Online Elicitation of Necessarily Optimal Matchings

Jannik Peters

Research Group Efficient Algorithms
TU Berlin, Germany
jannik.peters@tu-berlin.de

Abstract

In this paper, we study the problem of eliciting preferences of agents in the house allocation model. For this we build on a recent model of Hosseini et al. (2021)[AAAI'21] and focus on the task of eliciting preferences to find matchings which are necessarily optimal, i.e., optimal under all possible completions of the elicited preferences. In particular, we follow the approach of Hosseini et al. (2021) and investigate the elicitation of necessarily Pareto optimal (NPO) and necessarily rank-maximal (NRM) matchings. Most importantly, we answer their open question and give an online algorithm for eliciting an NRM matching in the next-best query model which is $\frac{3}{2}$-competitive, i.e., it takes at most $\frac{3}{2}$ as many queries as an optimal algorithm. Besides this, we extend this field of research by introducing two new natural models of elicitation and by studying both the complexity of determining whether a necessarily optimal matching exists in them, and by giving online algorithms for these models.

1 Introduction

One of the key settings in the area of matching under preferences is the so-called house allocation or assignment problem. In this problem we are given two sets, a set of agents A and a set of houses H with agents having preferences over houses. This simple setting has found multiple real life applications, for instance in the allocation of people to jobs (Hylland and Zeckhauser 1979), papers to reviewers (Garg et al. 2010), or students to student dorms (Chen and Sönmez 2002). Over the years, various solution concepts have been designed for the house allocation problem, for instance *Pareto optimality* (Abdulkadiroğlu and Sönmez 1998) (Abraham et al. 2004), *popularity* (Abraham et al. 2007) or *rank-maximality* (Irving et al. 2006).

However, most of the work on house allocation problems assumes the preferences of the agents to be given in their entirety, while in many real-world applications only partial preferences might be known and eliciting complete rankings from agents might be costly.

As an expository (non-serious) example (based on a real life story), imagine a group of AI researchers meeting in their office kitchen to celebrate the acceptance of multiple papers. For this occasion, the researchers decide to eat some ice pops. However, after opening the freezer, they notice that only one ice pop of each kind is left, causing discussion on how to fairly divide the ice. Quickly, the group agrees that a rank-maximal allocation would be the fairest they could currently think of. Now there is just one problem left, due to time constraints and hunger, the researchers do not want to all give their whole ranking to each other. Instead, they agree that they should start off with naming the ice pop they like the most. But how should they continue after this, and who should be asked for their second favorite ice?

To deal with this problem Hosseini, Menon, Shah, and Sikdar (2021) initiated the study of finding matchings that are necessarily optimal by eliciting partial preferences from the agents. In their model Hosseini et al. (2021) use so-called top-k preferences in which each agent has only elicited a prefix of their true preferences. To obtain these preferences, they introduce the *next-best* query model, in which agents can be asked to reveal the top house they have not revealed yet. The goal in this setting is to ask as few queries as possible in order to find a matching that is *necessarily optimal*, i.e., optimal under every possible linear extension of the top-k preferences. The performance of such an elicitation algorithm is then measured in terms of the so-called competitive ratio, i.e., the ratio between the number of queries of the algorithm and the number of queries of an optimal algorithm with knowledge of the complete preferences. As their main results, Hosseini et al. (2021) gave an $\mathcal{O}(\sqrt{n})$-competitive elicitation algorithm for finding a necessarily Pareto optimal matching, showed that no elicitation algorithm for finding a necessarily Pareto optimal matching can be $o(\sqrt{n})$-competitive, and proved that no elicitation algorithm for finding a necessarily rank-maximal matching can be $\frac{4}{3}-\varepsilon$-competitive for any $\varepsilon > 0$. Further, they conjectured that an online algorithm with a constant competitive ratio for eliciting necessarily rank-maximal matchings is possible, and left this as their most important open question.

Our results

We contribute to this line of research in the following way. First, we confirm the conjecture of Hosseini et al. (2021) and show that an online algorithm with a $\frac{3}{2}$-competitive ratio for eliciting a necessarily rank-maximal matching does exist in the next-best query model. Further, we show that this algorithm is optimal and no online-algorithm can have a competitive ratio better than $\frac{3}{2}$. Besides this, our many focus lies on the *hybrid-query* model in which agents can be

	next-best		set-compare		hybrid-query	
	LB	UB	LB	UB	LB	UB
Pareto optimal	$\Omega(\sqrt{n})^{\dagger}$	$\mathcal{O}(\sqrt{n})^{\dagger}$	1	1	$\Omega(n^{\frac{1}{3}})$	$\mathcal{O}(n^{\frac{1}{3}+\varepsilon})$
Rank-maximal	$\frac{3}{2}$	$\frac{3}{2}$	$\frac{3}{2}$	$\mathcal{O}(n)$	$\frac{3}{2}$	6

Table 1: Overview over the lower (LB) and upper bounds (UB) on the possible competitiveness of online algorithms derived for eliciting Pareto optimal and rank-maximal matchings in the different query models. Results marked with † were shown by Hosseini et al. (2021).

asked to either elicit a house of a given rank or to return the rank of a given house. In this model, we give an online algorithm with a constant competitive ratio for eliciting a necessarily rank-maximal matching, as well as, for any $\varepsilon > 0$, an $\mathcal{O}(n^{\frac{1}{3}+\varepsilon})$-competitive algorithm for eliciting a necessarily Pareto optimal matching which almost meets the lower bound of $\Omega(n^{\frac{1}{3}})$. To add on to this, we also give a polynomial time algorithm for determining whether an NRM matching exists and show that the same problem becomes NP-complete for Pareto optimal matchings.

Finally, we also introduce the *set-compare* model in which agents can be asked to give their top-choice element out of a set. Here, we show that this model is already powerful enough to obtain a 1-competitive elicitation algorithm for Pareto optimal matchings. We show that determining whether an NPO matching under this preference model exists, is NP-complete as well, and that the $\frac{3}{2}$ lower bound obtained for the next-best model is also valid for the set-compare model. For a brief overview over the competitiveness bounds derived in this paper, we refer the reader to Table 1.

