

Matchings under One-Sided Preferences with Soft Quotas

Santhini K. A.¹, Raghu Raman Ravi² and Meghana Nasre¹

¹Indian Institute of Technology Madras

²ETH Zurich

{santhini, meghana}@cse.iitm.ac.in, raghu.ravi.raman@gmail.com

Abstract

Assigning applicants to posts in the presence of the preferences of applicants and quotas associated with posts is extensively investigated. For a post, lower quota *guarantees*, and upper quota limits the number of applicants assigned to it. Typically, quotas are assumed to be fixed, which need not be the case in practice. We address this by introducing a soft quota setting, in which every post is associated with two values – lower target and upper target which together denote a range for the intended number of applicants in any assignment. Unlike the fixed quota setting, we allow the number of applicants assigned to a post to fall outside the range. This leads to assignments with deviation. Here, we study the problem of computing an assignment that has two orthogonal optimization objectives – minimizing the deviation (maximum or total) w.r.t. soft quotas and ensuring optimality w.r.t. preferences of applicants (rank-maximality or fairness). The order in which these objectives are considered, the different possibilities to optimize deviation combined with the well-studied notions of optimality w.r.t. preferences open up a range of optimization problems of practical importance. We present efficient algorithms based on flow-networks to solve these optimization problems.

1 Introduction

We study the many-to-one bipartite matching problem where we have a bipartite graph $G = (\mathcal{A} \cup \mathcal{P}, E)$ of applicants \mathcal{A} and posts \mathcal{P} in the presence of preferences of applicants over the posts. Typically, every post $p \in \mathcal{P}$ is associated with an input upper quota $q^+(p)$ that limits the maximum number of applicants assigned to p in any allocation. Every applicant ranks all the posts in its neighborhood and this ranking can involve ties between the posts. A matching M is a collection of edges where an applicant has at most one edge incident to it in M and a post p has at most $q^+(p)$ many edges incident on it in M . In the literature, this setting is termed as house allocation setting [Abraham *et al.*, 2004] and can be used to model different problems such as assigning applicants to jobs, allocating courses to students [Biró, 2017], assigning papers

to referees [Garg *et al.*, 2010], students to schools [Abraham, 2009], and rental inventories to customers [Abraham *et al.*, 2006].

In certain applications like allocating courses or projects to students [Biró *et al.*, 2010; Arulselvan *et al.*, 2018; Cseh *et al.*, 2022], the posts may require a minimum number of applicants to be assigned in order to be operational. Lower quotas are extensively investigated in the two-sided preference list setting called the Hospital Residents problem [Goko *et al.*, 2022; Goto *et al.*, 2014; Hamada *et al.*, 2016; Fleiner and Kamiyama, 2016; Monte and Tumennasan, 2013; Nasre and Nimbhorkar, 2018; Nasre *et al.*, 2021; Fragiadakis *et al.*, 2016]. Here, the lower quota of a hospital denotes the minimum number of residents (doctors) needed for the smooth functioning of the hospital. To capture this, for a post p in addition to the upper quota $q^+(p)$, a lower-quota $q^-(p)$ is associated with it. For a matching M and a post p , $M(p)$ denotes the set of applicants assigned to p in M . In this setting, a matching M is feasible if for every post p , $q^-(p) \leq |M(p)| \leq q^+(p)$. We refer to this setting as the *fixed quota setting* throughout the paper. In the presence of lower quotas, there are simple instances where no feasible matching exists (see Figure 1 for an example).

In many practical applications, quotas need not be rigid as is assumed in the fixed quota setting and relaxing the lower quotas and augmenting the upper quotas have been considered in the literature. For instance, instead of hospitals closing down for want of a minimum number of doctors, they may simply reduce their service levels as described in [Goko *et al.*, 2022]. Similarly, in the context of school choice, capacity planning to *augment* the number of available seats has been considered recently in multiple works [Abe *et al.*, 2022; Bobbio *et al.*, 2021; Bobbio *et al.*, 2022]. Motivated by such applications, we relax the fixed quota assumption by proposing and studying a *soft quota* setting for the assignment problem in the one-sided preference list model.

2 Soft Quota Setting

An instance of the problem in the soft quota setting consists of a bipartite graph $G = (\mathcal{A} \cup \mathcal{P}, E)$ where applicants have preference over posts and a post p has two values – a lower target $\ell(p)$ and an upper target $u(p)$. For a post p , the two values $\ell(p)$ and $u(p)$ define a range for the intended number of applicants for p in any matching. There is no restriction on

the number of applicants that can be assigned to a post in any matching. However, in order to control skew, we measure the deviation of a post p w.r.t. a matching M as defined below:

$$d_M(p) = \max\{0, \ell(p) - |M(p)|, |M(p)| - u(p)\}.$$

Our goal in the soft quota setting is to compute an assignment that has two orthogonal optimization objectives:

(i) Minimizing deviation. For a matching M , we define the total deviation $d_{tot}(M)$ and max deviation $d_{max}(M)$ as below, and our goal is to minimize one of them.

$$d_{tot}(M) = \sum_{p \in \mathcal{P}} d_M(p) \quad d_{max}(M) = \max_{p \in \mathcal{P}} d_M(p)$$

(ii) Optimality w.r.t. preferences. Two well-studied notions of optimality in the fixed quota setting for one-sided preferences are rank-maximality [Irving *et al.*, 2006; Paluch, 2013; Hosseini *et al.*, 2021; Peters, 2022], and fairness [Huang *et al.*, 2016]. A rank-maximal matching matches maximum number of applicants to rank 1 post, subject to that maximum number of applicants to rank 2 posts and so on. A fair matching minimizes the number of unmatched applicants, subject to that minimizes the number of applicants matched to the last rank post and so on. The *signature* of a matching allows us to formalize these notions. Let r be the maximum rank used by an applicant to order a post in G . The signature of a matching M ($\sigma(M)$) is an $r + 1$ tuple $(x_1, x_2, \dots, x_{r+1})$ where x_i denotes the number of applicants matched to their rank i post in M for $i \in [r]$ and x_{r+1} denotes the number of unmatched applicants in M .

Definition 1 (Rank-maximality). *For two matchings M and M' with signatures $\sigma(M) = (x_1, \dots, x_{r+1})$ and $\sigma(M') = (y_1, \dots, y_{r+1})$, we say M is more rank-maximal than M' ($M \succ_{\mathcal{R}} M'$) if there exists an ℓ for $\ell \in [r]$ such that $x_\ell > y_\ell$ and for $j \in [\ell - 1]$, we have $x_j = y_j$. A matching that has a maximum signature under the ordering $\succ_{\mathcal{R}}$ is a rank-maximal matching.*

The notion of fairness can be defined analogously. In the fixed quota setting where the lower quota of every post is zero, both these notions are guaranteed to exist, and optimal matchings are efficiently computable by the algorithms given in [Irving *et al.*, 2006; Paluch, 2013; Huang *et al.*, 2016]. Even in the presence of lower quotas an optimal matching amongst all the feasible matchings can be efficiently computed by extending the algorithm in [Huang *et al.*, 2016].

