Algorithms for Stable Matching Problems toward Real-World Applications

A Dissertation
Presented to the Graduate School of Informatics
Kyoto University
in Candidacy for the Degree of
Doctor of Philosophy

by
Koki Hamada
Kyoto University, Japan

Supervisor: Associate Prof. Shuichi Miyazaki

Abstract

In the *stable marriage problem* (*SM* for short), we are given sets of men and women, and each person's preference list that strictly orders the members of the other gender according to his/her preference. The question is to find a *stable matching*, that is, a matching containing no pair of man and woman who prefer each other to their partners. Such a pair is called a *blocking pair*. It is known that any instance admits at least one stable matching, and there is a polynomial time algorithm to find one. Many extensions are studied and they are collectively called *stable matching problems*.

Stable matching problems have already been applied in the real world, such as assigning residents to hospitals or assigning students to schools; however, some applications have requirements in addition to stability, for example, assigning as many residents as possible. In this thesis, we study computational tractability of various extensions of stable matching problems in order to fulfill such requirements and make them more widely applicable.

In Chapter 3, we give a hardness result for a problem of finding a maximum cardinality matching that is as stable as possible. In the stable marriage problem that allows incomplete preference lists, all stable matchings for a given instance have the same size. However, if we ignore the stability, there can be larger matchings. For a problem of finding a maximum cardinality matching that contains minimum number of blocking pairs, it was proved that this problem is not approximable within some constant unless P=NP. We substantially improve this lower bound.

In Chapter 4, we define a many-to-one extension of SM called *hospitals/residents problems with lower quotas* (*HRLQ* for short). In HRLQ, the two sets are *residents* and *hospitals*, and each hospital has lower and upper quotas on the number of residents to be assigned. Only matchings that satisfy both upper and lower quotas for all hospitals are feasible. In this setting, there can be instances that admit no stable matching, but the problem of asking if there is a stable matching is solvable in polynomial time. In case there is no stable matching, we consider the problem of finding a matching with minimum number of blocking pairs. We show that this problem is hard to approximate. We then consider another measure for optimization criteria, i.e., the number of residents who are involved in blocking pairs. We show that this problem is still NP-hard but has a polynomial-time algorithm with non-trivial approximation ratio.

In Chapter 5, we give algorithms and an NP-completeness proof for the problems of finding stable matching without edge crossings. As an extension of SM that can represent some of physical constraints, problems of finding a stable matching without edge crossings has been considered. There are two stability notions, *strongly stable noncrossing matching* (*SSNM*) and *weakly stable noncrossing matching* (*WSNM*), depending on the strength of blocking pairs. It was proved that a WSNM always exists and a polynomial-time algorithm to find one is known; however, the complexities of determining existence of an SSNM and finding a largest WSNM remained open. We show that both problems are solvable in polynomial time. We also show that our algorithms are applicable to extensions where preference lists may include ties, except for one case which we show to be NP-complete.

In Chapter 6, we consider strategy-proofness in an extension of SM. SM can be seen as a game among participants, who have true preferences in mind, but may submit a falsified preference list hoping to obtain a better partner than the one assigned when true preference lists are used. We say that an algorithm is *strategy-proof* if, when it is used, no person can obtain a better partner by submitting a falsified preference list in any instance. There are some positive and negative results on strategy-proofness for SM. The *stable marriage problem with ties and incomplete lists* (*SMTI* for short) is an extension of SM in which preference lists may contains ties and may include only a subset of the member of the opposite gender. By contrast to SM, there is an SMTI-instance that admits stable matchings of different sizes, and the problem of finding a stable matching of the maximum size, called *MAX SMTI*, is NP-hard. There are a plenty of approximability and inapproximability results for MAX SMTI, but there is no result on strategy-proofness. We introduce it to MAX SMTI, and investigate the trade-off between strategy-proofness and approximability.

These results contribute to understanding computational tractability of complex stable matching problems for real-world applications. Our results have also made several contributions to the overall study of stable matching problems. One is strengthening the common understanding that minimizing the number of blocking pairs is

difficult. The hardness results shown in Chapters 3 and 4 add evidences to it. Another contribution is obtaining tight results for the stable matching problems. We give tight upper and lower bounds on approximation ratios for several variants and a tight condition on the existence of polynomial-time algorithm for a decision problem. Our results also provide an avenue for subsequent studies. There are subsequent studies that circumvent our hardness results by considering alternative solution concepts. In addition, our proof technique for showing strategy-proofness given in Chapter 6 is generic, and was used in subsequent work.

Contents

List of Figures

Acknowledgments

First, I would like to express my sincere gratitude to my supervisor, Associate Professor Shuichi Miyazaki, for his guidance over the years. He gave me a lot of insightful advice and always warmly helped me when I faced difficulties. I owe my deep gratitude to Professor Yasuo Okabe. He kindly accepted me in his laboratory and gave me many advices in the laboratory seminars and in the review of this thesis. I would also like to appreciate Professor Tatsuya Akutsu and Professor Shin-ichi Minato for spending their time as members of my dissertation committee and for their valuable comments and suggestions.

I would like to express my thanks to my supervisor in the master's course, Professor Kazuo Iwama, for giving me the opportunity to start research on the stable matching problem and for his guidance even after graduation. I would also like to thank my collaborators, Dr. Hiroki Yanagisawa and Dr. Kazuya Okamoto, for giving me many helpful discussions on our research.