Related Work

The house allocation or assignment problem is one of the key matching settings in both computer science and economics. Besides the aforementioned classical works of for instance (Bogomolnaia and Moulin 2001; Hylland and Zeckhauser 1979; Shapley and Scarf 1974) recent papers on house allocation include work on envy-free house allocation (Gan, Suksompong, and Voudouris 2019; Beynier et al. 2019) on diversity constrains (Benabbou et al. 2018), incorporating cardinal queries in ordinal preferences (Ma, Menon, and Larson 2021) or closely related to us on Pareto optimal house allocation under probabilistic uncertainty (Aziz, Biró, de Haan, and Rastegari 2019).

Rank-maximal matchings were first introduced by Irving (2003) and were subsequently studied and characterized by Irving et al. (2006). Following these two initial papers, several works studied algorithmic aspects of various variants of the problem (Kavitha and Shah 2006; Michail 2007; Paluch 2013; Ghosal, Nasre, and Nimbhorkar 2019; Nasre, Nimbhorkar, and Pulath 2019). Besides this Belahcène, Mousseau, and Wilczynski (2021) studied rankmaximality (and popularity) in a variant of the house allocation problem, where not only the allocation of the houses, but also the selection of the allocated houses, i.e., which

houses are matched and which are unmatched, matter. Very recently Aziz and Sun (2021) used rank-maximality and algorithmic techniques of Irving et al. (2006) for the school choice problem with diversity constraints.

Besides the aforementioned works by Hosseini et al. (2021); Aziz et al. (2019) uncertainty in matching markets has been incorporated in several papers in the literature on two-sided matchings for instance by Rastegari et al. (2013); Liu et al. (2014). Besides this, Drummond and Boutilier (2014) studied preference elicitation for the stable matching problem or (Genc et al. 2017; Mai and Vazirani 2018; Chen, Skowron, and Sorge 2019) who studied stable matchings under various aspects of robustness, e.g., stable under probabilistic perturbations or after a certain number of swaps in the input rankings. Finally, very closely related to our work is also the study of possible and necessary winners in computational social choice, where given partial preferences of voters, a candidate winning every election or some election is required. See (Lang 2020) for a recent survey on this topic.

2 Preliminaries

For $a, b \in \mathbb{N}$ let $[a, b] = \{a, a+1, \ldots, b\}$ and $[a] = [1, a]$.

Throughout the paper, we let $A = \{a_1, \ldots, a_n\}$ denote our set of *agents* and $H = \{h_1, \ldots, h_n\}$ our set of *houses*. A *matching* in our setting is simply a subset $M \subseteq A \times H$ such that no agent and no house appear in more than one pair. If $(a_i, h_j) \in M$ for some agent $a_i \in A$ and house $h_j \in H$ we say that a_i is matched to h_j.

Further, we assume that each agent $a_i \in A$ has a strict *preference list* $\succ_i$ over all houses in H. If $h_j \succ_i h_k$ for two houses h_j and h_k we say that a_i *prefers* h_j to h_k. When h_j appears in the kth place in the preference list of a_i we say that the *rank* of h_j in the preference list of a_i is k and write $\text{rank}(a_i, h_j) = k$. For a given subset $H' \subseteq H$ and agent a_i, we call $\max_i(H')$ the maximum element of H' with regard to $\succ_i$, i.e., the house in H' which a_i likes the most. We refer to the collection of preference lists as a *preference profile* $\succ$.

We are now ready to define the two problems we investigate in our paper.

Pareto optimality. We begin with the classical notion of Pareto optimal matchings (Abdulkadiroğlu and Sönmez 1998). Given a matching M, we say that another matching M' *dominates* M if

- for every agent $a_i \in A$ it holds that $M'(a_i) \succ_i M(a_i)$ or $M'(a_i) = M(a_i)$;
- for at least one agent $a_i \in A$ it holds that $M'(a_i) \succ_i M(a_i)$.

Now, a matching M is *Pareto optimal* if there is no matching M' which dominates M.

Rank-maximality. As our second optimality concept we consider rank-maximality. For a given matching M let $r_l^M := |\{a_i \in A \mid \text{rank}(a_i, M(a_i)) = l\}|$ for any $l \in [n]$. Now a matching is *rank-maximal* if and only if there is no other matching M' and $l \in [n]$ such that $r_k^M = r_k^{M'}$ for all $k \in [l-1]$ and $r_l^M < r_l^{M'}$, i.e., the vector $(r_1^M, \ldots, r_n^M)$ is lexicographically maximal among all matchings. If such

a matching M' were to exist, we also say that M' *rank-dominates* M. Thus, a matching M is rank-maximal if it first maximizes the number of agents matched to their first choice, subject to that maximizes the number of agents matched to their second choice and so on.

Elicitation Protocols. Now, we turn to the three different elicitation protocols we study in our work. For the definition of the models, we assume that we are given a fixed instance of the house allocation problem with preference profile $\succ$.

- First, we investigate the *next-best* query model as defined by Hosseini et al. (2021). In this model, we are only allowed to ask one type of query. Namely, we can query a single agent, who will return the house they rank the highest, which has not been revealed yet, i.e., if this is the kth query asked to the agent, the query returns the house ranked kth by the agent in $\succ$. For any agent $a \in A$ we denote such a query as $\mathcal{Q}(a)$ and we refer to the set of agents revealed to by A as $\mathrm{rev}(a)$.

- Next we study the *hybrid-query* model. Here we can ask two types of queries. Firstly, a *rank query* $\mathcal{Q}(a_i, k)$ for an agent $a_i \in A$ and $k \leq n$ returns the house $h_j \in H$ with $\mathrm{rank}(a_i, h_j) = k$ and secondly a *house query* $\mathcal{Q}(a_i, h_j)$ for an $a_i \in A$ and a house $h_j \in H$ returns $\mathrm{rank}(a_i, h_j)$. Similarly to the next-best query model, for any $a \in A$ we refer to $\mathrm{rev}(a)$ as the set of houses $h \in H$ for which we know $\mathrm{rank}(a, h)$.

- As our third model, we study a less restricted version of the next-best query model, which we call the *set-compare* query model. Here a query $\mathcal{Q}(a_i, H')$ consists of an agent $a_i \in A$ and a subset of houses $H' \subseteq H$ and returns $\max_i(H')$, i.e., the house a_i likes best in H'. This model is inspired by recent works of learning rankings in the area of machine learning (Chen, Li, and Mao 2018; Ren, Liu, and Shroff 2019; Saha and Gopalan 2019, 2020).

