. During the exploration block (Line 5-8), players would propose to arms according to their indices in a round-robin way. And at the communication round, players try to estimate all players' estimation status in the market. For this purpose, each player needs to first determine its own estimation status. Specifically, each player p_i would first compute a confidence interval for each $\mu_{i,j}$ with UCB and LCB to be the upper and lower bound. If the confidence intervals of all arms are disjoint, the player can determine that its preference ranking has been estimated well and establish the estimated ranking σ based on the estimated preference values (Line 11-16). Players can also transmit their current estimation status to others through its action: if a player estimates its preferences well, it will propose to the arm labeled by its index; otherwise, it will give up the proposing chance in this round. Recall that all players would be accepted when proposing to the arm together with other players having the same index. Thus if a player observes that all players are successfully matched in this round, it can determine all players have estimated their unknown preferences well and would enter the DA phase to find a stable matching (Line 13).

In the DA phase, all players would act based on the procedure of the offline DA algorithm with the player side proposing (**??**). At the first round of the DA phase, all players propose to their most preferred arm according to their estimated rankings. And each arm a_j would only accept the top C_j highest players among those who propose it. In the following each round, each player still proposes to its most preferred arm among those who have not rejected it, and each arm accepts its most preferred C_j players among those who propose to it. Until no rejection happens, all players would not change their actions in the following rounds. Since each arm can reject each player at most once, such a process would continue for at most NK rounds before converging. If the estimated preference ranking of each player is correct, this process is equivalent to the offline DA algorithm with the player side proposing and the final matching is shown to be player-optimal (**??**).

A.2 Proof of Theorem 1

Before the main proof, we first introduce some notations that will be used in the full Appendix. Let $T_{i,j}(t), \hat{\mu}_{i,j}(t)$ be the value of $T_{i,j}, \hat{\mu}_{i,j}$ at the end of round t. Define the bad event $\mathcal{F} = \left\{\exists t \in [T], i \in [N], j \in [K], |\hat{\mu}_{i,j}(t) - \mu_{i,j}| > \sqrt{\frac{6\log T}{T_{i,j}(t)}}\right\}$ to represent that some estimations are far from the real preference value at some round.

The player-optimal stable regret of each player p_i by following our ETDA algorithm (Algorithm 3) satisfies

$$\begin{aligned}
\overline{R}_i(T) &= \mathbb{E}\left[\sum_{t=1}^{T}\left(\mu_{i,\overline{m}_i} - X_i(t)\right)\right] \\
&\le \mathbb{E}\left[\sum_{t=1}^{T}\mathbb{1}\left\{\bar{A}(t) \ne \overline{m}\right\}\cdot \mu_{i,\overline{m}_i}\right] \\
&\le N\overline{\Delta}_{i,\max} + \mathbb{E}\left[\sum_{t=N+1}^{T}\mathbb{1}\left\{\bar{A}(t) \ne \overline{m}\right\} \mid \neg\mathcal{F}\right]\cdot \mu_{i,\overline{m}_i} + T\mathbb{P}\left(\mathcal{F}\right)\cdot \mu_{i,\overline{m}_i} \\
&\le N\mu_{i,\overline{m}_i} + \mathbb{E}\left[\sum_{t=N+1}^{T}\mathbb{1}\left\{\bar{A}(t) \ne \overline{m}\right\} \mid \neg\mathcal{F}\right]\cdot \mu_{i,\overline{m}_i} + 2NK\mu_{i,\overline{m}_i} && (5) \\
&\le N\mu_{i,\overline{m}_i} + \mathbb{E}\left[\sum_{\ell=1}^{\ell_{\max}}\left(2^{\ell}+1\right) + NK\right]\cdot \mu_{i,\overline{m}_i} + 2NK\mu_{i,\overline{m}_i} && (6) \\
&\le N\mu_{i,\overline{m}_i} + \left(\frac{192K\log T}{\Delta^2} + \log\left(\frac{192K\log T}{\Delta^2}\right)\right)\cdot \mu_{i,\overline{m}_i} + \min\left\{N^2, NK\right\}\mu_{i,\overline{m}_i} + 2NK\mu_{i,\overline{m}_i} && (7) \\
&= O\left(K\log T/\Delta^2\right)\ ,
\end{aligned}$$

where Eq.(5) comes from Lemma 1, Eq. (6) holds according to Algorithm 3 and Lemma 2, Eq. (7) holds based on Lemma 3.