I want to thank all the past and current members in Okabe Laboratory. Especially, Assistant Professor Daisuke Kotani gave me many comments in the laboratory seminars and helped me in creating my research environment. Ms. Tamayo Hayashi helped me with various procedures and tasks in my doctoral student life.

I also express my thanks to members in NTT Social Informatics Laboratories for understanding and encouraging me to start and continue my doctoral program. I would especially like to thank Dr. Yo Kanemoto for sharing his experience as a senior doctoral student with me.

Lastly, I would like to thank my family for their constant support and encouragement. Without them, it would be impossible for me to complete my study.

Chapter 1

Introduction

1.1 Background

The *stable marriage problem* (*SM* for short) is a widely known problem first studied by Gale and Shapley [GS62]. We are given sets of men and women, and each person's preference list that strictly orders the members of the other gender according to his/her preference. The question is to find a *stable matching*, that is, a matching containing no pair of man and woman who prefer each other to their partners. Such a pair is called a *blocking pair*. Gale and Shapley proved that any instance admits at least one stable matching, and gave an algorithm to find one, known as the *Gale-Shapley algorithm*. SM and its extensions are collectively referred to as *stable matching problems*.

Some of stable matching problems have already been applied in the real world. In the United States, the National Intern Matching Program (currently the National Resident Matching Program, NRMP) has been matching residents with hospitals since 1952. The algorithm used there was essentially the same as the Gale-Shapley algorithm. Since then, stable matching problems have been applied to assigning residents to hospitals [GI89, CaR, IMS00], assigning students to schools [TST01, APR05, APRS05], and finding donors for kidney exchange [RSÜ04]. One of the reasons for this widespread application of stable matching problems is that stability is important in real world applications. Roth has shown empirical evidence that algorithms which always output stable matchings last longer than those which may output unstable matchings [Rot02].

When applying stable matching problems, various conditions are required in addition to stability. For example, in the 1970s, there was a decline in the number of participants in the NRMP because the algorithm used at the time could not satisfy the desire of resident couples to be assigned to hospitals near each other [Rot84]. To solve this problem, the algorithm was redesigned to take the couple's preferences into account [RP99]. Another example is the school choice in New York [APR05]. When revising the school choice system in New York, in addition to stability, strategy-proofness was considered important. *Strategy-proofness* for an algorithm ensures that no participant can falsify his or her input to obtain a better outcome. Without this property, participants will have an incentive to submit a fake preference list, and the correct output will not be obtained. As such, it is desirable to be able to satisfy a variety of requirements in order for stable matching problems to continue to be applied.

Various stable matching problems have been considered for real-world applications. One of the main classes of stable matching problems is a set of variants with generalized inputs or outputs. As for output generalization, the problem of finding a many-to-one matching, called the *hospitals/residents problem* (*HR* for short) [GS62], and the one for many-to-many [Sot99] are studied. As for input generalization, variants with different number of parties are considered. The *stable roommates problem* (*SR* for short) [GS62] is a problem of finding a stable matching within a single party rather than two parties. There are also problems with three parties [Knu76] and more parties [Lic15].

Another class of variants includes preference list generalization. A typical example of this class is the problem that allows an *incomplete list* where the preference list comprises only a subset of the opposite set [GS62]. This is called *SM with incomplete lists* (*SMI* for short). *SM with ties* (*SMT* for short) [GI89] is another typical variant, which allows *ties* in preference lists.

There is also a set of variants with constraints to exclude undesirable matchings. In applications, there are often requirements for matching other than stability, as in the case with couples [RP99] we saw above. The constraints of this class of problems are considered in connection with such requirements. For example, an extension of HR called the *student-project allocation problem* [AIM07] models the assignment of students to projects in a university. In this problem, in addition to a capacity of each project, there is a capacity for a lecturer that bounds the number of students assigned to the projects offered by her. In addition, the *lower quota* constraint, which gives a lower bound on the number of residents assigned to each hospital [BFIM10], has been studied to prevent imbalance in the number of residents assigned to hospitals in HR.

For the same purpose, the *common quota* constraint, which gives an upper bound on the sum of the number of residents assigned to each hospital for a subset of hospitals [BFIM10, KK10], and the *classified* constraint, which sets upper and lower bounds on the number of residents assigned to some subset of residents [Hua10], have also been studied.

1.2 Purpose of this Thesis

In real-world applications of stable matching problems, conditions such as above generalizations and constraints, and even combinations of them, are often required. However, since such a complex problem is rarely computationally tractable, it is desirable to be able to balance the conditions by weakening each condition to make it computationally tractable. In this thesis, as steps toward making this possible, we aim to investigate the computational tractability of stable matching problems under various conditions. Specifically, we study algorithms for problems that prioritize constraints other than stability, algorithms for stable matching problems with physical constraints, and strategy-proof algorithms for stable matching problems.

We first study algorithms for stable matching problems in which stability is not a constraint but an objective function for optimization. In the real world, stability is not necessarily the top priority. For example, when assigning residents, it may be desirable to increase the number of assigned residents or reduce the imbalance in the number of assigned residents, even at the cost of stability. We therefore study, in Chapters 3 and 4, the problem of finding a matching that is "as stable as possible" while satisfying some conditions.