For any of the three aforementioned query models, let $\mathcal{Q}_1, \ldots, \mathcal{Q}_k$ be a sequence of queries with answers $\alpha_1, \ldots, \alpha_k$ (we also refer to this as partial preferences throughout the paper). We call a preference profile $\succ$ consistent with these queries if the output of these queries on $\succ$ would be $\alpha_1, \ldots, \alpha_k$ as well. A matching M is now *necessarily Pareto optimal(NPO)* (*necessarily rank-maximal(NRM)*) for a given sequence of queries if M is Pareto optimal (rank-maximal) for all preference profiles consistent with these queries. The goal is now to design an online-algorithm which can ask queries according to one of the three aforementioned models and outputs a matching that is either necessarily Pareto optimal or necessarily rank-maximal according to the queries the algorithm asked. We assume that the online-algorithm only has access to the agents, houses and the answers to the queries, but not to the underlying preferences themselves.

In order to compare the performance of these algorithms, we measure their competitive ratio in comparison to an optimal algorithm which also knows the underlying preference profile. For any instance, such an optimal algorithm asks the minimum number of queries, after which a necessarily optimal matching with regard to these queries, asked by the optimal algorithm, can be given. We call an online algorithm

α-competitive if for any preference profile $\succ$ the online algorithm asks at most $\alpha \cdot \mathrm{OPT}_\succ$ queries, where $\mathrm{OPT}_\succ$ is the number of queries of the optimal algorithm on this instance.

Finally, we note that partial preferences in the next-best query model can be equivalently expressed by an incomplete preference profile $\succ'$ (with induced rank function rank'), in the hybrid-query model, by an incomplete rank function rank' (with induced partial preference profile $\succ'$) in which each agent only lists ranks for a subset of houses, and in the set-compare model, by having a partial order $\succ'_i$ for each agent a_i.

3 Pareto-optimal Matchings

We start off with Pareto-optimal matchings. Here, Hosseini et al. (2021) managed to give an asymptotically tight $\mathcal{O}(\sqrt{n})$-competitive elicitation algorithm for the next-best query model. As our main results, we first give a 1-competitive algorithm for eliciting NPO matchings in the set-compare model, followed by a classification of NPO matchings in the hybrid-query model together with an $\mathcal{O}(n^{\frac{1}{3}+\varepsilon})$-competitive elicitation algorithm for any $\varepsilon > 0$. Before we turn to our elicitation algorithms, we quickly recap the famous serial dictatorship mechanism (Abdulkadiroğlu and Sönmez 1998) and its relation to Pareto optimal matchings.

Definition 1 (Serial Dictatorship Mechanism)**.** *The serial dictatorship mechanism takes as input a permutation σ of the agents together with a preference profile $\succ$ and returns a matching $SD_\succ(\sigma)$ which iteratively matches agent $\sigma(i)$ to their most preferred house in $\succ$ not matched to by an agent in $\sigma(1), \ldots, \sigma(i-1)$.*

As shown by Abdulkadiroğlu and Sönmez (1998) the serial dictatorship mechanism is already enough to classify all Pareto optimal matchings.

Theorem 1 (Abdulkadiroğlu and Sönmez (1998))**.** *Given a preference profile $\succ$ a matching M is Pareto optimal if and only if there is a permutation of the agents σ such that $M = SD_\succ(\sigma)$.*

This immediately brings us to the set-compare query model where we can show that this model is already sufficient to simulate the serial dictatorship mechanism, which allows us to construct a 1-competitive algorithm.

Theorem 2. *There exists a 1-competitive algorithm in the set-compare model for computing a necessarily Pareto-optimal matching.*

Proof. Our algorithm is a simple adaption of the serial dictatorship mechanism to the set-compare model. It works iteratively by constructing a matching M. In iteration i let H_i be the set of houses already matched by M in previous iterations. Then in iteration i we query $h := \mathcal{Q}(a_i, H \setminus H_i)$ and add (a_i, h) to M. Note that we do not need to query in iteration n since only one agent/house pair is left. It is easy to see that for all possible preference extensions, in iteration i agent a_i is matched to the currently unmatched house they prefer the most. Therefore, this algorithm simulates the

serial dictatorship mechanism and thus produces a (necessarily) Pareto optimal matching. Furthermore, the algorithm only uses $n-1$ queries and is therefore 1-competitive, since at most 1 agent can be left unqueried by the optimal algorithm. □

We further note that this proof can also be generalized to the setting where the set H in each query can contain at most k houses, e.g., if $k=2$ this would mean that only pairwise comparisons could be asked to the agents. For a proof sketch, we refer to the full version (Peters 2021). For the hybrid-query model, we start off with the complexity of determining whether an NPO matching exists. While it is still polynomial time checkable if a given matching is NPO, we also show that it is NP-complete to determine the existence of an NPO matching.

Theorem 3. *Given a matching M and partial preferences* rank' *in the hybrid-query model, it can be determined in polynomial time whether M is necessarily Pareto optimal.*

Proof. This follows fairly simply by adapting the algorithm of Hosseini et al. (2021) for determining whether a matching M is NPO in the next-best model. We create an auxiliary directed graph $G = (A, E)$ in which we add an arc from agent a_i to agent a_j if it is possible for a_i to prefer $M(a_j)$ to $M(a_i)$. Then a cycle in G implies that a matching dominating M exists in a preference extension of rank'. To be more precise, we add an edge from a_i to a_j if $M(a_i), M(a_j) \in \mathrm{rev}(i)$ and $\mathrm{rank}'(a_i, M(a_j)) < \mathrm{rank}'(a_i, M(a_i))$; or if $M(a_i) \in \mathrm{rev}(i), M(a_j) \notin \mathrm{rev}(i)$ and there is a rank $k < \mathrm{rank}(a_i, M(a_i))$ with no revealed house for a_i; or if $M(a_i) \notin \mathrm{rev}(i), M(a_j) \in \mathrm{rev}(i)$ and there is a rank $k > \mathrm{rank}(a_i, M(a_j))$ with no revealed house for a_i; or if $M(a_i), M(a_j) \notin \mathrm{rev}(i)$. It is easy to see that there is a preference profile consistent with the partial preferences in which a_i prefers $M(a_j)$ to $M(a_i)$ if and only if there is an edge from a_i to a_j. Thus, a cycle in G would indeed imply that we could extend the preferences in such a way, that we could construct a matching M' dominating M, by swapping the houses along the cycle. On the other hand, if a matching M' dominates M in some preference extension, there has to be a cycle of agents $a_1, \ldots, a_k, a_{k+1} = a_1$ with agent a_i preferring $M(a_{i+1})$ to $M(a_i)$ for this preference extension and thus also forming a cycle in G. □

This algorithm also translates into an algorithm for determining whether a matching M is necessarily Pareto optimal in the set-compare model, again by adding edges from one agent to another, if there is any extension where one agent could prefer the house of the other agent.