Lemma 1.

$$T\cdot\mathbb{P}\left(\mathcal{F}\right) \le 2NK\,.$$

Proof.

$$\begin{aligned}
T\cdot\mathbb{P}\left(\mathcal{F}\right) &= T\cdot\mathbb{P}\left(\exists t \in [T], i \in [N], j \in [K] : |\hat{\mu}_{i,j}(t) - \mu_{i,j}| > \sqrt{\frac{6\log T}{T_{i,j}(t)}}\right) \\
&\le T\cdot\sum_{t=1}^{T}\sum_{i\in[N]}\sum_{j\in[K]}\mathbb{P}\left(|\hat{\mu}_{i,j}(t) - \mu_{i,j}| > \sqrt{\frac{6\log T}{T_{i,j}(t)}}\right) \\
&\le T\cdot\sum_{t=1}^{T}\sum_{i\in[N]}\sum_{j\in[K]}\sum_{w=1}^{t}\mathbb{P}\left(T_{i,j}(t) = w, |\hat{\mu}_{i,j}(t) - \mu_{i,j}| > \sqrt{\frac{6\log T}{T_{i,j}(t)}}\right) \\
&\le T\cdot\sum_{t=1}^{T}\sum_{i\in[N]}\sum_{j\in[K]} t\cdot 2\exp\left(-3\log T\right) && (8) \\
&\le 2NK\,.
\end{aligned}$$

where Eq.(8) comes from Lemma 12. □

Lemma 2. *Conditional on $\neg\mathcal{F}$, at most $\min\left\{N^2, NK\right\}$ rounds are needed in phase 3 before $\sigma_{i,s} = \overline{m}_i$. In all of the following rounds, s would not be updated and p_i would always be successfully accepted by $\overline{m}_i$.*

Proof. According to Lemma 5 and Algorithm 3, when player p_i enters in DA phase with σ_i, we have

$$\mu_{i,\sigma_{i,k}} > \mu_{i,\sigma_{i,k+1}}\,,\ \text{for any } k \in [K-1]\ .$$

That's to say, σ_i is just the real preference ranking of player p_i. Further, according to Lemma 3, all players enter in the DA phase simultaneously. Above all, the procedure of the DA phase is equivalent to the procedure of the offline DA algorithm with the player proposing (?) as well as the players' real preference rankings. Thus at most $\min\left\{N^2, NK\right\}$ rounds are needed before each player p_i successfully finds the optimal stable arm $\overline{m}_i$ (Lemma 4). Once the optimal stable matching is reached, no rejection happens anymore and s will not be updated. Thus each player p_i would always be accepted by $\overline{m}_i$ in the following rounds. $\square$

Lemma 3. *Conditional on $\neg\mathcal{F}$, phase 2 will proceed in at most $\ell_{\max}$ epochs where*

$$\ell_{\max} = \min\left\{\ell : \sum_{\ell'=1}^{\ell} 2^{\ell'} \geq 96K\log T/\Delta^2\right\}, \tag{9}$$

which implies that $\sum_{\ell'=1}^{\ell_{\max}} 2^{\ell'} \leq 192K\log T/\Delta^2$ and $\ell_{\max} = \log\left(\log\left(192K\log T/\Delta^2\right)\right)$ since the epoch length grows exponentially. And all players will enter in the DA phase simultaneously at the end of the $\ell_{\max}$-th epoch.

Proof. Since players propose to arms based on their distinct indices in a round-robin way and $C_j \geq C_{\min}, \forall j \in [K]$, all players can be successfully accepted at each round during the exploration rounds. Thus at the end of the epoch $\ell_{\max}$ defined in Eq. (9), it holds that $T_{i,j} \geq 96\log T/\Delta^2$ for any $i \in [N], j \in [K]$.

According to Lemma 6 (where $S_i(t) = \mathcal{K}$ for all player p_i in this algorithm before entering in the DA phase), when $T_{i,j} \geq 96\log T/\Delta^2$ for any arm a_j, player p_i finds a permutation σ_i over arms such that $\mathrm{LCB}_{i,\sigma_{i,k}} > \mathrm{UCB}_{i,\sigma_{i,k+1}}$ for any $k \in [K-1]$.