In Chapter 3, we study the computational tractability of the problem of finding maximum matching that is "as stable as possible" in SMI. Possible measures of instability are the number of blocking pairs [EH08, NR04, KMV94] and the number of blocking residents that are included in a blocking pair [EH08, RX97]. We consider minimizing the number of blocking pairs, which is the more natural of the two. The problem is defined by Biró et al. [BMM10].

In Chapter 4, we define an extension of HR, which we call *HR with lower quotas* (*HRLQ* for short), with a lower bound on the number of residents assigned to each hospital. We show some hardness and approximability results for the problem of finding a matching that is "as stable as possible" among the ones satisfying the lower quotas. The lower quota is a constraint that naturally responds to the request of real-world applications to reduce the imbalance in the number of residents assigned to each hospital. For HR with lower quota constraint, three models—i.e., Biró et al.'s [BFIM10], Huang's [Hua10], and ours—were proposed at around the same time. These models have led to many subsequent studies.

Next, we consider physical constraints. When we try to apply matching algorithms in the real world, we are often faced with physical constraints. For example, when we wire a circuit or build a bridge over a river, we need to find a matching such that no wires or bridges cross each other. Ruangwises and Itoh [RI19] incorporated the notion of noncrossing matchings [Ata85, CLW15, KT86, MOP93, WW85] to stable matching problems. In Chapter 5, we positively solve the two open problems proposed in [RI19] and extend it to an extension of SM called *SMTI*.

Finally, we study strategy-proof algorithms for stable matching problems. Optimization algorithms find a solution based on the assumption that the input is correct; however, this is not always the case in the real world. A participant who submits an input may try to get a better output by submitting false information. To discourage such attempts, strategy-proofness has been studied in an area of economics. We say that an algorithm is strategy-proof if no participant can obtain a better output by falsifying his/her input. It is known that there is no strategy-proof algorithm that finds a stable matching in SM [Rot82]. By contrast, one form of the Gale-Shapley algorithm is *man-strategy-proof* [DF81, Rot82], which means that no man can obtain a better output by falsifying his input. In Chapter 6, we give strategy-proof algorithms for SMTI, for which no strategy-proof algorithm was previously known.

1.3 Results of this Thesis

In Chapter 3, we give a hardness result for a problem of finding a maximum cardinality matching that is as stable as possible. In SMI, all stable matchings for a given instance have the same size. However, if we ignore the stability, there can be larger matchings. Biró et al. [BMM10] defined the problem of finding a maximum cardinality matching that contains minimum number of blocking pairs. A restriction of the problem is called MAX SIZE MIN BP (p, q)-SMI, where p (q) is an upper bound on the length of each man's (woman's, respectively) preference list. They showed the following results; (1) MAX SIZE MIN BP (∞, ∞)-SMI is NP-hard and cannot be approximated within the ratio of $n^{1-\varepsilon}$ for any constant $\varepsilon > 0$, unless P=NP; (2) MAX SIZE MIN BP (3, 3)-SMI is APX-hard and cannot be approximated within the ratio of $\frac{3557}{3556} \simeq 1.00028$

unless P=NP; (3) MAX SIZE MIN BP (2, ∞)-SMI is solvable in $O(n^3)$ time, where n is the number of men in an input. We improved the lower bound of (2), namely, we improved the constant $\frac{3557}{3556}$ to $n^{1-\varepsilon}$ for any $\varepsilon > 0$.

In Chapter 4, we define HRLQ. In HRLQ, the two sets are *residents* and *hospitals*, and each hospital has lower and upper quotas on the number of residents to be assigned. Only matchings that satisfy both upper and lower constraints for all hospitals are feasible. In this setting, there can be instances that admit no stable matching, but the problem of asking if there is a stable matching is solvable in polynomial time. In case there is no stable matching, we consider the problem of finding a matching that is "as stable as possible", namely, a matching with a minimum number of blocking pairs. We show that this problem is hard to approximate within the ratio of $(|H| + |R|)^{1-\epsilon}$ for any positive constant ϵ where H and R are the sets of hospitals and residents, respectively. We then tackle this hardness from two different angles. First, we give an exponential-time exact algorithm whose running time is $O((|H||R|)^{t+1})$, where t is the number of blocking pairs in an optimal solution. Second, we consider another measure for optimization criteria, i.e., the number of residents who are involved in blocking pairs. We show that this problem is still NP-hard but has a polynomial-time $\sqrt{|R|}$-approximation algorithm.

In Chapter 5, we give algorithms and an NP-completeness proof for the problems of finding stable matching without edge crossings. Ruangwises and Itoh [RI19] introduced *stable noncrossing matchings*, where participants of each side are aligned on each of two parallel lines, and no two matching edges are allowed to cross each other. They defined two stability notions, *strongly stable noncrossing matching* (*SSNM*) and *weakly stable noncrossing matching* (*WSNM*), depending on the strength of blocking pairs. They proved that a WSNM always exists and presented an $O(n^2)$-time algorithm to find one for an instance with n men and n women. They also posed open questions of the complexities of determining existence of an SSNM and finding a largest WSNM. We show that both problems are solvable in polynomial time. Our algorithms are applicable to extensions where preference lists may include ties, except for one case which we show to be NP-complete. This NP-completeness holds even if each person's preference list is of length at most two and ties appear in only men's preference lists. To complement this intractability, we show that the problem is solvable in polynomial time if the length of preference lists of one side is bounded by one (but that of the other side is unbounded).