Corollary 1. *Given a matching M and partial preferences* $\succ'$ *in the set-compare model it can be determined in polynomial time whether M is necessarily Pareto optimal.*

Using the simple algorithm in Theorem 3 we can also give a succinct classification of necessarily Pareto optimal matchings in the hybrid-query model using the Serial Dictatorship mechanism.

Lemma 1. *Given partial preferences* rank' *in the hybrid-query model a matching M is necessarily Pareto optimal if and only if there is a permutation σ of A such that for all possible preference extensions $\succ$ of* rank' *it holds that $M = SD_\succ(\sigma)$.*

Proof. Let M be an NPO matching and G the graph constructed in Theorem 3 for M. Since M is NPO the graph G is acyclic. Therefore, there exists a topological ordering of G. Let σ be a reversed topological ordering of G. Then for every $i \in [n]$ and every possible preference extension (and thus also in $\succ$), the agent $\sigma(i)$ could only possibly prefer the houses already matched to the agents $\sigma(1), \ldots, \sigma(i-1)$ if they were assigned their partner in M. Therefore, $SD_\succ(\sigma)$ would set $SD_\succ(\sigma)(\sigma(i)) = M(\sigma(i))$. Thus, by induction, we get that $SD_\succ(\sigma) = M$. □

This however does not translate to an algorithm for finding an NPO matching or determining that one exists. To show the NP-completeness of this problem we reduce from the NP-complete $(2,2)$-E3-SAT problem, (Berman, Karpinski, and Scott 2004). In an instance of the $(2,2)$-E3-SAT problem we are given a set of variables X and a set of clauses $\mathcal{C}$ over X with each clause in $\mathcal{C}$ having length exactly 3 such that each variable in X appears exactly twice in negated form and twice in positive form in $\mathcal{C}$.

Theorem 4. *Given partial preferences* rank' *in the hybrid-query model it is* NP*-complete to determine whether an NPO matching exists.*

The proof of this theorem and all further missing proofs are in the full version of the paper (Peters 2021). Using this result, we can also show that the same problem is NP-complete in the set-compare model. For this we simply show how to, given an instance in the hybrid-query model, construct an instance in the set-compare model, such that a matching is NPO in the former model if and only if is also NPO in the latter.

Theorem 5. *Given partial preferences* $\succ'$ *in the set-compare model, it is* NP*-complete to determine whether a necessarily Pareto-optimal matching exists.*

Even though it is inherently hard to find an NPO matching given partial preferences in the hybrid query model, we can still give an elicitation algorithm improving upon the competitive ratio for the next-best query model. Before proving this, we give a useful lower bound on the number of queries asked to an agent by using the serial dictatorship characterization of NPO matchings.

Lemma 2. *Let* rank' *be partial preferences and M be an NPO matching in the hybrid-query model. Then there exists a permutation of agents σ such that for all $i \in [n]$ agent $\sigma(i)$ has revealed at least $\min(\mathrm{rank}'(\sigma(i), M(\sigma(i))), n-i)$ of their preference list if $M(\sigma(i)) \in \mathrm{rev}(\sigma(i))$. If $M(\sigma(i)) \notin \mathrm{rev}(\sigma(i))$ the agent must have revealed at least $n-i$ houses.*

Proof. By Lemma 1 we know that there has to be a permutation of agents σ such that $M = SD_\succ(\sigma)$ for all possible preference extensions $\succ$ of rank'. Then since for every

Algorithm 1: Elicitation algorithm for Pareto optimal matchings in the hybrid query model

---

Input: Set of agents A, set of houses H, parameter $c_0 > \frac{1}{3}$.
Output: A necessarily Pareto optimal matching.

1: set $E \leftarrow \emptyset, M \leftarrow \emptyset, j \leftarrow 0$
2: **for** $i = 1, \ldots, n$ **do**
3:     **for all** $a \in A$ **do**
4:         $E \leftarrow E \cup \{\{a, \mathcal{Q}(a,i)\}\}$
5:     **end for**
6:     $M \leftarrow$ maximum size Pareto-optimal matching in $(A \cup H, E)$
7:     **if** $i = \lceil n^{c_j} \rceil$ **then**
8:         **if** $|M| \geq n - \lceil n^{(c_j+1)/2} \rceil$ **then**
9:             break for loop
10:         **else**
11:             $j \leftarrow j + 1, c_j \leftarrow (3c_{j-1} + 1)/2 + c_0 - 1$
12:         **end if**
13:     **end if**
14: **end for**
15: $H' \subseteq H \leftarrow$ subset matched by M
16: $A' \subseteq A \leftarrow$ subset matched by M
17: **for all** $a \in A \setminus A'$ **do**
18:     $h \leftarrow \arg\min_{h \in H \setminus H'} \mathcal{Q}(a,h)$
19:     $M \leftarrow M \cup \{\{a,h\}\}$
20:     $H' \leftarrow H' \cup \{h\}$
21: **end for**

---

preference extension $SD_{\succ}(\sigma)$ matched $\sigma(i)$ to $M(\sigma(i))$ we know that all houses matched to $\sigma(i+1), \ldots, \sigma(n)$ must be ranked lower than $M(\sigma(i))$ in all preference extensions. Thus, we either know the preferences of all houses matched to $\sigma(i+1), \ldots, \sigma(n)$ and have thus revealed at least $n - i$ of the preference list of $\sigma(i)$ or there is at least one house matched to $\sigma(i+1), \ldots, \sigma(n)$ we do not know the preference of. Then we must have queried the preferences of all houses ranked at least as high as $M(\sigma(i))$ thus requiring us to reveal at least $\operatorname{rank}'(\sigma(i), M(\sigma(i)))$ houses in the preference list of $\sigma(i)$. $\qquad\square$