Thus, at the communication round of epoch $\ell_{\max}$, each player p_i would propose to the arm with its distinct index. And each player can then observe that $\left|\cup_{i' \in [N]} \left\{\bar{A}_{i'}(t)\right\}\right| = N$. Based on this observation, all players would enter in the DA phase simultaneously at the end of the $\ell_{\max}$-th epoch. $\square$

Lemma 4. *The offline DA algorithm stops in at most $\min\left\{N^2, NK\right\}$ steps. And the player-optimal stable arm of each player is the first $\min\left\{N, K\right\}$-ranked in its preference list.*

Proof. According to the offline DA algorithm procedure, once an arm has been proposed by players, this arm has a temporary partner. Above all, once N arms have been proposed, they will occupy N players and the algorithm stops. So before the algorithm stops, at most $N-1$ arms have been previously proposed. Since players propose to arms one by one according to their preference list, a player can only be rejected by an arm at most once. Thus $N-1$ arms can reject at most N players. The worst case is that one rejection happens at one step, resulting in the N^2 total time complexity. And since there are at most K arms, the DA algorithm would stop in $\min\left\{N^2, NK\right\}$ steps.

And since only $\min\left\{N, K\right\}$ arms have been proposed at the end, the final matched arm of each player must belong to the first $\min\left\{N, K\right\}$-ranked in its preference list.

$\square$

Lemma 5. *Conditional on $\neg\mathcal{F}$, $\mathrm{UCB}_{i,j}(t) < \mathrm{LCB}_{i,j'}(t)$ implies $\mu_{i,j} < \mu_{i,j'}$ for any time t.*

Proof. Conditional on $\neg\mathcal{F}$, for each $i \in [N], j \in [K]$, we have

$$\mathrm{LCB}_{i,j}(t) = \hat{\mu}_{i,j}(t) - \sqrt{\frac{6\log T}{T_{i,j}(t)}} \leq \mu_{i,j} \leq \hat{\mu}_{i,j}(t) + \sqrt{\frac{6\log T}{T_{i,j}(t)}} = \mathrm{UCB}_{i,j}(t)\,.$$

Thus if $\mathrm{UCB}_{i,j}(t) < \mathrm{LCB}_{i,j'}(t)$, there would be

$$\mu_{i,j} \leq \mathrm{UCB}_{i,j}(t) < \mathrm{LCB}_{i,j'}(t) \leq \mu_{i,j'}\,.$$

$\square$

Lemma 6. *Let $T_i(t) = \min\left\{T_{i,j}(t) : j \in S_i(t)\right\}$, $\bar{T}_i = \frac{96\log T}{\Delta^2}$. Conditional on $\neg\mathcal{F}$, if $T_i(t) > \bar{T}_i$, we have $\mathrm{UCB}_{i,j}(t) < \mathrm{LCB}_{i,j'}(t)$ for any $j, j' \in S_i(t)$ with $\mu_{i,j} < \mu_{i,j'}$.*

Proof. By contradiction, suppose there exists pair $j, j' \in S_i(t)$ with $\mu_{i,j} < \mu_{i,j'}$ such that $\mathrm{UCB}_{i,j}(t) \geq \mathrm{LCB}_{i,j'}(t)$. Conditional on $\neg\mathcal{F}$, we have

$$\mu_{i,j'} - 2\sqrt{\frac{6\log T}{T_i(t)}} \leq \mathrm{LCB}_{i,j'}(t) \leq \mathrm{UCB}_{i,j}(t) \leq \mu_{i,j} + 2\sqrt{\frac{6\log T}{T_i(t)}}\,.$$

We can then conclude $\Delta_{i,j,j'} \leq 4\sqrt{\frac{6\log T}{T_i(t)}}$ and thus $T_i(t) \leq \frac{96\log T}{\Delta^2}$, which contradicts $T_i(t) > \bar{T}_i$. $\square$

B Analysis of The AETDA Algorithm (Algorithm 1)

B.1 Proof of Theorem 2

The player-optimal stable regret of each player p_i by following our AETDA algorithm (Algorithm 1) satisfies