In Chapter 6, we give strategy-proof algorithms for finding largest cardinality matchings in *SM with ties and incomplete lists* (SMTI for short). SMTI is an extension of SM in which preference lists may contains ties and may include only a subset of the members of the opposite gender. In SM, a mechanism that always outputs a stable matching is called a *stable mechanism*. One of the well-known stable mechanisms is the *man-oriented Gale-Shapley algorithm* (*MGS* for short), which is a form of Gale-Shapley algorithm. MGS is strategy-proof to the men's side, i.e., no man can obtain a better outcome by falsifying a preference list [DF81, Rot82]. We call such a mechanism a *man-strategy-proof mechanism*. Unfortunately, MGS is not a *woman-strategy-proof mechanism*.[*1] Roth has shown that there is no stable mechanism that is simultaneously man-strategy-proof and woman-strategy-proof, which is known as Roth's impossibility theorem [Rot82]. We extend these results to SMTI. Since it is an extension of SM, Roth's impossibility theorem takes over to it. Therefore, we focus on the one-sided-strategy-proofness. In SMTI, one instance can have stable matchings of different sizes, and it is natural to consider the problem of finding a largest stable matching, known as *MAX SMTI*. Thus we incorporate the notion of approximation ratios used in the theory of approximation algorithms. We say that a stable-mechanism is a *c-approximate-stable mechanism* if it always returns a stable matching of size at least $1/c$ of a largest one. We also consider a restricted variant of MAX SMTI, which we call *MAX SMTI-1TM*, where only men's lists can contain ties (and women's lists must be strictly ordered). Since MAX SMTI-1TM is NP-hard [MII$^+$02] and current best upper bounds for the approximation ratios of MAX-SMTI and MAX SMTI-1TM are 1.5 [McD09, Pal14, Kir13] and $1 + 1/e \simeq 1.368$ [LP19], respectively, we work on designing strategy-proof approximation algorithms. Our results are summarized as follows: (i) MAX SMTI admits both a man-strategy-proof 2-approximate-stable mechanism and a woman-strategy-proof 2-approximate-stable mechanism. (ii) MAX SMTI-1TM admits a woman-strategy-proof 2-approximate-stable mechanism. (iii) MAX SMTI-1TM admits a man-strategy-proof 1.5-approximate-stable mechanism. All these results are tight in terms of approximation ratios. Also, all these results apply for strategy-proofness against coalitions. The current best polynomial-time approximation algorithms for MAX SMTI and MAX SMTI-1TM have the approximation ratios better than those in our negative results. Hence our results provide

[*1] Of course, if we flip the roles of men and women, we can see that the *woman-oriented Gale-Shapley algorithm* (*WGS*) is a woman-strategy-proof but not a man-strategy-proof mechanism.

gaps between polynomial-time computation and strategy-proof computation.

These results contribute to understanding computational tractability of complex stable matching problems for real-world applications. Our results have also made several contributions to the overall study of stable matching problems. One is strengthening the common understanding that minimizing the number of blocking pairs is difficult. The hardness results shown in Chapters 3 and 4 add evidences to it. Another is obtaining tight results, in terms of upper and lower bounds on the approximation ratio and condition for the existence of a polynomial-time algorithm. Our results also provide an avenue for subsequent studies. There are subsequent studies that circumvent our hardness results by considering alternative solution concepts. In addition, our proof technique for showing strategy-proofness given in Chapter 6 is generic, and was used in subsequent work. More detailed discussions will be given in Chapter 7.

Chapter 2

Preliminaries

2.1 Stable Matching Problems

2.1.1 Stable Marriage Problem

The *stable marriage problem* (*SM* for short), introduced by Gale and Shapley [GS62] (see also [GI89]), is defined as follows: An instance consists of n men $m_1, m_2, \ldots, m_n$, n women $w_1, w_2, \ldots, w_n$, and each person's *preference list*, which is a total order of all the members of the opposite gender. If a person q_i precedes a person q_j in a person p's preference list, then we write $q_i \succ_p q_j$ and interpret it as "p prefers q_i to q_j". In this thesis, we denote a preference list in the following form:

$$m_2 : w_3 \; w_1 \; w_4 \; w_2,$$

which means that m_2 prefers w_3 best, w_1 second, w_4 third, and w_2 last (this example is for $n = 4$). A *matching* is a set of n (man, woman)-pairs in which no person appears more than once. For a matching M, $M(p)$ denotes the partner of a person p in M. If, for a man m and a woman w, both $w \succ_m M(m)$ and $m \succ_w M(w)$ hold, then we say that (m, w) is a *blocking pair* (*BP* for short) for M or (m, w) *blocks* M. Note that both m and w have incentive to be matched with each other ignoring the given partner, so it can be thought of as a threat for the current matching M. A matching with no blocking pair is a *stable matching.* The problem requires to find a stable matching.

It is known that any instance admits at least one stable matching, and one can be found by the *Gale-Shapley algorithm* (or *GS algorithm* for short) in $O(n^2)$ time [GS62].