2.1 Our Problems

Deviation first, preference optimality next. There exist applications where having as low deviation as possible is required and subject to that optimality with respect to preferences needs to be respected. For example, in allocating projects to students [Cooper, 2020] or dividing political representatives to seats [Dhamala and Thapa, 2009; Pukelsheim *et al.*, 2012; Serafini and Simeone, 2012] it is desirable to minimize the difference between input values and the assigned numbers. Let \mathcal{M} be the set of all matchings in the instance G of the soft quota setting. Let \mathcal{M}_t (\mathcal{M}_m) denote the set of matchings that have minimum value of total deviation (respectively the minimum value of maximum deviation). We define our first set of problems:

a_1	:	p_1	p_1	[1, 1]	a_1	—	p_1	✓
a_2	:	p_1	p_2	[1, 1]	a_2		p_3	⊗
a_3	:	p_1, p_2, p_3	p_3	[1, 1]	a_3	—	p_2	✓
a_4	:	p_1, p_6, p_7	p_4	[1, 1]	a_4	—	p_7	✓
a_5	:	p_1, p_4, p_5	p_5	[1, 1]	a_5	—	p_4	✓
a_6	:	p_6, p_7	p_6	[0, 1]	a_6	—	p_6	✓
a_7	:	p_1, p_6	p_7	[0, 1]	a_7		p_5	⊗
					M_1			
applicants			posts					

Figure 1: Example instance G with $|\mathcal{A}| = |\mathcal{P}| = 7$. Applicants along with their preferences are given. Each post has two values – a *lower value* and an *upper value*. In the fixed quota setting these numbers represent rigid lower and upper quotas. In the soft quota setting these numbers are lower and upper target respectively. Matching M_1 shown beside is an infeasible matching in the fixed quota setting but is valid assignment in the soft quota setting.

- **OPT-MIN-TOT:** The goal is to compute a matching that satisfies the notion OPT w.r.t. preferences in the set \mathcal{M}_t . For instance, if the notion OPT is rank-maximality then the problem RMM-MIN-TOT outputs a rank-maximal matching in the set \mathcal{M}_t .
- **OPT-MIN-MAX:** The goal is to compute a matching that satisfies the notion OPT in the set \mathcal{M}_m .

Preference optimality first, deviation next. Analogously, there exists applications where preference optimality has to be respected first and subject to that we would like to minimize a deviation parameter of interest. Since in the soft quota setting, there is no bound on the upper quota of a post, it is not immediate how to determine the set of matchings in which we wish to compute a rank-maximal or fair matching. To address this, we first introduce an additional input in our problem – a signature requirement ρ for the output matching. We consider all matchings in the soft quota setting that respect the signature requirement with respect to the preference optimality under consideration. This is similar to what is done in [K. A. *et al.*, 2022] where the signature is a part of the input.

For a fixed notion of preference optimality, a matching M meets the signature requirement if $\sigma(M) \geq_{OPT} \rho$ and let \mathcal{M}_{OPT} denote all the matchings that meet the signature requirement. Let \mathcal{M}'_t (\mathcal{M}'_m) denote the set of matchings that have minimum value of total deviation (minimum value of maximum deviation respectively) in \mathcal{M}_{OPT} . We consider the following two problems here:

- **OPT-SIGN-MIN-TOT:** The goal is to output any matching in \mathcal{M}'_t .
- **OPT-SIGN-MIN-MAX:** The goal is to output any matching in \mathcal{M}'_m .

In the rest of the paper, we present the results when OPT is set to rank-maximality (RMM). Analogous results for fairness can be obtained using minor modifications to our algorithms. These are deferred to the full version of the paper.

Example. We illustrate our problems using the example instance G in Figure 1. Table 1 shows four matchings along with parameters of the matching. We first consider the problems of the type deviation first, preference optimality next.

Input	Optimality notion	Matching	Parameters of matching		
			Signature	Max dev	Total dev
G	RMM-MIN-TOT	$M_1 = \{(a_1, p_1), (a_3, p_2), (a_4, p_7), (a_5, p_4), (a_6, p_6)\}$	(2, 2, 1, 2)	1	2
G	RMM-MIN-MAX	$M_2 = \{(a_1, p_1), (a_2, p_1), (a_3, p_2), (a_4, p_7), (a_5, p_4), (a_6, p_6), (a_7, p_6)\}$	(3, 3, 1, 0)	1	4
G, ρ	RMM-SIGN-MIN-TOT	$M_3 = \{(a_1, p_1), (a_2, p_1), (a_3, p_2), (a_4, p_7), (a_5, p_4), (a_6, p_6), (a_7, p_1)\}$	(4, 2, 1, 0)	2	4
G, ρ	RMM-SIGN-MIN-MAX	$M_4 = \{(a_1, p_1), (a_2, p_1), (a_3, p_2), (a_4, p_1), (a_5, p_4), (a_6, p_6), (a_7, p_6)\}$	(4, 3, 0, 0)	2	5

Table 1: Optimal matchings in the instance given in Figure 1 corresponding to our problems. For the preference optimality first type problems, we consider the input signature as (4, 0, 0, 3). We consider preference optimality as rank-maximality.

Any matching in the instance leaves one of p_2 or p_3 (similarly, one of p_4 or p_5) unmatched. Thus, any matching in G must have a maximum deviation of at least one and total deviation of at least two. It can be verified that the matchings M_1 and M_2 shown in Table 1 are optimal for the problems RMM-MIN-TOT and RMM-MIN-MAX respectively.

Now we turn to problems where preference optimality is considered first and subject to that deviation is minimized. For concreteness, we fix the requirement for signature as (4, 0, 0, 3). In order to meet the signature requirement, we must violate the upper target of the post p_1 by 2 and hence the max deviation of any matching in $\mathcal{M}_{\mathcal{R}}$ is at least 2. By the remark in the preceding paragraph, the total lower deviation of any matching in the instance is at least 2 and hence the total deviation of any matching in $\mathcal{M}_{\mathcal{R}}$ is at least 4. Furthermore, it can be verified that the matchings M_3 and M_4 shown in Table 1 are optimal for the problems RMM-SIGN-MIN-TOT and RMM-SIGN-MIN-MAX respectively.

The order in which we consider the two orthogonal parameters to optimize provides us with the trade-off between deviation and preference optimality. As expected, when the deviation is optimized first, the output matching has a *weak* signature. In contrast, when preference optimality is given precedence, a higher deviation is incurred. Thus, our problems offer a trade-off based on the user requirement.