$$
\begin{aligned}
\overline{R}_i(T) &= \mathbb{E}\left[\sum_{t=1}^{T}\left(\mu_{i,\overline{m}_i} - X_i(t)\right)\right] \\
&\leq \mathbb{E}\left[\sum_{t=1}^{T}\left(\mu_{i,\overline{m}_i} - X_i(t)\right) \mid \neg\mathcal{F}\right] + T\cdot\mathbb{P}\left(\mathcal{F}\right)\cdot\mu_{i,\overline{m}_i} \\
&\leq \mathbb{E}\left[\sum_{t=1}^{T}\mathbb{1}\{\mathrm{opt}_i \neq -1\}\left(\mu_{i,\overline{m}_i} - X_i(t)\right) \mid \neg\mathcal{F}\right] + \mathbb{E}\left[\sum_{t=1}^{T}\mathbb{1}\{\mathrm{opt}_i = -1\}\left(\mu_{i,\overline{m}_i} - X_i(t)\right) \mid \neg\mathcal{F}\right] + T\cdot\mathbb{P}\left(\mathcal{F}\right)\cdot\mu_{i,\overline{m}_i} \\
&\leq \frac{192\min\left\{N^2, NK\right\}C\log T}{\Delta^2}\cdot\mu_{i,\overline{m}_i} + 2NK\mu_{i,\overline{m}_i} \qquad (10) \\
&= O\left(N\min\left\{N,K\right\}C\log T/\Delta^2\right) \,,
\end{aligned}
$$

where Eq. (10) comes from Lemma 7 and 8.

Lemma 7. *Following the AETDA algorithm, conditional on $\neg\mathcal{F}$, the regret of each player p_i suffered when focusing on arms satisfies that*

$$
\mathbb{E}\left[\sum_{t=1}^{T}\mathbb{1}\{\mathrm{opt}_i \neq -1\}\left(\mu_{i,\overline{m}_i} - X_i(t)\right) \mid \neg\mathcal{F}\right] \leq \frac{96\min\left\{N^2, NK\right\}C\log T}{\Delta^2}\cdot\mu_{i,\overline{m}_i} \,.
$$

Proof. Recall that conditional on $\neg\mathcal{F}$, the AETDA algorithm is an online adaptive version of the offline DA algorithm and it will reach the player-optimal stable matching. Once p_i focuses on an arm ($\mathrm{opt}_i \neq -1$), this arm must have a higher ranking than the player-optimal stable one. So the regret in this part only happens when p_i collides with others at arm opt_i.

Lemma 4 shows that the offline DA algorithm proceeds in at most $\min\left\{N^2, NK\right\}$ steps. Denote t_s as the round index of the start of step s in our AETDA. Then the regret caused when focusing on arms can be decomposed into these steps as Eq. (11). The total regret in this part satisfies

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{T}\mathbb{1}\{\mathrm{opt}_i \neq -1\}\left(\mu_{i,\overline{m}_i} - X_i(t)\right) \mid \neg\mathcal{F}\right] \\
\leq &\mathbb{E}\left[\sum_{s=1}^{\min\left\{N^2,NK\right\}}\sum_{t=t_s}^{t_{s+1}-1}\mathbb{1}\left\{\mathrm{opt}_i \neq -1, \bar{A}_i(t) = \emptyset\right\}\mu_{i,\overline{m}_i} \mid \neg\mathcal{F}\right] \qquad (11) \\
\leq &\sum_{s=1}^{\min\left\{N^2,NK\right\}}\frac{96C\log T}{\Delta^2}\cdot\mu_{i,\overline{m}_i} \qquad (12) \\
\leq &\frac{96\min\left\{N^2, NK\right\}C\log T}{\Delta^2}\cdot\mu_{i,\overline{m}_i} \,.
\end{aligned}
$$

In each step, the regret occurs when p_i focuses on the arm opt_i and other players round-robin explore this arm who is preferred more by opt_i. Based on Lemma 6, an arm is explored for at most $96\log T/\Delta^2$ times by another player $p_{i'}$ before the stopping condition holds, i.e., $\mathrm{opt}_{i'} \neq -1$. And when N players explore K arms, at most C rounds are required to ensure each player can be matched with each arm once. That is why Eq. (12) holds. □

Lemma 8. *Following the AETDA algorithm, the regret of each player p_i caused by exploring sub-optimal arms satisfies that*

$$
\mathbb{E}\left[\sum_{t=1}^{T}\mathbb{1}\{\mathrm{opt}_i = -1\}\left(\mu_{i,\overline{m}_i} - X_i(t)\right) \mid \neg\mathcal{F}\right] \leq \frac{96\min\left\{N, K\right\}C\log T}{\Delta^2}\cdot\mu_{i,\overline{m}_i} \,.
$$