2.1.2 Incomplete Lists

One possible extension of SM is to allow incomplete preference lists, which we call *SM with incomplete lists* (*SMI* for short); namely, each person includes a *subset* of the members of the opposite gender in the preference list. We call such an instance an *SMI-instance.* If a person q appears in a person p's preference list, we say that q is *acceptable* to p. If p and q are acceptable to each other, we say that (p, q) is an *acceptable pair.* We assume without loss of generality that acceptability is mutual, i.e., p is acceptable to q if and only if q is acceptable to p. Now a matching is defined as a set of disjoint pairs of mutually acceptable man and woman, and hence is not necessarily perfect. If a person p is not included in a matching M, we say that p is *single* in M and write $M(p) = \emptyset$. Every person prefers to be matched with an acceptable person rather than to be single, i.e., $q \succ_p \emptyset$ holds for any p and any q acceptable to p. Accordingly, the definition of a blocking pair is extended as follows: A mutually acceptable pair of man m and woman w is a blocking pair for a matching M if (i) m and w are not matched together in M, (ii) either m is single or prefers w to his partner in M, and (iii) either w is single or prefers m to her partner in M. The *size* of a matching M, denoted $|M|$, is the number of pairs in M. There can be many stable matchings for one instance, but all stable matchings are of the same size [GS85].

2.1.3 Ties

We then extend the above definitions to the case where preference lists may contain ties. Such an extension is called *SM with ties and incomplete lists* denoted *SMTI.* A *tie* of a person p's preference list is a set of one or more persons who are equally preferred by p, and p's preference list is a strict order of ties. We call such an instance an *SMTI-instance.* In a person p's preference list, suppose that a person q_1 is in tie T_1, q_2 is in tie T_2, and p prefers T_1 to T_2. Then we say that p *strictly prefers* q_1 to q_2 and write $q_1 \succ_p q_2$. If q_1 and q_2 are in the same tie (including the case that q_1 and q_2 are the same person), we write $q_1 =_p q_2$. If $q_1 \succ_p q_2$ or $q_1 =_p q_2$ holds, we write $q_1 \succeq_p q_2$ and say that p *weakly prefers* q_1 to q_2. When there are ties, we denote a preference list in the following form:

$$m_2 : w_3 \ (w_1 \ w_4),$$

which represents that m_2 prefers w_3 best, w_1 and w_4 second with equal preference, but does not want to be matched with w_2. When ties are present, there are three possible definitions of blocking pairs, and accordingly, there are three stability notions, *super-stability*, *strong stability*, and *weak stability* [Irv94]:

- In the super-stability, a blocking pair for a matching M is an acceptable pair $(m, w) \notin M$ such that $w \succeq_m M(m)$ and $m \succeq_w M(w)$.
- In the strong stability, a blocking pair for a matching M is an acceptable pair $(p, q) \notin M$ such that $q \succeq_p M(p)$ and $p \succ_q M(q)$. Note that the person q, who strictly prefers the counterpart p of the blocking pair, may be either a man or a woman.
- In the weak stability, a blocking pair for a matching M is an acceptable pair $(m, w) \notin M$ such that $w \succ_m M(m)$ and $m \succ_w M(w)$.

In the case of super and strong stabilities, there exist instances that do not admit a stable matching. (See [GI89, Man13] for more details.)

Note that in the case of SM, the size of a matching is always n by definition, but it may be less than n in the case of SMTI. In fact, there is an SMTI-instance that admits stable matchings of different sizes, and the problem of finding a stable matching of the maximum size, called *MAX SMTI*, is NP-hard [IMMM99, MII$^+$02]. There are a plenty of approximability and inapproximability results for MAX SMTI. The current best upper bound on the approximation ratio is 1.5 [McD09, Pal14, Kir13] and lower bounds are $33/29 \simeq 1.1379$ assuming P$\neq$NP and $4/3 \simeq 1.3333$ assuming the *unique games conjecture* (*UGC* for short) [Yan07]. There are several attempts to obtain better algorithms (e.g., polynomial-time exact algorithms or polynomial-time approximation algorithms with better approximation ratio) for restricted instances; one of the most natural restrictions is to admit ties in preference lists of only one gender, which we call *SMTI-1T*. *MAX SMTI-1T* (i.e., the problem of finding a maximum cardinality stable matching in SMTI-1T) remains NP-hard, and as for the approximation ratio, the current best upper bound is $1 + 1/e \simeq 1.368$ [LP19] and lower bounds are $21/19 \simeq 1.1052$ assuming P$\neq$NP and $5/4 = 1.25$ assuming UGC [HIMY07, Yan07].

2.1.4 Many-to-One Extension

The *hospitals/residents problem* (*HR* for short) is a many-to-one extension of SMI. The two sets are *residents* and *hospitals*, and a hospital may match more than one residents. Each hospital h has an *upper quota* q. We write the name of a hospital with its quota, such as $h[q]$. A matching is an assignment of residents to hospitals (possibly leaving some residents unassigned), where matched residents and hospitals are in the preference list of each other. Let $M(r)$ be the hospital to which resident r is assigned under a matching M (if it exists), and $M(h)$ be the set of residents assigned to hospital h. A *feasible matching* is a matching such that $|M(h)| \leq q$ for each hospital $h[q]$. We may sometimes call a feasible matching simply a matching when there is no fear of confusion. For a matching M and a hospital $h[q]$, we say that h is *full* if $|M(h)| = q$, *under-subscribed* if $|M(h)| < q$, *over-subscribed* if $|M(h)| > q$, and *empty* if $|M(h)| = 0$. For a matching M, we say that a pair comprising a resident r and a hospital h who include each other in their lists forms a blocking pair for M if the following two conditions are met: (i) r is either unassigned or prefers h to $M(r)$, and (ii) h is under-subscribed or prefers r to one of the residents in $M(h)$.