Our problems consider optimizing two objective functions – deviation and preference optimality and hence can be regarded as a special case of a multi-objective optimization problem [Esfandiari *et al.*, 2016; Korula *et al.*, 2013; Aggarwal *et al.*, 2014]. These works differ from our work in terms of the problem setup and the objectives considered.

2.2 Our Results

We show the following new results in our paper.

Theorem 1. OPT-SIGN-MIN-MAX and OPT-SIGN-MIN-TOT admit polynomial time algorithms, where OPT can be one of rank-maximality (RMM) or fairness (FAIR).

Theorem 2. OPT-MIN-MAX and OPT-MIN-TOT admit polynomial time algorithms, where OPT can be one of rank-maximality or fairness.

Our Techniques. The algorithms for computing minimum total deviation matchings are based on a flow network that we design. Flow networks are used to compute Pareto-optimal matchings [Cechlárová *et al.*, 2016; Cechlárová and Fleiner, 2017; Cseh *et al.*, 2022], rank-maximal matchings [Nasre *et al.*, 2019] and cumulative signature matchings [K. A. *et al.*,

2022] in the literature. For computing matchings with minimum value of maximum deviation we reduce the problem to an appropriate problem in the fixed quota setting which can be solved efficiently.

2.3 Other Related Works

Biproportional apportionment. Our soft quota setting is similar to the biproportional apportionment problem [Dhamala and Thapa, 2009; Pukelsheim *et al.*, 2012; Serafini and Simeone, 2012] which deals with fair division of political representatives to states. Political representatives correspond to applicants, and states correspond to posts in our model. In our setting, we have applicant preferences which are absent in the biproportional apportionment problem.

Soft bounds. The recent works [Ehlers *et al.*, 2014; Echenique and Yenmez, 2015; Gonczarowski *et al.*, 2019; Aziz *et al.*, 2020; Kurata *et al.*, 2017] in the two-sided preferences model allow soft bounds on the capacity constraints. These works focus on the controlled school choice problem in which students are associated with types, and schools have soft bounds on the quota of each type. For more information, refer to the survey paper by [Aziz *et al.*, 2022].

Other models of flexibility. In the context of school choice problem, capacity planning is studied in [Gajulapalli *et al.*, 2020; Rios *et al.*, 2021; Bobbio *et al.*, 2021; Abe *et al.*, 2022; Chen and Csáji, 2023]. In the work of [Limaye and Nasre, 2023; K. A. *et al.*, 2022] programs have costs instead of quotas which allows flexibility. In all these works, flexibility of upper quotas is considered, whereas in our work, we allow posts to deviate on both upper and lower targets.

To the best of our knowledge, this is the first work that investigates soft quotas in the one-sided preference list model.

3 Preference Optimality First

In this section, we present the results for the preference optimality first, deviation next type of problems. As mentioned earlier, apart from the input instance these problems have a signature as a part of the input.

3.1 Algorithm for RMM-SIGN-MIN-TOT

An instance of RMM-SIGN-MIN-TOT problem consists of a graph G with applicant preferences, soft quotas of posts and an input signature ρ . Let $\mathcal{M}_{\mathcal{R}}$ denote the set of matchings where for every $M \in \mathcal{M}_{\mathcal{R}}$, we have $\sigma(M) \geq_{\mathcal{R}} \rho$. Our goal is to output a matching M that has minimum total deviation in the set $\mathcal{M}_{\mathcal{R}}$. We present a flow-network based algorithm to solve the RMM-SIGN-MIN-TOT problem.

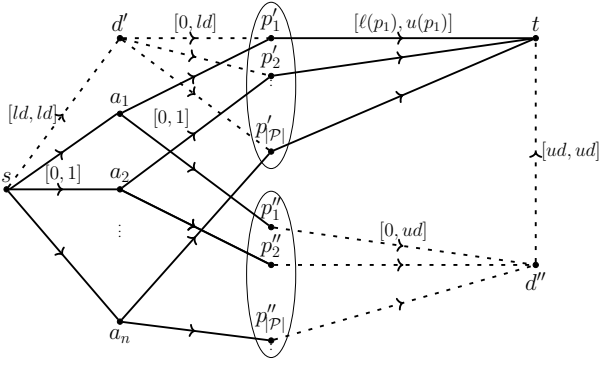


Figure 2: Flow network $H_{(ld,ud)}$ for solving RMM-SIGN-MIN-TOT, where $\Delta = ld + ud$

Overall idea. Consider the optimal matching M^* for the RMM-SIGN-MIN-TOT problem. Let the total deviation of M^* be Δ^* . In M^* , a subset of posts incur deviation for the lower target whereas a subset of posts (disjoint from the previous set) incur deviation for the upper target of the corresponding post. Let ld^* (ud^*) denote the sum of the deviation of posts that deviate w.r.t. the lower target (respectively w.r.t. the upper target). Note that $\Delta^* = ld^* + ud^*$. We show that there exists a flow network $H_{(ld^*,ud^*)}$ such that the matching M^* has a feasible flow f^* in $H_{(ld^*,ud^*)}$. We use this flow network to obtain the optimal value of total deviation.

Construction of the network. Given an instance G of RMM-SIGN-MIN-TOT, we fix value Δ which can be split as the sum of two non-negative integers ld and ud . We construct the flow network $H_{(ld,ud)}$ with costs and demands on the edges. See Figure 2 for an illustration. Vertices of $H_{(ld,ud)}$ are:

$$V(H_{(ld,ud)}) = \{s, d', d'', t\} \cup \mathcal{A} \cup \{p', p'' : p \in \mathcal{P}\}.$$

For every applicant a in G , we have a node a in the network $H_{(ld,ud)}$. For every post p in G , we have two nodes in $H_{(ld,ud)}$, that is p' (original copy) and p'' (extended copy). In addition to the above nodes, we have a source s , sink t and two special nodes d' and d'' which help us in capturing the lower deviation and upper deviation respectively.

Every edge $e \in E(H_{(ld,ud)})$ in the network has a vector $[q^-(e), q^+(e)]$ that represents the demand and capacity associated with it. The edge set $E(H_{(ld,ud)})$ is the union of the following sets:

- $\mathcal{E}_1 = \{(s, a) : a \in \mathcal{A}\}$ with vector $[0, 1]$
- $\mathcal{E}_2 = \{(a, p'), (a, p'') : (a, p) \in E\}$ with vector $[0, 1]$
- $\mathcal{E}_3 = \{(p', t) : p \in \mathcal{P}\}$ with vector $[\ell(p), u(p)]$
- $\mathcal{E}_4 = \{(p'', d'') : p \in \mathcal{P}\}$ with vector $[0, ud]$
- $\mathcal{E}_5 = \{(d', p') : p \in \mathcal{P}\}$ with vector $[0, ld]$
- $\mathcal{E}_6 = \{(s, d'), (d'', t)\}$ with vectors $[ld, ld]$ and $[ud, ud]$ resp.