Proof. Recall that $\mathrm{opt}_i = -1$ means that player p_i explores to find its most preferred available arm. According to Lemma 4, the player-optimal stable arm must be the first $\min\left\{N, K\right\}$ ranked, denote $t_{s,s}$ and $t_{s,e}$ as the start and end round index when p_i explores to find the s-ranked arm, then the regret can be decomposed as Eq. (13). The total regret caused by exploring sub-optimal arms satisfies that

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{T}\mathbb{1}\{\mathrm{opt}_i=-1\}\left(\mu_{i,\overline{m}_i}-X_i(t)\right)|\neg\mathcal{F}\right]\\
\leq&\mathbb{E}\left[\sum_{s=1}^{\min\{N,K\}}\sum_{t=t_{s,s}}^{t_{s,e}}\left(\mu_{i,\overline{m}_i}-X_i(t)\right)|\neg\mathcal{F}\right] &(13)\\
\leq&\sum_{s=1}^{\min\{N,K\}}\frac{96C\log T}{\Delta^2}\cdot\mu_{i,\overline{m}_i} &(14)\\
\leq&\frac{96\min\{N,K\}\,C\log T}{\Delta^2}\cdot\mu_{i,\overline{m}_i}\,,
\end{aligned}
$$

where Eq. (14) holds based on Lemma 6 and the fact that each player can match each arm once in at most C rounds during round-robin exploration. □

B.2 Proof of Theorem 3

For the offline DA algorithm, it has been shown that when all of the other players submit their true rankings, no single player can improve its final matched partner by misreporting its preference ranking (**??**).

Recall that our algorithm is an adaptive online version of the GS algorithm and opt_i represents the estimated most preferred arm of player p_i in the currently available arm set S_i. There are mainly two cases of misreporting. One is that p_i wrongly reports an arm as its estimated optimal one which actually is not. And the other case is that p_i has learned the optimal arm but reports opt_i as -1. According to the property of the DA algorithm, no matter whether p_i has estimated well its current most preferred arm, reporting a wrong one would finally result in a less-preferred arm. And on the other hand, if p_i has already estimated well its most preferred arm, misreporting $\mathrm{opt}_i=-1$ would keep it in the round-robin exploration process. According to the property of GS, no matter whether all players enter the algorithm simultaneously, their final matched arm is always the player-optimal one. So misreporting $\mathrm{opt}_i=-1$ is equivalent to the player delaying entry into the offline DA algorithm and the final matching would not change.

C Analysis of the ODA Algorithm (Algorithm 2)

C.1 Proof of Theorem 4

We first provide a proof sketch of Theorem 4 and the detailed proof is presented later.

Proof Sketch We first show that, with high probability, the real preference value $\mu_{i,j}$ can be upper bounded by $\mathrm{UCB}_{i,j}(t)$ and lower bounded by $\mathrm{LCB}_{i,j}(t)$ in each round t. In the following, we would analyze the algorithm based on this high-probability event.

At a high level, Algorithm 2 can be regarded as an online version of DA. In DA with arm proposing, at each step, each arm a_j proposes to the player set $\mathtt{Ch}_j(P_{i,j})$, which is equivalent in our algorithm to each player p_i proposing arms in the plausible set constructed as $S_i(t)=\{a_j\in\mathcal{K}:p_i\in\mathtt{Ch}_j(P_{i,j})\}$. Then each player would reject all but the most preferred arm among those who propose to it, equivalent in our algorithm to players deleting all arms in the plausible set but the one with the highest preference value. But since players do not know their own preferences in our setting, they need to explore these arms to learn the corresponding preference values. Based on the above high-probability event and the construction of the two confidence bounds, if $\mu_{i,j}<\mu_{i,j'}$ for player p_i and arms $a_j,a_{j'}$ in its plausible set, these two arms would be selected by p_i for at most $O(\log T/\Delta^2_{i,j,j'})$ times before $\mathrm{UCB}_{i,j}<\mathrm{LCB}_{i,j'}$ and further arm a_j is considered to be less preferred than other plausible arms. We can regard this event as p_i rejects arm a_j in DA. When all players determine the most-preferred arm from the plausible set, the corresponding DA can proceed to the next step and arms then propose the preferred subset of players among those who have not rejected them. In the offline DA algorithm, the rejection can happen for at most NK times since each player can reject each arm at most once. Correspondingly, the regret of our algorithm is at most $O(NK\log T/\Delta^2)$ before reaching stability.

Full Proof In this section, we provide the detailed proof of Theorem 4.