It is also known that any HR-instance admits at least one stable matching, and one can be found by the GS algorithm in $O(m)$ time [GS62, GI89], where m is the number of acceptable pairs.

2.2 Gale-Shapley Algorithm

Algorithm 1 Gale-Shapley Algorithm [GS62]

Let $M := \emptyset$.
while there is an unassigned resident r in M whose preference list is non-empty **do**
  Let h be the first hospital on r's current preference list.
  Remove h from r's preference list.
  Let $M := M \cup \{(r, h)\}$.
  if h is over-subscribed in M **then**
    Let r' be the worst resident for h in $M(h)$.
    Let $M := M \setminus \{(r', h)\}$.
  end if
end while
Output M.

For completeness, the GS algorithm [GS62] for HR[*1] is shown in Algorithm 1. We call this the *resident-oriented Gale-Shapley* algorithm (*RGS* for short). Since SM and SMI are special cases of HR, this also outputs a stable matching for a given SM or SMI instance. In SM and SMI, this is called the *man-oriented Gale-Shapley* algorithm (*MGS* for short). In addition, this is also referred to as *woman-oriented Gale-Shapley* algorithm (*WGS* for short) when the gender roles are swapped.

2.3 Strategy-Proofness

The stable marriage problem can be seen as a game among participants, who have true preferences in mind, but may submit a falsified preference list hoping to obtain a better partner than the one assigned when true preference lists are used. Formally, let S be a *mechanism*, that is, a mapping from instances to matchings, and we denote $S(I)$ the matching output by S for an instance I. We say that S is a *stable mechanism* if, for any instance I, $S(I)$ is a stable matching for I. For a mechanism S, let I be an instance, M be a matching such that $M = S(I)$, and p be a person. We say that p *has a successful strategy in* I if there is an instance I' in which people except for p have the same preference lists in I and I', and p prefers M' to M (i.e., $M'(p) \succ_p M(p)$ with respect to p's preference list in I), where M' is a matching such that $M' = S(I')$. This situation is interpreted as follows: I is the set of true preference lists, and by submitting a falsified preference list (which changes the set of lists to I'), p can obtain a better partner $M'(p)$. We say that S is a *strategy-proof mechanism* if, when S is used, no person has a successful strategy in any instance. Also we say that S is a *man-strategy-proof mechanism* if, when S is used, no man has a successful strategy in any instance. A *woman-strategy-proof mechanism* is defined analogously. A mechanism is a *one-sided-strategy-proof mechanism* if it is either a man-strategy-proof mechanism or a woman-strategy-proof mechanism.

It is known that there is no strategy-proof stable mechanism for SM [Rot82], which is known as *Roth's impossibility theorem*. By contrast, MGS, described in Algorithm 1, is a man-strategy-proof stable mechanism for SM [Rot82, DF81]. By the symmetry of men and women, WGS is a woman-strategy-proof stable mechanism.

[*1] To be more precise, Algorithm 1 is a modified version of the original algorithm by Gale and Shapley [GS62]. In the original algorithm, all unassigned residents apply to the first hospital on their preference list at the same time, whereas in Algorithm 1, each resident applies one by one.

2.4 Measure of Approximation Algorithms

We say that an algorithm A is an $r(n)$-*approximation algorithm* for a minimization (maximization, respectively) problem if it satisfies $A(x)/opt(x) \leq r(n)$ ($opt(x)/A(x) \leq r(n)$, respectively) for any instance x of size n, where $opt(x)$ and $A(x)$ are the costs (e.g., the size of a stable matching in the case of MAX SMTI) of the optimal and the algorithm's solutions, respectively. The infimum $r(n)$ such that A is an $r(n)$-approximation algorithm is called the approximation ratio of A.

2.5 Related Work

There has been a huge amount of studies on the stable matching problem so that even several books have been published on the subject [Knu76, GI89, RS90, Man13]. In this section, we introduce studies that are closely related to the variants we study in this thesis.

Since Abraham et al.'s work on *SR with incomplete lists* (*SRI* for short) [ABM05], there have been some studies on the problem of finding a matching that is "as stable as possible", including SMI by Biró et al. [BMM10] and HRLQ in Chapter 4 of this thesis. In all three problems, it has been shown that minimizing the number of BPs is not only NP-hard but also hard to approximate within a constant ratio. In response to these hardness results, there have been several studies, such as the computational tractability of SRI [BMM12] and *SR with ties and incomplete lists* (*SRTI*) [CIM19] when the length of the preference list is limited, and the parametrized complexities in SR [CHSY18] and HRLQ [MS20].

Many studies have been conducted to consider other stability notions in HRLQ. For example, Fragiadakis et al. [FIT$^+$16] studied a problem of finding a matching with relaxation of stability, called *envy-freeness*. Krishnaa et al. defined another relaxation of stability called *relaxed stability* and presented an algorithm for finding a matching that satisfies a lower quota under this stability [KLNN20]. Nasre and Nimbhorkar [NN17] gave an algorithm to find a maximum *popular matching*, which is another relaxation of stability. Arulselvan et al. showed that the problem of finding a maximum-weight many-to-one matching in a bipartite graph is NP-hard [ACG$^+$18]. Furthermore, extensions of these results are proposed [GIK$^+$16, HG21, Lim21, MNNR18].