The crucial idea behind our Algorithm 1 for eliciting an NPO matching is now as follows. Assume that we want to construct an algorithm with competitive ratio n^c for some constant $c > 0$ and that we know that at least k agents must be matched to a house of at least rank r with $k \geq r$, for instance by knowing that the maximum matching in the top-r preferences has size $n-k$. Then by Lemma 2 we know that at least $\frac{k}{2}$ agents must be asked at least $\min(r, \frac{k}{2})$ queries, since at least $\frac{k}{2}$ of the agents matched to a house with a rank of at least r must be listed between r and $n - \frac{k}{2}$ by σ. Thus, we know that any optimal algorithm must ask at least $\Omega(kr)$ queries which allows our online algorithm to ask $\mathcal{O}(n^c kr)$ queries. The trick behind Algorithm 1 is now to choose these values of $c, k,$ and r appropriately. To give some further idea behind the value of $\frac{1}{3}$ we show that $c_0 > \frac{1}{3}$ implies that the (c_j) series in Algorithm 1 is increasing.

Lemma 3. *The series $(c_j)_{j \in \mathbb{N}}$ with $c_{j+1} = (3c_j + 1)/2 + c_0 - 1$ is increasing if $c_0 > \frac{1}{3}$.*

Using this Lemma, we can now turn to the correctness of Algorithm 1.

Theorem 6. *For any $c > \frac{1}{3}$ Algorithm 1 is an $\mathcal{O}(n^{c_0})$-competitive algorithm for eliciting a necessarily Pareto optimal matching in the hybrid-query model.*

As an easy corollary, this implies that for any $\varepsilon > 0$, we can reach a competitive ratio of $\mathcal{O}(n^{\frac{1}{3} + \varepsilon})$. Further, we can also show that this competitive ratio is almost asymptotically optimal by showing that no online algorithm can be $o(n^{\frac{1}{3}})$-competitive.

Theorem 7. *There is no online algorithm in the hybrid-query model for computing a necessarily Pareto optimal matching with a competitive ratio of $o(n^{\frac{1}{3}})$.*

This of course still leaves the possibility of an online algorithm with a competitive ratio of $\Theta(n^{\frac{1}{3}})$.

4 Rank-Maximal Matchings

In this section, we turn to the problem of eliciting rank-maximal matchings. In their paper, Hosseini et al. (2021) showed that no online algorithm for the rank-maximal matching problem in the next-best query setting can be better than $\frac{4}{3}$ competitive. Here, we give an algorithm that is $\frac{3}{2}$-competitive and improve the lower bound of Hosseini et al. (2021) to $\frac{3}{2}$ as well, thus showing that the competitive ratio of our algorithm is tight. However, before defining this algorithm, we first need to recap some results from the classical work of Irving et al. (2006). First, we recall the definition of the so-called Dulmage–Mendelsohn decomposition.

Theorem 8 (Irving et al. (2006), Manlove (2013)). *Given a bipartite graph $G = (V, E)$ there exists a partition of V into three sets $\mathcal{O}, \mathcal{E}, \mathcal{U}$ such that*

- *Any maximum matching only contains edges between $\mathcal{U}$ and $\mathcal{U}$ as well as between $\mathcal{E}$ and $\mathcal{O}$.*
- *Any maximum matching matches every vertex in $\mathcal{U}$ and in $\mathcal{O}$.*
- *The cardinality of a maximum matching is $|\mathcal{O}| + \frac{1}{2}|\mathcal{U}|$.*
- *There is no edge between $\mathcal{E}$ and $\mathcal{E}$ as well as between $\mathcal{E}$ and $\mathcal{U}$.*

Further, it is shown by Irving et al. (2006) that $\mathcal{E}$ is the set of vertices which can reach an unmatched vertex in any maximum matching with an alternating path of even length, vertices in $\mathcal{O}$ can reach an unmatched vertex with an alternating path of odd length, and vertices in $\mathcal{U}$ cannot reach any unmatched vertex using an alternating path. Using the Dulmage–Mendelsohn decomposition Irving et al. (2006) defined the following iterative algorithm for finding a rank-maximal matching. In each iteration i the algorithm maintains a graph $G_i = (A \cup H, E_i)$ and a matching M_i in G. It is initialized with $E_0 = \emptyset, M_0 = \emptyset, \mathcal{O}_0 = \emptyset = \mathcal{U}_0$ and $\mathcal{E}_0 = A \cup H$. For each $i = 1, \ldots, n$ the algorithm

- adds all edges of rank i or less that have not been deleted yet to E_i and computes a maximum matching M_i in $G_i = (A \cup H, E_i)$ by augmenting M_{i-1};

Algorithm 2: Elicitation algorithm for rank-maximal matchings in the next-best model

Input: Set of agents A, set of houses H.
Output: A necessarily rank-maximal matching
1: $U \leftarrow A$ {set of unfinished agents}
2: $V \leftarrow H$ {set of available houses}
3: $E \leftarrow \emptyset$, $M \leftarrow \emptyset$, $F \leftarrow \emptyset$
4: **if** $|A| = 2$ **then**
5:     let $h_1 = \mathcal{Q}(a_1)$, return $\{\{a_1, h_1\}, \{a_2, h_2\}\}$
6: **end if**
7: **for** i in $1, \ldots n - 1$ **do**
8:     **for all** $a \in U$ **do**
9:         $h \leftarrow \mathcal{Q}(a)$
10:         **if** $h \in V$ and $\{a, h\} \notin F$ **then**
11:             $E \leftarrow E \cup \{\{a, h\}\}$
12:         **end if**
13:     **end for**
14:     augment M to be a maximum matching in $(A \cup H, E)$
15:     compute Dulmage-Mendelsohn decomposition $\mathcal{U}, \mathcal{E}, \mathcal{O}$ for M
16:     If an agent $a \in A$ is in $\mathcal{U}$ or $\mathcal{O}$ remove a from U
17:     If a house $h \in H$ is in $\mathcal{U}$ or $\mathcal{O}$ remove h from V
18:     Add any edges between $\mathcal{O}, \mathcal{O}$ and $\mathcal{O}, \mathcal{U}$ to F and remove them from E
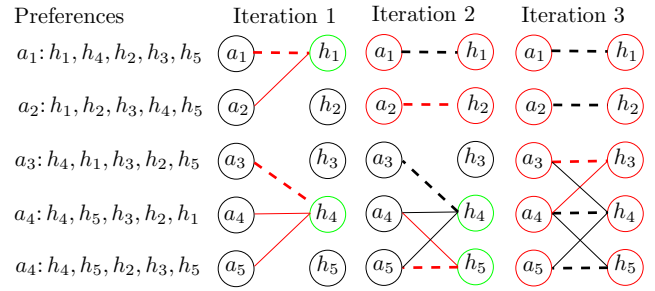19: **end for**
20: return M