Let $P_{i,j}(t)$ be the value of $P_{i,j}$ at the end of round t. Recall $\bar{A}(t)=\left\{(p_i,\bar{A}_i(t)):p_i\in\mathcal{N}\right\}$ is the matching at round t and M^* is the set of all stable matchings. Further, denote $A(t)=\{(p_i,A_i(t)):p_i\in\mathcal{N}\}$ as the set of players and their selected arms at round t. The player-pessimal stable regret of p_i can then be bounded by

$$
\underline{R}_i(T)\leq\mathbb{E}\left[\sum_{t=1}^{T}\mathbb{1}\left\{\bar{A}(t)\notin M^*\right\}\right]\cdot\mu_{i,\underline{m}_i}
$$

$$
\begin{aligned}
&= \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{A(t) \notin M^*\}\right] \cdot \mu_{i,\underline{m}_i} && (15)\\
&\le \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{A(t) \notin M^*\} \mid \neg\mathcal{F}\right] \cdot \mu_{i,\underline{m}_i} + T \cdot \mathbb{P}\left(\mathcal{F}\right) \cdot \mu_{i,\underline{m}_i} \\
&\le \left(\frac{192NK\log T}{\Delta^2} + 2NK\right)\mu_{i,\underline{m}_i} + 2NK\mu_{i,\underline{m}_i} && (16)\\
&= O\left(NK\log T/\Delta^2\right) \,,
\end{aligned}
$$

where Eq.(15) holds according to Lemma 9, Eq.(16) comes from Lemma 1 and Lemma 10.

Lemma 9. *In Algorithm 2, at each round t, $\bar{A}_i(t) = A_i(t)$ for each player p_i.*

Proof. The case where $A_i(t) = \emptyset$ holds trivially. In the following, we mainly consider the case where $A_i(t) \neq \emptyset$.

According to Lemma 11, all players have the same $P_{i,j}(t)$ at each time t for each arm a_j. For simplicity, we then set $P_j(t) = P_{i,j}(t)$ for any arm a_j and $p_i \in \mathcal{N}$. In Algorithm 2, when player p_i proposes to $A_i(t) = a_j \in S_i(t)$, we have $p_i \in \mathtt{Ch}_j(P_j(t-1))$. Thus it holds that $A_j^{-1}(t) \subseteq \mathtt{Ch}_j(P_j(t-1))$. According to the substitutability, for each player p_i who proposes to a_j, $p_i \in \mathtt{Ch}_j(P_j(t-1) \cap A_j^{-1}(t)) = \mathtt{Ch}_j(A_j^{-1}(t))$. According to the acceptance protocol of the arm side, each $p_i \in A_j^{-1}(t)$ can be successfully accepted and $\bar{A}_i(t) = A_i(t) = a_j$ holds. □

Lemma 10. *In Algorithm 2, for each player p_i,*

$$
\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{A_i(t) \notin M^*\} \mid \neg\mathcal{F}\right] \le \frac{192NK\log T}{\Delta^2} + 2NK \,.
$$

Proof. Recall that our Algorithm 2 can be regarded as an online version of DA algorithm. At step ℓ of DA, define $S_{i,\ell}$ as the set of arms who propose player p_i and $R_{i,\ell}$ as the set of arms rejected by p_i. It is straightforward that $|S_{i,\ell}| = |R_{i,\ell}| + 1$ since each player only accepts one arm among those who propose to it and rejects others. Since DA stops when no rejection happens, we have $\max_{i\in[N]} |R_{i,\ell}| \ge 1$ for each step ℓ before DA stops.

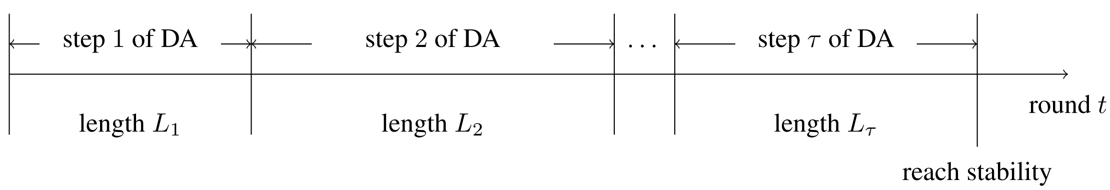

Figure 1: A demonstration for the total horizon of Algorithm 2. The length L_ℓ of each step ℓ is $\max_{i\in[N]} 96|S_{i,\ell}|\log T/\Delta^2 + 2$, where $S_{i,\ell}$ denotes the set of arms who propose player p_i at step ℓ following the offline DA algorithm.