Biró, et al. [BFIM10] also considered lower quotas in HR. In contrast to our model, which requires the lower quotas of all the hospitals to be satisfied, their model allows some hospitals to be closed, i.e., to receive no residents. They proved that the problem of deciding whether there is a feasible solution is NP-complete. Boehmer and Heeger showed parameterized complexity for this problem [BH20]. The problem of finding a *Pareto optimal matching* for the *house allocation problem* has also been studied [MT13, Kam13, CF17, CFP21] under this model.

As another variant of HRLQ, Huang [Hua10] considered *classified stable matchings*, in which each hospital defines a family of subsets of residents and declares upper and lower quotas for each of the subsets. He proved a dichotomy theorem for the problem of deciding the existence of a stable matching; namely, if the subset families satisfy some structural property, then the problem is in P, otherwise, it is NP-complete. Fleiner and Kamiyama [FK12] generalized Huang's result to many-to-many case, where not only hospitals' side but also the residents' side can declare upper and lower quotas. Yokoi [Yok17] further extended the model and showed a polynomial-time algorithm.

In relation to noncrossing matching, Arkin et al. [ABE$^+$09] considered the problem where each participant is given as a point in $\mathbb{R}^d$ $(d \geq 1)$ and the preference lists are set in ascending order of the Euclidean distance between the two points. They gave polynomial-time algorithms to find a stable matching in SR and *SR with ties* (*SRT*). They also considered an extension in SR where the matching is defined as a set of triples instead of a set of pairs, and showed an instance that admits no stable matching. The problem of determining the existence of stable matching was open, but it was recently solved by Chen and Roy [CR21], who showed that it is NP-complete when $d = 2$.

There are some literature studying trade-offs between approximability and strategy-proofness. Krysta et al. [KMRZ19] consider to approximate the size of a Pareto optimal matching in the house allocation problem, where preference lists may include ties. They give upper and lower bounds on the approximation ratio of randomized strategy-proof mechanisms for computing a Pareto optimal matching. Dughmi and Ghosh [DG10] study the *generalized assignment problem* (*GAP* for short) and its variants. Their objective is to maximize the sum of the values of the assigned jobs. They present a strategy-proof $O(\log n)$-approximate mechanism for the GAP, where n represents the number of jobs. The following papers discuss strategy-proofness in the stable matching problem with indifference. Erdil and Ergin [EE08] consider HR where only hospitals' preference lists may have ties. They consider the algorithm that first breaks ties according to a tie-breaking rule τ and then applies RGS (let us call this algorithm GS^{τ}). They give an instance and a tie-breaking rule τ such that GS^{τ} does not produce a resident-optimal stable matching. They also show that seeking for a resident-optimal stable matching loses strategy-proofness, that is, no deterministic resident-optimal stable mechanism can be resident-strategy-proof. Abdulkadiroğlu et al. [APR09] give an evidence to support GS^{τ}. They show that for any tie-breaking rule τ, no resident-strategy-proof mechanism dominates GS^{τ} (with respect to residents).

Chapter 3

Almost Stable Maximum Matchings

In this chapter, we improve the lower bound on the approximation ratio for the problem of finding the maximum matching with the minimum number of blocking pairs in SMI.

Biró et al. [BMM10] defined the following optimization problem, called *MAX SIZE MIN BP SMI*: Given an SMI instance, find a matching that minimizes the number of blocking pairs among all the maximum cardinality matchings. For integers p and q, MAX SIZE MIN BP (p, q)-SMI is the restriction of MAX SIZE MIN BP SMI so that each man's preference list is of length at most p, and each woman's preference list is of length at most q. $p = \infty$ or $q = \infty$ means that the lengths of preference lists are unbounded. Let n be the number of men in an input. Biró et al. [BMM10] showed the following results; (1) MAX SIZE MIN BP (∞, ∞)-SMI is NP-hard and cannot be approximated within the ratio of $n^{1-\varepsilon}$ for any constant $\varepsilon > 0$, unless P=NP; (2) MAX SIZE MIN BP (3, 3)-SMI is APX-hard and cannot be approximated within the ratio of $\frac{3557}{3556} \simeq 1.00028$ unless P=NP; (3) MAX SIZE MIN BP (2, ∞)-SMI is solvable in $O(n^3)$ time.

We improve the hardness of the above (2), namely, we improve the constant $\frac{3557}{3556}$ to $n^{1-\varepsilon}$ for any constant $\varepsilon > 0$. Our reduction uses basically the same idea as the one used in [BMM10] to prove the above (1). In [BMM10], some persons need to have preference lists of unbounded lengths for two reasons: One is for garbage collection, and the other is to create a large gap on the costs between "yes"-instances and "no"-instances. We perform a non-trivial modification of the construction and demonstrate that such gadgets can be replaced by persons with preference lists of length at most three.

3.1 Inapproximability of MAX SIZE MIN BP (3, 3)-SMI

Theorem 1. *MAX SIZE MIN BP* $(3,3)$*-SMI is not approximable within* $n^{1-\varepsilon}$ *where* n *is the number of men in a given instance, for any* $\varepsilon > 0$*, unless* $P = NP$.