All edges in $H_{(ld,ud)}$ except the edges in \mathcal{E}_2 have a cost of 0. For an edge (a, p') or (a, p'') in \mathcal{E}_2 , if the corresponding edge (a, p) in G is of rank i , then the cost of that edge in the flow network is $-n^{r-i}$, where $n = |\mathcal{A}|$. We remark that, for

an (a, p) edge of rank i , we use the same assignment of costs as mentioned in [Irving *et al.*, 2006]. It is worth noting that our reductions apply to any notion of optimality encoded as a maximum weight matching problem.

Let f be a feasible flow in $H_{(ld,ud)}$. Recall that for a post $p \in \mathcal{P}$, there are two nodes p' and p'' in the network. For the node p' , we define lower deviation of p' w.r.t. the flow f as $f(d', p')$ and upper deviation is undefined for this node. For the node p'' , we define upper deviation of p'' w.r.t. f as $f(p'', d'')$, and lower deviation is undefined for this node. For a post $p \in \mathcal{P}$, we define the total deviation of p w.r.t. f as $d_f(p) = f(d', p') + f(p'', d'')$. Similar to the definition of total deviation of a matching, we define total deviation of the flow f in $H_{(ld,ud)}$ as $d_{tot}(f) = \sum_{p \in \mathcal{P}} f(d', p') + f(p'', d'')$. By the construction of the network $H_{(ld,ud)}$, and by the definition of total deviation of a flow f in $H_{(ld,ud)}$, it is easy to verify that $d_{tot}(f) = \Delta = ld + ud$. There is a natural correspondence between the flow f in the flow network $H_{(ld,ud)}$ and matching M as follows.

$$M = \{(a, p) \mid f(a, p') = 1 \text{ or } f(a, p'') = 1, a \in \mathcal{A}, p \in \mathcal{P}\}$$

It would be ideal to have a correspondence between the total deviation of f and the total deviation of M . However, it is not true as illustrated in the simple example. Consider an input graph G that has a single edge (a, p) , and the post p has $\ell(p) = 0$ and $u(p) = 1$. Consider the flow network $H_{(1,1)}$ for $\Delta = 2$. The network $H_{(1,1)}$ admits a feasible flow f with value 2 that routes 1 unit of flow through the path $\langle s, d', p', t \rangle$ and 1 unit of flow through the path $\langle s, a, p'', d'', t \rangle$. The flow f has a total deviation $d_{tot}(f) = 2$ as mentioned earlier. The matching corresponding to f contains a single edge (a, p) and has a total deviation of 0. Thus there is a mismatch between the total deviation of the flow and the total deviation of the corresponding matching. We remark that the issue arises because both the edges (d', p') and (p'', d'') carry a non-zero flow. See full version for additional scenarios. To address these discrepancies and ensure that the total deviation of the flow corresponds to the total deviation of the matching, a feasible flow in our flow network should satisfy additional properties which are listed below as Property 1.

Property 1. Let f be a feasible flow in $H_{(ld,ud)}$. The flow f is good if for every post $p \in \mathcal{P}$ all of the following hold.

- (i) at most one of the edges (d', p') and (p'', d'') has a non-zero flow in f
- (ii) if $f(p'', d'') > 0$ then $f(p', t) = u(p)$
- (iii) if $f(d', p') > 0$ then $f(p', t) = \ell(p)$

We note that for a good flow f in $H_{(ld,ud)}$, the corresponding matching M in G satisfies $d_{tot}(f) = d_{tot}(M)$.

It is not immediate how to compute a good flow in a flow network. But given a feasible flow in $H_{(ld,ud)}$, we show how to construct a good flow in a different flow network. We need the following notations. For an edge (a, p) in G , we call the path $\langle s, a, p', t \rangle$ in the network as the *true path*, and the path $\langle s, a, p'', d'', t \rangle$ as the *extended path*. For a post p in G , we call the path $\langle s, d', p', t \rangle$ as *dummy path*. Note that corresponding to an applicant a and a post p in G , we have a true

path and an extended path using either one of this we can route a maximum of 1 unit flow in the network. Corresponding to a post p in G , we have a unique dummy path, and we can route a maximum of $\min\{ld, u(p)\}$ units of flow along this path. Let f be a feasible flow in $H_{(ld,ud)}$. We say that the status of a post $p \in \mathcal{P}$ is *good* if the corresponding pair of nodes p', p'' in the network satisfies all the conditions given in Property 1, and otherwise the status of p is *bad*.

Suppose we have a bad post p in G w.r.t. the flow f . We will reroute the flow in a certain way to turn the post p a good post without changing the status of any other posts in G . Now we describe different cases under which p can be a bad post and the rerouting required in order to turn it to a good post.

Definition 2 (correcting flow). *Let f be a feasible flow in the network $H_{(ld,ud)}$ that has a bad post p w.r.t. f and let f' be another flow in $H_{(ld,ud)}$ that differs in only the flow across some of the s - t paths corresponding to the post p . Then f' is called a correcting flow if it satisfies the following conditions.*

- (i) *There exists a network $H_{(ld',ud')}$ such that f' is feasible in the network $H_{(ld',ud')}$ where $ld' + ud' < ld + ud$.*
- (ii) *The matchings corresponding to f and f' are the same.*
- (iii) *Status of a post $q \in \mathcal{P} \setminus \{p\}$ remains same in f and f' .*

We describe the types of a correcting flow f' .

- *(ld, ud) -correcting flow.* We assume that the post p is bad due to the violation of Property 1.(i). Thus, $f(d', p') > 0$ and $f(p'', d'') > 0$. We reduce the flow along the dummy path corresponding to p by 1. Then we select an arbitrary extended path carrying unit flow corresponding to the post p , say $\langle s, a, p'', d'', t \rangle$ and reroute the flow along the path $\langle s, a, p', t \rangle$. Note that the resulting flow f' is feasible in a network $H_{(ld',ud')}$ where $ld' = ld - 1$ and $ud' = ud - 1$.
- *ud -correcting flow.* We assume that the post p is bad due to the violation of Property 1.(ii). Thus, $f(p'', d'') > 0$ and $f(p', t) < u(p)$. We select an arbitrary extended path carrying unit flow corresponding to the post p , say $\langle s, a, p'', d'', t \rangle$ and reroute the flow along the path $\langle s, a, p', t \rangle$. Note that the resulting flow f' is feasible in a network $H_{(ld',ud')}$ where $ld' = ld$ and $ud' = ud - 1$.
- *ld -correcting flow.* We assume that p is bad due to the violation of Property 1.(iii). Thus, $f(d', p') > 0$ and $f(p', t) > \ell(p)$. We reduce the flow along the dummy path corresponding to p by 1. Note that the resulting flow f' is feasible in a network $H_{(ld',ud')}$ where $ld' = ld - 1$ and $ud' = ud$.