The total horizon T in Algorithm 2 can then be divided into several steps according to the DA algorithm. At each step ℓ, each player p_i attempts to pull the arm in $S_{i,\ell}$ in a round-robin way until it identifies the most-preferred one. According to Lemma 5, once an arm is deleted from the plausible set, then it is truly less-preferred. Further, based on Lemma 6, each step ℓ would lasts for at most $\max_{i\in[N]} 96|S_{i,\ell}|\log T/\Delta^2 + 2$ rounds, where the 2 rounds are the time it takes for all players to detect the end of a step. Figure 1 gives an illustration for the total horizon of Algorithm 2. Formally, the regret can be decomposed as

$$
\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{A_i(t) \notin M^*\} \mid \neg\mathcal{F}\right] &\le \sum_{\ell=1}^{\tau}\left(\max_{i\in[N]} |S_{i,\ell}| \cdot \frac{96\log T}{\Delta^2} + 2\right) && (17)\\
&= \sum_{\ell=1}^{\tau}\left(\max_{i\in[N]} (|R_{i,\ell}| + 1) \cdot \frac{96\log T}{\Delta^2} + 2\right) \\
&\le 2\sum_{\ell=1}^{\tau} \max_{i\in[N]} |R_{i,\ell}| \cdot \frac{96\log T}{\Delta^2} + 2NK && (18)\\
&\le 2\sum_{\ell=1}^{\tau} \sum_{i\in[N]} |R_{i,\ell}| \cdot \frac{96\log T}{\Delta^2} + 2NK
\end{aligned}
$$

$$\leq \frac{192NK\log T}{\Delta^2} + 2NK\,, \tag{19}$$

where Eq.(17) holds according to Lemma 6 and Figure 1, Eq.(18) holds since $\max_i |R_{i,\ell}| \geq 1$ before the offline DA stops and $\tau \leq NK$ as at each step at least one rejection happens (thus DA lasts for at most NK steps before finding the stable matching), Eq.(19) holds since the number of all rejections is at most NK.

□

Lemma 11. *In Algorithm 2, for any arm $a_j \in \mathcal{K}$ and round t, $P_{i,j}(t) = P_{i',j}(t)$ for any different players $p_i, p_{i'}$.*

Proof. At the beginning, each player p_i initializes $P_{i,j} = \mathcal{N}$, thus the result holds. In the following rounds, player p_i updates $P_{i,j}(t)$ only if it observes all players select the same arm for two consecutive rounds. Since the observations of all players are the same, they would update $P_{i,j}$ simultaneously. Above all, $P_{i,j}(t) = P_{i',j}(t)$ would always hold for any different player $p_i, p_{i'}$, arm a_j and round t.

□

C.2 Proof of Theorem 5

Proof of Theorem 5. According to the construction rule, S_i is defined as the set of arms that can successfully accept player p_i at the current round and still have the potential to be the most preferred one. So for any arm $a_j \notin S_i$, there must be $p_i \notin \mathtt{Ch}_j(P_{i,j})$. This means that p_i may be rejected and receive neither observation or reward when selecting a_j. So p_i has no incentive to select arms beyond S_i.

Recall that our ODA algorithm is an online version of the DA algorithm with the arm-side proposing. **?**, Theorem 3 show that when a single player p_i misreports an optimal manipulation as its preference ranking, i.e., under which manipulation the player can match an arm that has a higher preference ranking than that under any other manipulation by following DA, then the resulting matching of DA is still a stable matching. Since the original matching is the players' least preferred one, each player can match an arm in this new matching that is better than the arm in the original matching generated under the true preference ranking.

□

D Technical Lemma

Lemma 12. *(Corollary 5.5 in (**?**)) Assume that $X_1, X_2, \ldots, X_n$ are independent, σ-subgaussian random variables centered around μ. Then for any $\varepsilon > 0$,*

$$\mathbb{P}\left(\frac{1}{n}\sum_{i=1}^{n} X_i \geq \mu + \varepsilon\right) \leq \exp\left(-\frac{n\varepsilon^2}{2\sigma^2}\right)\,, \quad \mathbb{P}\left(\frac{1}{n}\sum_{i=1}^{n} X_i \leq \mu - \varepsilon\right) \leq \exp\left(-\frac{n\varepsilon^2}{2\sigma^2}\right)\,.$$