Figure 1: Exemplary run of Algorithm 2. The edges displayed are the edges in E at the end of each iteration. Vertices in black are in $\mathcal{E}$, vertices in green are in $\mathcal{O}$ and vertices in red are in $\mathcal{U}$. Edges in red are the edges added in each iteration and dashed edges are matched by M in each iteration(Of course other matchings would also be valid).

- computes a Dulmage–Mendelsohn decomposition $\mathcal{U}_i, \mathcal{E}_i, \mathcal{O}_i$ for G_i
- deletes all edges incident to a node in $\mathcal{U}_i$ and $\mathcal{O}_i$ with a rank greater or equal to i, as well as all edges connecting either two nodes in $\mathcal{O}_i$ or a node in $\mathcal{O}_i$ with a node in $\mathcal{U}_i$;

The key observations of Irving et al. (2006) are now that

Lemma 4. • *Every rank-maximal matching in the instance restricted to top-k preferences is a maximum matching in G_k.*
- *Every M_k is a rank-maximal matching in the instance restricted to top-k preferences.*

Using these observations, we can now simulate the algorithm of Irving et al. (2006) by tracking the set of forbidden edges F, i.e., the set of edges deleted in the third step of (Irving et al. 2006) in each iteration, and by not asking any queries to an agent who has been in $\mathcal{U}_i$ or $\mathcal{O}_i$ for any $i \in [n]$. Please refer to Figure 1 for an example of Algorithm 2 being executed. To get a good bound on the competitive ratio of Algorithm 2 we observe a simple lemma based on the Dulmage-Mendelsohn decomposition and its relation to preferences of agents in a necessarily rank-maximal matching. For a given house allocation instance and agent $a \in A$ let $r_a := \min_{i \in [n]} a \notin \mathcal{E}_i$. It is easy to see that for any instance, our algorithm asks exactly r_a queries to a. We can now show that the optimal algorithm must ask at least $r_a - 1$ queries to agent a if this agent is matched to a preference the agent has revealed.

Lemma 5. *Given a house allocation instance $(A, H, \succ)$ partial preferences $\succ'$ in the next-best query model and nec-*

essarily rank-maximal matching M for $\succ'$ it has to hold that $|\text{rev}(a)| \geq r_a - 1$ for all agents matched to a revealed preference by M.

Proof. Towards a contradiction, we assume that there is some agent $a \in A$ with $M(a) \in \text{rev}(a)$ and $|\text{rev}(a)| < r_a - 1$. Then since M is rank-maximal in all completions of $\succ'$ and thus also in $\succ$ we know that $a \in \mathcal{E}_{r_a - 1}$. Hence, there has to be an alternating path of even length $\rho := a = a_1, M(a_1), a_2, \ldots, M(a_{k-1}), a_k$ in $G_{r_a - 1}$ from a to an unmatched agent a_k. Since a_k is unmatched in $G_{r_a - 1}$ by M we know that $\text{rank}(a_k, M(a_k)) > r_a - 1$ in $\succ$.

Further, assume that there is some a_i with $i > 1$ such that $\text{rank}(a_i, M(a_{i-1})) > \text{rank}(a_{i-1}, M(a_{i-1}))$. Then, since a_{i-1} must be in $\mathcal{E}_j$ for all $j \in [r_a - 1]$ due to ρ, we know that $M(a_{i-1})$ must be in $\mathcal{O}_{\text{rank}(a_{i-1}, M(a_{i-1}))}$ which in turn implies that $\{a_i, M(a_{i-1})\}$ is not an edge in $G_{r_a - 1}$. Thus, $\text{rank}(a_i, M(a_{i-1})) \leq \text{rank}(a_{i-1}, M(a_{i-1}))$ has to hold in $\succ$. Further, we can extend $\succ'$ in such a way that $\text{rank}(a_1, M(a_k)) \leq r_a - 1$, while keeping all other preferences according to $\succ$. For these preferences, we can augment M with the path

$$a_1, M(a_k), a_k, M(a_{k-1}), a_{k-1}, \ldots a_2, M(a_1)$$

and get a matching that rank-dominates M' since $\text{rank}(a_i, M(a_{i-1})) \leq \text{rank}(a_{i-1}, M(a_{i-1}))$, as well as $\text{rank}(a_1, M(a_k)) \leq r_a - 1$, while previously $\text{rank}(a_k, M(a_k)) > r_a - 1$. Therefore, no agent $a \in A$ matched to a revealed preference can have less than $r_a - 1$ of their preference revealed. $\square$

This simple lemma is in fact already sufficient to get a constant upper bound of at most 3 on the competitive ratio of Algorithm 2. To see this, consider the following. We know from Lemma 5 and the fact that at most one agent can be matched to an unrevealed preference and that all but one agent are asked at least $\min(r_a - 1, 1)$ queries by the optimal algorithm and exactly r_a queries by our algorithm. Further, the agent matched to the unrevealed preference is asked at most $n - 1$ queries by our algorithm. Let A' be the
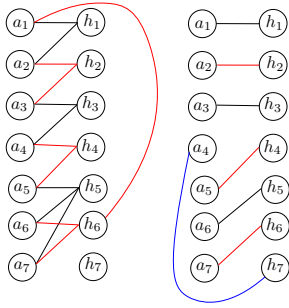
Figure 2: Construction of Theorem 10 for $k = 3$ and special agent a_4. The basic preferences for ranks 1 and 2 are shown on the left, with rank 1 edges in black and rank 2 edges in red and the edge of the special agent in blue. The matching on the right, is the rank-maximal matching.

set of agents matched to revealed preferences. Then, we can bound the competitive ratio by

$$\frac{\sum_{a \in A'}(r_a) + n - 1}{\sum_{a \in A'}(\min(r_a - 1, 1))} \leq \frac{\sum_{a \in A'}(r_a + 1)}{\sum_{a \in A'}(\min(r_a - 1, 1))}.$$

Since for any $a \in A'$ the term $\frac{r_a+1}{\min(r_a-1,1)}$ is at most 3, this implies the upper bound of 3 on the competitive ratio. However, we can improve upon this and can even show that the algorithm is $\frac{3}{2}$-competitive.