Note that in each of the above cases, if applicant a is matched to the post p prior to the rerouting, then it continues to remain matched to p after the rerouting. Since we have not changed the flow w.r.t. any post $q \in \mathcal{P} \setminus \{p\}$, the status of the post q remains unchanged. Thus f' is a correcting flow.

Lemma 1. *If f is a feasible flow in $H_{(ld,ud)}$ that does not satisfy Property 1, then there exists a correcting flow f^* in $H_{(ld,ud)}$ that satisfies Property 1.*

Proof Sketch. For each bad post p , we identify the cause(s) of it being bad. That is, we identify which of the Property 1.(i), (ii) or (iii) being violated. Then we apply the corresponding correcting flow to change the status of post p to good. \square

To compute RMM-SIGN-MIN-TOT, we guess the value of the total deviation Δ which can be split as $ld + ud$. We construct the network $H_{(ld,ud)}$ and check if the corresponding matching M has a signature $\sigma(M) \geq_{\mathcal{R}} \rho$. We do a binary search on Δ in the range $[0, |A|]$ to compute the smallest value of Δ such that we get a matching that respects preference optimality and the corresponding flow satisfies Property 1.

3.2 Algorithm for RMM-SIGN-MIN-MAX

Given an instance of the soft quota setting, and an input signature ρ , our goal is to compute a matching M in $\mathcal{M}_{\mathcal{R}}$ that has a minimum value of max deviation. We start by guessing the max deviation value Δ and solve the rank-maximal matching in a fixed quota instance. This problem is called the RMM-LQ problem (see full version for details). The fixed quota instance has the same set of applicants and posts and applicant preferences. For each post p , we set the lower quota $q^-(p)$ as $\ell(p) - \Delta$ and upper quota $q^+(p)$ as $u(p) + \Delta$. Let M be the resulting matching. If $\sigma(M) \geq_{\mathcal{R}} \rho$, then we get a feasible matching that respects rank-maximality. We do a binary search on Δ in the range $[0, |A|]$ to find the smallest Δ such that we get a feasible matching. We remark that by applying the techniques used in [Huang *et al.*, 2016], RMM-LQ problem can be solved in $\tilde{O}(rmn)$ time.

4 Deviation First

In this section, we present the results for the deviation first, preference optimality next type of problems. In the interest of space, we defer some proofs to the full version.

4.1 Algorithm for RMM-MIN-TOT

Given an instance of the soft quota setting, our goal is to compute a matching that is rank-maximal amongst all matchings that have a minimum value of total deviation. We first discuss how to compute a minimum total deviation matching in G without applicant preferences being considered. Given an instance G of the soft quota setting, we denote the same instance viewed as an instance of the fixed quota setting by G_x . We note that w.r.t. the problem of minimizing total deviation, there is a useful connection between matchings in G and G_x . If the fixed quota setting instance G_x admits a feasible matching M , then M has a total deviation of 0 in G and hence an optimal matching in G . However, if G_x does not admit a feasible matching, any minimum total deviation matching in G has a positive value for the total deviation. Additionally, we observe that in an optimal matching M^* in G , no post incurs a violation on the upper target. To see this, if for a post p we have more than $u(p)$ many applicants matched to it in M^* , then we can remove the excess number of applicants $|M^*(p)| - u(p)$ matched to p from M^* and obtain a matching with a smaller value of total deviation. We use the above observation and the connection between G and G_x to compute a matching with minimum total deviation without applicant preferences.

Minimizing total deviation. Given the instance G , we construct an instance G' of the soft quota setting, where for every post p , $\ell'(p) = u'(p) = \ell(p)$. Consider the fixed quota setting instance G'_x and let M' be an arbitrary maximum matching in G'_x . By the maximality of M' in a fixed quota setting instance G'_x where the lower and upper quota for every post is the same, it is easy to observe that M' is a minimum total deviation matching in G' . To obtain a minimum total deviation matching in the original instance G , we use the instance G_x and augment the matching M' in G_x . We remark that augmenting a matching never decreases the number of applicants matched to any post. Thus, augmenting M' in G_x increases the size of the matching while preserving its total deviation.

For an illustration, consider the soft quota instance G given in Figure 1. We construct the soft quota instance G' , and let G'_x be the corresponding fixed quota instance. Consider the maximum matching $M' = \{(a_1, p_1), (a_3, p_3), (a_5, p_5)\}$ in G'_x . We augment the matching M' in G_x and obtain a matching $M = \{(a_1, p_1), (a_3, p_3), (a_5, p_5), (a_6, p_7), (a_7, p_6)\}$. Note that M is a maximum matching in G_x and it has a total deviation of 2 in G , which is the optimum value of total deviation.

Note that since we disregarded preferences, the above procedure does not guarantee a matching that is optimal w.r.t. the preferences. For example, the matching M computed above is not a solution of RMM-MIN-TOT problem in G since the matching M_1 (illustrated in Table 1) is more rank-maximal than M . In the rest of the section, we refine our ideas above to compute an RMM-MIN-TOT matching in G .

- We construct the instance G' from G as discussed above. We compute an RMM-MIN-TOT matching M in G' using the procedure given in step (A) below.
- We augment the matching M in G_x to obtain an RMM-MIN-TOT matching M^* in G . The procedure for computing M^* is given in step (B).

Step (A): Computing RMM-MIN-TOT in G' . Given the instance G' , we reduce the problem to computing a min-cost max-flow in a suitably designed flow network H . We show that a matching M corresponding to a min-cost max-flow in H has the following properties:

- M is a min total deviation matching in G' and also in G .
- M is a rank-maximal matching amongst all min total deviation matchings in G' . Note that M need not be a solution to RMM-MIN-TOT in G .

For every post p , we introduce $\ell(p)$ many dummy applicants. Let \mathcal{D}_p denote the set of dummy applicants associated with the post p , and let $\mathcal{D} = \bigcup_{p \in \mathcal{P}} \mathcal{D}_p$. We construct the flow network H as follows. The vertices of H are:

$$V(H) = \{s, t\} \cup (\mathcal{A} \cup \mathcal{D}) \cup \mathcal{P}.$$

The edge set $E(H)$ is the union of the following sets:

$$\begin{aligned} \mathcal{E}_{sA} &= \{(s, a) : a \in \mathcal{A}\} \\ \mathcal{E}_{sD} &= \{(s, d) : d \in \mathcal{D}_p, p \in \mathcal{P}\} \\ \mathcal{E}_{Pt} &= \{(p, t) : p \in \mathcal{P}\} \\ \mathcal{E}_{DP} &= \{(d, p) : d \in \mathcal{D}_p, p \in \mathcal{P}\} \\ \mathcal{E}_{AP} &= \{(a, p) : a \in \mathcal{A}, p \in \mathcal{P}\} \end{aligned}$$

Cost of the edges in H . For an edge (a, p) in G' having rank i , we set cost of the corresponding edge in \mathcal{E}_{AP} as $-(n^r + n^{r-i})$. For all the other edges in H , we set the cost as 0.

Capacity of the edges in H . For a (p, t) edge in \mathcal{E}_{Pt} , we set capacity as $\ell(p)$. All other edges in H have a unit capacity.

We define the deviation of a max-flow f in H .

Definition 3 (Total deviation of a max-flow). The total deviation of a max-flow f in H is defined as

$$d_{tot}(f) = \sum_{p \in \mathcal{P}} d_f(p)$$

where $d_f(p) = \sum_{d \in \mathcal{D}_p} f(d, p)$ is the incoming flow to the vertex p through the dummy applicants.

There is a natural correspondence between a flow f in H and a matching M in G' as follows

$$M = \{(a, p) \mid f(a, p) = 1, (a, p) \in \mathcal{E}_{AP}\}.$$

Cost of a max-flow in H and total deviation of a matching in G' : By the design of our network H , we ensure that the cost of a max-flow in H is directly proportional to the total deviation of the corresponding matching in G' . This is proved using Lemma 2 and Lemma 3.

Lemma 2. Let f, f' be two max-flows in H with $d_{tot}(f) < d_{tot}(f')$. Then $cost(f) < cost(f')$.

Proof Sketch. Due to the term $-n^r$ in the cost of each $(a, p) \in \mathcal{E}_{AP}$, observe that $cost(f) < cost(f')$. \square

Lemma 3. Let f, f' be two max-flows in H with $cost(f) < cost(f')$. Then $d_{tot}(f) \leq d_{tot}(f')$.

Proof of Lemma 3 follows from Lemma 2.

Lemma 4. Any min-cost max-flow in H corresponds to a min total deviation matching in G' .

Lemma 5. If k is the minimum value of the total deviation of any matching in G' , then the minimum value of the total deviation of any matching in G is k .

Corollary 1. Any min-cost max-flow in H corresponds to a min total deviation matching in G .

We now prove the preference optimality of the matching obtained by step (A).

Lemma 6. Any min-cost max-flow in H corresponds to a solution to RMM-MIN-TOT in G' .

Proof Sketch. By Corollary 1, a matching M corresponding to a min-cost max-flow f is a min total deviation matching in G' . Using the definition of rank-maximality and the correspondence between matching in G' and a max-flow in H , we show that M is a solution to RMM-MIN-TOT in G' . \square

In step (A), we have computed a matching M which is an RMM-MIN-TOT matching in G' . In step (B), we show how to compute a matching M^* which is an RMM-MIN-TOT matching in G by augmenting the matching M in G_x . We augment the matching M , such that in the j -th augmentation we get a rank-maximal matching M_j amongst all matchings in the fixed quota instance G_x with cardinality $|M| + j$.

Step (B): Computing RMM-MIN-TOT in G by augmenting M in G_x . Our algorithm is iterative and we begin with the matching $M_0 = M$. In iteration j , we augment M_{j-1} in G_x by computing an s - t path in a directed graph H_j with costs as defined below. Vertices of H_j are defined as follows.

$$V(H_j) = \{s, t\} \cup A \cup \mathcal{P}.$$

Let \mathcal{A}' denote the set of applicants that are unmatched in M_{j-1} and let \mathcal{P}' denote the set of posts with $\ell(p) \leq |M_{j-1}(p)| < u(p)$. Thus, \mathcal{P}' is the set of posts with $d_{M_{j-1}}(p) = 0$. The edge set $E(H_j)$ is the union of the following sets:

$$\begin{aligned} \mathcal{E}_{s\mathcal{A}'} &= \{(s, a) : a \in \mathcal{A}'\} \\ \mathcal{E}_{AP} &= \{(a, p) : (a, p) \in E \setminus M\} \\ \mathcal{E}_{\mathcal{P}'t} &= \{(p, t) : p \in \mathcal{P}'\} \\ \mathcal{E}_{PA} &= \{(p, a) : (a, p) \in E \cap M\} \end{aligned}$$

Cost of the edges in H_j . For an edge $(a, p) \in E$ of rank i ,

- if $(a, p) \in \mathcal{E}_{AP}$, cost of the edge is $-n^{r-i}$
- if $(p, a) \in \mathcal{E}_{PA}$, cost of the edge is n^{r-i}

All other edges in H_j have a cost of 0. Let T'_j denote a min-cost s - t path in H_j , and let $T_j = T'_j \cap E$. We augment the matching M_{j-1} using T_j , and the resulting matching is $M_j = M_{j-1} \oplus T_j$. For a set $S \subseteq E$, let $\#_i(S)$ denotes the number of rank i edges in S . We require Definition 4 to prove the correctness.

Definition 4 (Signature of a path T_j). *We define the signature of a path T_j w.r.t. the matching M_{j-1} as an r -tuple (x_1, x_2, \dots, x_r) denoted by $\sigma(T_j)$, where*

$$x_i = \#_i(T_j \cap M_{j-1}) - \#_i(T_j \setminus M_{j-1})$$

Note that $\sigma(T_j)$ can contain negative values. It is useful to note the connection between the signature of a path $\sigma(T_j)$ and the cost of the path $\text{cost}(T_j)$.

$$\text{cost}(T_j) = \sum_{i=1}^r x_i n^{r-i}.$$

We observe the following:

Observation 1. $\sum_{i=1}^r x_i = -1$.

Observation 1 is trivial because T_j is an augmenting path. Let $\sigma(M_{j-1}) = (y_1, y_2, \dots, y_r)$. The value $x_i > 0$ in $\sigma(T_j)$ indicates that T_j has more number of matched rank i edges than the unmatched rank i edges, and the value x_i cannot be greater than the number of matched rank i edges y_i in M_{j-1} . This is precisely stated in Observation 2.

Observation 2. For any $i \in [r]$, if $x_i > 0$ then $x_i \leq y_i$.

Since $M_j = M_{j-1} \oplus T_j$, it is easy to observe that $\sigma(M_j) = \sigma(M_{j-1}) - \sigma(T_j)$.

Observation 3. Every component in $\sigma(M_j)$ is non-negative.

Observation 3 follows from Observation 2. We state a series of claims to prove the correctness of step (B). We ensure that our augmentations do not change the value of total deviation of the resulting matching, and this is stated in Claim 1.

Claim 1. If $d_{\text{tot}}(M) = k$, then $d_{\text{tot}}(M_j) = k$.

Next, we show that as the cost of path decreases, signature of the path w.r.t. rank-maximality also decreases.

Claim 2. Let T'_j and Q'_j be two s - t paths in H_j such that $\text{cost}(T'_j) < \text{cost}(Q'_j)$. Then $\sigma(T_j) <_{\mathcal{R}} \sigma(Q_j)$.