Theorem 9. *Algorithm 2 is a $\frac{3}{2}$-competitive algorithm for eliciting a rank-maximal matching in the next-best query model.*

Further, we can show that this bound is tight (up to subconstant factors) by showing that for any $\varepsilon > 0$, no online algorithm can have a competitive ratio better than $\frac{3}{2} - \varepsilon$.

Theorem 10. *No online algorithm in the next-best query model can achieve a competitive ratio better than $\frac{3}{2} - \varepsilon$ for any $\varepsilon > 0$.*

Proof. Let $k \geq 2$ be an integer and consider the following instance with $n = 2k + 1$. We start off with the *basic preferences* of the agents. Here for any $i \in [k]$ (with the assumption that $1 - 1 = k$), we assume that the basic preference of agent a_{2i-1} is $h_{2i-1} \succ h_{2(i-1)} \succ \cdots \succ h_{2k+1}$ and the basic preference of agent a_{2i} is $h_{2i-1} \succ h_{2i} \succ \cdots \succ h_{2k+1}$ with the preferences between the second and the last house being arbitrary. Further, the basic preference of agent a_{2k+1} is $h_{2k-1} \succ h_{2k} \succ \cdots \succ h_{2k+1}$ (just like for agent a_{2k}). For our adversarial instance, we assume that all but one agent have their basic preference, with the one *special agent* instead listing h_{2k+1} as their third preference. It is easy to see that in such an instance, a rank-maximal matching must match k agents to their first choice, k agents to their second choice and the special agent to h_{2k+1}, since the special agent is the only one not listing h_{2k+1} last. The existence of such a matching easily follows by matching the special agent a_i to h_{2k+1}, any agent a_j with $j < i$ to h_j and any agent a_j with $j > i$ to h_{j-1}. This is a valid matching and matches exactly k agents to their first choice, namely all agents with an odd

index that is smaller than i and all agents with an even index that is larger than i, k to their second choice and a_i to their third choice, thus being rank-maximal. For an example of this construction, we refer to Figure 2.

Hence, the optimal algorithm can ask 2 queries to each non-special agent and 3 queries to the special agent and can thus elicit an NRM matching. An adversary on the other hand can simply reveal other houses than h_{2k+1} when asked for the third house of any agent until the last agent is asked. Thus, any online algorithm needs to ask every agent at least 3 queries.

Therefore, the competitive ratio of any online algorithm has to be at least $\frac{3(2k+1)}{2(2k+1)+1} = \frac{3}{2} - \frac{3}{8k+6}$ and thus we get that no online algorithm can be $\frac{3}{2} - \varepsilon$-competitive for any $\varepsilon > 0$. $\qquad \square$

With some slight adjustments to the adversary, the same construction also works for the hybrid-query and set-compare model, thus also implying a lower bound of $\frac{3}{2}$ in both models.

Corollary 2. *In both the set-compare and hybrid query models, there is no online algorithm with a competitive ratio of $\frac{3}{2} - \varepsilon$ for any $\varepsilon > 0$.*

We also construct an algorithm, which decides in polynomial time whether a necessarily rank-maximal matching exists, thus standing in contrast to the NP-hardness of the same decision problem for Pareto optimal matchings.

Theorem 11. *Given partial preferences rank' in the hybrid-query model, it can be decided in polynomial time whether a necessarily rank-maximal matching exists and whether a given matching M is necessarily rank-maximal.*

Finally, we modify Algorithm 2 to get an algorithm with a constant competitive ratio in the hybrid-query model.

Theorem 12. *There exists a 6-competitive algorithm for eliciting a rank-maximal matching in the hybrid query model.*

However, we were not able to find an online algorithm for the set-compare setting achieving a sublinear competitive ratio for eliciting a rank-maximal matching.

5   Discussion

There are multiple open questions and possible future research directions which can be derived from this work.

Firstly, there are still gaps left between the upper bounds and lower bounds we showed in this paper. Most importantly, it would be very interesting to find out whether there is an algorithm with a constant (or even sublinear) competitive ratio for eliciting an NRM matching in the set-compare model. Besides this, the complexity of determining whether a matching is NRM is also open in the set-compare model.

Secondly, there are several other notions of optimality left to explore, for instance (Huang et al. 2016) fair matchings or the general class of profile-based matchings (Kwanashie et al. 2014) encompassing both fair and rank-maximal matchings. Of course, it might also be interesting to study further querying models or models of partial preferences.

6 Acknowledgments

References

Abdulkadiroğlu, A.; and Sönmez, T. 1998. Random Serial Dictatorship and the Core from Random Endowments in House Allocation Problems. *Econometrica*, 66(3): 689–701.

Abraham, D. J.; Cechlárová, K.; Manlove, D. F.; and Mehlhorn, K. 2004. Pareto Optimality in House Allocation Problems. In *16th International Symposium on Algorithms and Computation (ISAAC)*, 1163–1175.

Abraham, D. J.; Irving, R. W.; Kavitha, T.; and Mehlhorn, K. 2007. Popular Matchings. *SIAM Journal on Computing*, 37(4): 1030–1045.

Aziz, H.; Biró, P.; de Haan, R.; and Rastegari, B. 2019. Pareto Optimal Allocation under Uncertain Preferences: Uncertainty Models, Algorithms, and Complexity. *Artif. Intell.*, 276: 57–78.