Our iterative procedure halts in iteration j , if the graph H_j does not admit any s - t path. Let M^* be the matching obtained at the end of step (B).

Lemma 7. M^* is a solution for RMM-MIN-TOT in G

Proof. From Claim 1 and Lemma 5, M^* is a min total deviation matching in G . In each iteration, the algorithm chooses an s - t path T'_j in H_j of minimum cost, by Claim 2 the path T_j has the smallest signature possible amongst all s - t paths in H_j . Since we are subtracting the smallest signature, at the end of step (B), the matching M^* has largest signature w.r.t. rank-maximality amongst all matchings with minimum total deviation in G . \square

We note that our iterative algorithm in step (B) can be converted to a min-cost max-flow problem in a suitably designed network such that a single min-cost max-flow computation captures the repeated augmentation of the matching M in G_x .

4.2 Algorithm for RMM-MIN-MAX

Given an instance of the soft quota setting, our goal is to compute a rank-maximal matching amongst all matchings that have a minimum value of max deviation. To compute the minimum value of max deviation, we start by guessing the max deviation value Δ and solve a maximum matching in the fixed quota setting instance defined as follows. The fixed quota setting has the same set of applicants and posts and applicant preferences. For each post p , we set the lower quota $q^-(p)$ as $\ell(p) - \Delta$ and upper quota $q^+(p)$ as $u(p) + \Delta$. We do a binary search on Δ in the range $[0, |A|]$ to find the smallest Δ such that we get a feasible solution. For checking feasibility we can ignore the preferences and use the standard flow network for the bipartite matching with demands on the edges. Let Δ^* be the minimum value of Δ obtained. Then corresponding to Δ^* , we solve one instance of RMM-LQ where for every post p , it has a lower quota of $\ell(p) - \Delta^*$ and an upper quota of $u(p) + \Delta^*$. Since we solve the RMM-LQ instance for this, we get the best possible signature with the minimum max-deviation.

5 Discussion

In this work we study the problem of computing optimal matchings with one-sided preferences in the soft quota setting. We consider two orthogonal optimization parameters namely the deviation of the matching and the preference optimality. For the preference optimality notions of rank-maximality and fairness, we present efficient algorithms. Our algorithms need to use exponential weights on the edges of the flow network. It would be interesting to design algorithms that do not need large weights.

Acknowledgements

We thank Govind S Sankar for his valuable contributions and insightful discussions that greatly assisted us in the initial phase of this work. We extend our gratitude to both Microsoft Research India and the IJCAI-AIJ 2023 travel grant committee for their generous contribution towards the partial funding of the travel grant.

References

- [Abe *et al.*, 2022] Kenshi Abe, Junpei Komiyama, and Atsushi Iwasaki. Anytime capacity expansion in medical residency match by monte carlo tree search. In Lud De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 3–9. International Joint Conferences on Artificial Intelligence Organization, 7 2022. Main Track.
- [Abraham *et al.*, 2004] David J Abraham, Katarína Cechlárová, David F Manlove, and Kurt Mehlhorn. Pareto optimality in house allocation problems. In *International Symposium on Algorithms and Computation*, pages 3–15, Berlin, Heidelberg, 2004. Springer.
- [Abraham *et al.*, 2006] David Abraham, Ning Chen, Vijay Kumar, and Vahab S. Mirrokni. Assignment problems in rental markets. In Paul Spirakis, Marios Mavronicolas, and Spyros Kontogiannis, editors, *Internet and Network Economics*, pages 198–213, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
- [Abraham, 2009] David John Abraham. *Matching markets: Design and analysis*. PhD thesis, Carnegie Mellon University, 2009.
- [Aggarwal *et al.*, 2014] Gagan Aggarwal, Yang Cai, Aranyak Mehta, and George Pierrakos. Biobjective online bipartite matching. In *International Conference on Web and Internet Economics*, pages 218–231. Springer, 2014.
- [Arulselvan *et al.*, 2018] Ashwin Arulselvan, Ágnes Cseh, Martin Groß, David F. Manlove, and Jannik Matuschke. Matchings with lower quotas: Algorithms and complexity. *Algorithmica*, 80(1):185–208, 2018.
- [Aziz *et al.*, 2020] Haris Aziz, Serge Gaspers, and Zhaohong Sun. Mechanism design for school choice with soft diversity constraints. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 153–159. International Joint Conferences on Artificial Intelligence Organization, 7 2020. Main track.
- [Aziz *et al.*, 2022] Haris Aziz, Péter Biró, and Makoto Yokoo. Matching market design with constraints. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(11):12308–12316, Jun. 2022.
- [Biró *et al.*, 2010] Péter Biró, Tamás Fleiner, Robert W Irving, and David F Manlove. The college admissions problem with lower and common quotas. *Theoretical Computer Science*, 411(34-36):3136–3153, 2010.
- [Biró, 2017] Péter Biró. Applications of matching models under preferences. In Ulle Endriss, editor, *Trends in Computational Social Choice*, chapter 18, pages 345–373. AI Access, 2017.
- [Bobbio *et al.*, 2021] Federico Bobbio, Margarida Carvalho, Andrea Lodi, and Alfredo Torrico. Capacity expansion in the college admission problem. *arXiv preprint arXiv:2110.00734*, 2021.
- [Bobbio *et al.*, 2022] Federico Bobbio, Margarida Carvalho, Andrea Lodi, and Alfredo Torrico. Capacity variation in the many-to-one stable matching. *arXiv preprint arXiv:2205.01302*, 2022.
- [Cechlárová and Fleiner, 2017] Katarína Cechlárová and Tamás Fleiner. Pareto optimal matchings with lower quotas. *Mathematical Social Sciences*, 88:3–10, 2017.
- [Cechlárová *et al.*, 2016] Katarína Cechlárová, Pavlos Eirinakis, Tamás Fleiner, Dimitrios Magos, David Manlove, Ioannis Mourtos, Eva Oceláková, and Baharak Rastegari. Pareto optimal matchings in many-to-many markets with ties. *Theory of Computing Systems*, 59(4):700–721, 2016.
- [Chen and Csáji, 2023] Jiehua Chen and Gergely Csáji. Optimal capacity modification for many-to-one matching problems. In *Proceedings of the 22nd International Conference on Autonomous Agents and Multiagent Systems, AAMAS '23*, Richland, SC, 2023. International Foundation for Autonomous Agents and Multiagent Systems.
- [Cooper, 2020] Frances Cooper. *Fair and large stable matchings in the stable marriage and student-project allocation problems*. PhD thesis, University of Glasgow, 2020.
- [Cseh *et al.*, 2022] Ágnes Cseh, Tobias Friedrich, and Jannik Peters. Pareto optimal and popular house allocation with lower and upper quotas. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems, AAMAS '22*, page 300–308, Richland, SC, 2022. International Foundation for Autonomous Agents and Multiagent Systems.
- [Dhamala and Thapa, 2009] Tanka Nath Dhamala and Gyan Bahadur Thapa. Apportionment approach for just-in-time sequencing problem. *Journal of the Institute of Engineering*, 7(1):76–89, 2009.
- [Echenique and Yenmez, 2015] Federico Echenique and M Bumin Yenmez. How to control controlled school choice. *American Economic Review*, 105(8):2679–94, 2015.
- [Ehlers *et al.*, 2014] Lars Ehlers, Isa E Hafalir, M Bumin Yenmez, and Muhammed A Yildirim. School choice with controlled choice constraints: Hard bounds versus soft bounds. *Journal of Economic theory*, 153:648–683, 2014.
- [Esfandiari *et al.*, 2016] Hossein Esfandiari, Nitish Korula, and Vahab Mirrokni. Bi-objective online matching and submodular allocations. *Advances in Neural Information Processing Systems*, 29, 2016.
- [Fleiner and Kamiyama, 2016] Tamás Fleiner and Naoyuki Kamiyama. A matroid approach to stable matchings

- with lower quotas. *Mathematics of Operations Research*, 41(2):734–744, 2016.
- [Fragiadakis *et al.*, 2016] Daniel Fragiadakis, Atsushi Iwasaki, Peter Troyan, Suguru Ueda, and Makoto Yokoo. Strategyproof matching with minimum quotas. *ACM Transactions on Economics and Computation (TEAC)*, 4(1):1–40, 2016.
- [Gajulapalli *et al.*, 2020] Karthik Gajulapalli, James A Liu, Tung Mai, and Vijay V Vazirani. Stability-preserving, time-efficient mechanisms for school choice in two rounds. In *40th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, 2020.
- [Garg *et al.*, 2010] Naveen Garg, Telikepalli Kavitha, Amit Kumar, Kurt Mehlhorn, and Julián Mestre. Assigning papers to referees. *Algorithmica*, 58(1):119–136, 2010.
- [Goko *et al.*, 2022] Hiromichi Goko, Kazuhisa Makino, Shuichi Miyazaki, and Yu Yokoi. Maximally satisfying lower quotas in the hospitals/residents problem with ties. In *39th International Symposium on Theoretical Aspects of Computer Science*, 2022.
- [Gonczarowski *et al.*, 2019] Yannai A Gonczarowski, Noam Nisan, Lior Kovalio, and Assaf Romm. Matching for the israeli” mechinot” gap-year programs: Handling rich diversity requirements. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 321–321, 2019.
- [Goto *et al.*, 2014] Masahiro Goto, Naoyuki Hashimoto, Atsushi Iwasaki, Yujiro Kawasaki, Suguru Ueda, Yosuke Yasuda, and Makoto Yokoo. Strategy-proof matching with regional minimum quotas. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 1225–1232, 2014.
- [Hamada *et al.*, 2016] Koki Hamada, Kazuo Iwama, and Shuichi Miyazaki. The hospitals/residents problem with lower quotas. *Algorithmica*, 74(1):440–465, 2016.
- [Hosseini *et al.*, 2021] Hadi Hosseini, Vijay Menon, Nisarg Shah, and Sujoy Sikdar. Necessarily optimal one-sided matchings. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(6):5481–5488, May 2021.
- [Huang *et al.*, 2016] Chien-Chung Huang, Telikepalli Kavitha, Kurt Mehlhorn, and Dimitrios Michail. Fair matchings and related problems. *Algorithmica*, 74(3):1184–1203, 2016.
- [Irving *et al.*, 2006] Robert W Irving, Telikepalli Kavitha, Kurt Mehlhorn, Dimitrios Michail, and Katarzyna E Paluch. Rank-maximal matchings. *ACM Transactions on Algorithms (TALG)*, 2(4):602–610, 2006.
- [K. A. *et al.*, 2022] Santhini K. A., Govind S. Sankar, and Meghana Nasre. Optimal matchings with one-sided preferences: Fixed and cost-based quotas. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, AAMAS ’22, page 696–704, Richland, SC, 2022. International Foundation for Autonomous Agents and Multiagent Systems.
- [Korula *et al.*, 2013] Nitish Korula, Vahab S Mirrokni, and Morteza Zadimoghaddam. Bicriteria online matching: Maximizing weight and cardinality. In *International conference on web and internet economics*, pages 305–318. Springer, 2013.
- [Kurata *et al.*, 2017] Ryoji Kurata, Naoto Hamada, Atsushi Iwasaki, and Makoto Yokoo. Controlled school choice with soft bounds and overlapping types. *Journal of Artificial Intelligence Research*, 58:153–184, 2017.
- [Limaye and Nasre, 2023] Girija Limaye and Meghana Nasre. Optimal cost-based allocations under two-sided preferences. In *34th International Workshop on Combinatorial Algorithms (IWOCA) (accepted)*, 2023.
- [Monte and Tumennasan, 2013] Daniel Monte and Norovsambuu Tumennasan. Matching with quorums. *Economics Letters*, 120(1):14–17, 2013.
- [Nasre and Nimbhorkar, 2018] Meghana Nasre and Prajakta Nimbhorkar. Popular matchings with lower quotas. In *37th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2017)*, 2018.
- [Nasre *et al.*, 2019] Meghana Nasre, Prajakta Nimbhorkar, and Nada Pulath. Classified rank-maximal matchings and popular matchings – algorithms and hardness. In Ignasi Sau and Dimitrios M. Thilikos, editors, *Graph-Theoretic Concepts in Computer Science*, pages 244–257, Cham, 2019. Springer International Publishing.
- [Nasre *et al.*, 2021] Meghana Nasre, Prajakta Nimbhorkar, Keshav Ranjan, and Ankita Sarkar. Popular matchings in the hospital-residents problem with two-sided lower quotas. In *41st IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2021)*, 2021.
- [Paluch, 2013] Katarzyna Paluch. Capacitated rank-maximal matchings. In *International Conference on Algorithms and Complexity*, pages 324–335, Berlin, Heidelberg, 2013. Springer.
- [Peters, 2022] Jannik Peters. Online elicitation of necessarily optimal matchings. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(5):5164–5172, Jun. 2022.
- [Pukelsheim *et al.*, 2012] Friedrich Pukelsheim, Federica Ricca, Bruno Simeone, Andrea Scozzari, and Paolo Serafini. Network flow methods for electoral systems. *Networks*, 59(1):73–88, 2012.
- [Rios *et al.*, 2021] Ignacio Rios, Tomas Larroucau, Giorgiogiulio Parra, and Roberto Cominetti. Improving the chilean college admissions system. *Operations Research*, 69(4):1186–1205, 2021.
- [Serafini and Simeone, 2012] Paolo Serafini and Bruno Simeone. Parametric maximum flow methods for min-max approximation of target quotas in biproportional apportionment. *Networks*, 59(2):191–208, 2012.