Aziz, H.; and Sun, Z. 2021. Multi-Rank Smart Reserves. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, 105–124.

Belahcène, K.; Mousseau, V.; and Wilczynski, A. 2021. Combining Fairness and Optimality when Selecting and Allocating Projects. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021*, 38–44.

Benabbou, N.; Chakraborty, M.; Ho, X.-V.; Sliwinski, J.; and Zick, Y. 2018. Diversity Constraints in Public Housing Allocation. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018)*, 973–981. ACM.

Berman, P.; Karpinski, M.; and Scott, A. 2004. Approximation Hardness of Short Symmetric Instances of MAX-3SAT. Technical report.

Beynier, A.; Chevaleyre, Y.; Gourvès, L.; Harutyunyan, A.; Lesca, J.; Maudet, N.; and Wilczynski, A. 2019. Local Envy-freeness in House Allocation Problems. *Autonomous Agents and Multi-Agent Systems*, 33(5): 591–627.

Bogomolnaia, A.; and Moulin, H. 2001. A New Solution to the Random Assignment Problem. *Journal of Economic Theory*, 100(2): 295–328.

Chen, J.; Skowron, P.; and Sorge, M. 2019. Matchings under Preferences: Strength of Stability and Trade-offs. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, 41–59.

Chen, X.; Li, Y.; and Mao, J. 2018. A Nearly Instance Optimal Algorithm for Top-k Ranking Under the Multinomial Logit Model. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2504–2522. SIAM.

Chen, Y.; and Sönmez, T. 2002. Improving Efficiency Of On-campus Housing: An Experimental Study. *American economic review*, 92(5): 1669–1686.

Drummond, J.; and Boutilier, C. 2014. Preference Elicitation and Interview Minimization in Stable Matchings. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 645–653.

Gan, J.; Suksompong, W.; and Voudouris, A. A. 2019. Envy-freeness in House Allocation Problems. *Mathematical Social Sciences*, 101: 104–106.

Garg, N.; Kavitha, T.; Kumar, A.; Mehlhorn, K.; and Mestre, J. 2010. Assigning Papers to Referees. *Algorithmica*, 58: 119–136.

Genc, B.; Siala, M.; O'Sullivan, B.; and Simonin, G. 2017. Robust Stable Marriage. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 4925–4926.

Ghosal, P.; Nasre, M.; and Nimbhorkar, P. 2019. Rank-Maximal Matchings–Structure and Algorithms. *Theoretical Computer Science*, 767: 73–82.

Hosseini, H.; Menon, V.; Shah, N.; and Sikdar, S. 2021. Necessarily Optimal One-Sided Matchings. In *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence*, 6, 5481–5488.

Huang, C.-C.; Kavitha, T.; Mehlhorn, K.; and Michail, D. 2016. Fair Matchings and Related Problems. *Algorithmica*, 74(3): 1184–1203.

Hylland, A.; and Zeckhauser, R. 1979. The Efficient Allocation of Individuals to Positions. *Journal of Political economy*, 87(2): 293–314.

Irving, R. W. 2003. Greedy Matchings. *Technical Report Tr-2003-136, University of Glasgow*.

Irving, R. W.; Kavitha, T.; Mehlhorn, K.; Michail, D.; and Paluch, K. E. 2006. Rank-Maximal Matchings. *ACM Transactions on Algorithms (TALG)*, 2(4): 602–610.

Kavitha, T.; and Shah, C. D. 2006. Efficient Algorithms for Weighted Rank-Maximal Matchings and Related Problems. In *International Symposium on Algorithms and Computation*, 153–162. Springer.

Kwanashie, A.; Irving, R. W.; Manlove, D. F.; and Sng, C. T. 2014. Profile-based Optimal Matchings in the Student/Project Allocation Problem. In *International Workshop on Combinatorial Algorithms*, 213–225. Springer.

Lang, J. 2020. Collective Decision Making under Incomplete Knowledge: Possible and Necessary Solutions. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, 4885–4891.

Liu, Q.; Mailath, G. J.; Postlewaite, A.; and Samuelson, L. 2014. Stable Matching with Incomplete Information. *Econometrica*, 82(2): 541–587.

Ma, T.; Menon, V.; and Larson, K. 2021. Improving Welfare in One-sided Matching using Simple Threshold Queries. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021*, 321–327.

Mai, T.; and Vazirani, V. V. 2018. Finding Stable Matchings That Are Robust to Errors in the Input. In *26th Annual*

European Symposium on Algorithms (ESA 2018). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Manlove, D. F. 2013. *Algorithmics of Matching Under Preferences*. World Scientific Publishing Company.

Michail, D. 2007. Reducing Rank-Maximal to Maximum Weight Matching. *Theoretical Computer Science*, 389(1-2): 125–132.

Nasre, M.; Nimbhorkar, P.; and Pulath, N. 2019. Classified Rank-Maximal Matchings and Popular Matchings– Algorithms and Hardness. In *International Workshop on Graph-Theoretic Concepts in Computer Science*, 244–257. Springer.

Paluch, K. 2013. Capacitated Rank-Maximal Matchings. In *International Conference on Algorithms and Complexity*, 324–335. Springer.

Peters, J. 2021. Online Elicitation of Necessarily Optimal Matchings. Technical report, arXiv:2112.04227 [cs.GT].

Rastegari, B.; Condon, A.; Immorlica, N.; and Leyton-Brown, K. 2013. Two-sided Matching with Partial Information. In *Proceedings of the fourteenth ACM conference on Electronic Commerce*, 733–750.

Ren, W.; Liu, J. K.; and Shroff, N. 2019. On Sample Complexity Upper and Lower Bounds for Exact Ranking from Noisy Comparisons. *Advances in Neural Information Processing Systems*, 32: 10014–10024.

Saha, A.; and Gopalan, A. 2019. Combinatorial Bandits with Relative Feedback. *Advances in Neural Information Processing Systems*, 32: 985–995.

Saha, A.; and Gopalan, A. 2020. From PAC to Instance-Optimal Sample Complexity in the Plackett-Luce Model. In *International Conference on Machine Learning*, 8367–8376. PMLR.

Shapley, L. S.; and Scarf, H. 1974. On Cores and Indivisibility. *Journal of Mathematical Economics*, 1(1): 23–37.