We demonstrate a polynomial-time reduction from the same problem as [BMM10], EXACT Maximal Matching (EXACT-MM) restricted to subdivision graphs of cubic graphs, which is NP-complete [O'M07]. A graph G is a subdivision graph if it is obtained from another graph H by replacing each edge (u,v) of H by two edges (u,w) and (w,v) where w is a new vertex. In this problem, we are given a graph G which is a subdivision graph of some cubic graph, as well as a positive integer K, and asked if G contains a maximal matching of size exactly K. Hereafter, we simply say "EXACT-MM" to mean EXACT-MM with the above restrictions.

Given an instance (G,K) of EXACT-MM, we construct an instance I of MAX SIZE MIN BP $(3,3)$-SMI in such a way that (i) I has a perfect matching, (ii) if (G,K) is a "yes"-instance of EXACT-MM, then I has a perfect matching with small number of blocking pairs, and (iii) if (G,K) is a "no"-instance of EXACT-MM, then any perfect matching of I has many blocking pairs.

3.1.1 $\binom{m}{r}$–gadget

Before going to the main body of the reduction, we first introduce the $\binom{m}{r}$–gadget. This gadget plays a role of garbage collection, just as X and Y in the proof of Theorem 1 of [BMM10].

Let X be a set of men of size m where $X = \{x_1, \cdots, x_m\}$, and r $(0 < r \leq m)$ be an integer. The $\binom{m}{r}$–gadget (with respect to X and r), denoted $\mathcal{C}(X,r)$, consists of the following $2mr - r$ men $(\bigcup_{1\leq i\leq m} A_i) \cup (\bigcup_{1\leq j\leq r} C_j)$ and $2mr$ women $(\bigcup_{1\leq i\leq m} B_i) \cup (\bigcup_{1\leq j\leq r} D_j)$.

$$\begin{array}{llll} A_i & = \{a_i^j : 1 \leq j \leq r\}, & B_i & = \{b_i^j : 1 \leq j \leq r\} \quad (1 \leq i \leq m) \\ C_j & = \{c_j^i : 2 \leq i \leq m\}, & D_j & = \{d_j^i : 1 \leq i \leq m\} \quad (1 \leq j \leq r) \end{array}$$

Each person's preference list is defined in Fig. 3.1. A person p's preference list "$p : a\ b\ c$" means that p prefers a, b, and c in this order. For each $x_i \in X$, the unique woman b_i^1 of $\mathcal{C}(X,r)$ who includes x_i in her preference list is referred to as $\mathcal{C}(X,r)[x_i]$.

The role of the gadget $\mathcal{C}(X,r)$ is to receive any subset $X' \subseteq X$ such that $|X'| = r$ without creating many blocking pairs, as formally stated in the following lemmas.

$$
\begin{array}{llll}
a_i^1 & : & d_1^i & b_i^2 & b_i^1 \\
a_i^2 & : & d_2^i & b_i^3 & b_i^2 \\
a_i^3 & : & d_3^i & b_i^4 & b_i^3 \\
\vdots & & & & \\
a_i^{r-1} & : & d_{r-1}^i & b_i^r & b_i^{r-1} \\
a_i^r & : & d_r^i & b_i^r & \\
 & & & & \\
 & & & & \\
c_j^2 & : & d_j^2 & d_j^1 & \\
c_j^3 & : & d_j^3 & d_j^2 & \\
c_j^4 & : & d_j^4 & d_j^3 & \\
\vdots & & & & \\
c_j^{m-1} & : & d_j^{m-1} & d_j^{m-2} & \\
c_j^m & : & d_j^m & d_j^{m-1} &
\end{array}
\qquad
\begin{array}{lllll}
b_i^1 & : & a_i^1 & x_i & \\
b_i^2 & : & a_i^2 & a_i^1 & \\
b_i^3 & : & a_i^3 & a_i^2 & \\
\vdots & & & & \\
b_i^{r-1} & : & a_i^{r-1} & a_i^{r-2} & \\
b_i^r & : & a_i^r & a_i^{r-1} & \\
 & & & & \\
d_j^1 & : & c_j^2 & a_1^j & \\
d_j^2 & : & c_j^3 & c_j^2 & a_2^j \\
d_j^3 & : & c_j^4 & c_j^3 & a_3^j \\
d_j^4 & : & c_j^5 & c_j^4 & a_4^j \\
\vdots & & & & \\
d_j^{m-1} & : & c_j^m & c_j^{m-1} & a_{m-1}^j \\
d_j^m & : & c_j^m & a_m^j &
\end{array}
$$

Fig. 3.1 Preference lists of $\mathcal{C}(X, r)$

In the following lemmas, we assume that each man $x_i \in X$ includes the woman $\mathcal{C}(X, r)[x_i](= b_i^1)$ in his preference list.

Lemma 1. *Let X be a set of men and r be an integer such that $0 < r \leq |X|$. Then, for any $X' \subseteq X$ such that $|X'| = r$, there is a matching M for X and $\mathcal{C}(X, r)$ such that (i) all members of $\mathcal{C}(X, r)$ are matched, (ii) all men in X' are matched with women in $\mathcal{C}(X, r)$ and all men in $X \setminus X'$ are single, and (iii) no person in X is included in a blocking pair, and the number of blocking pairs for M is at most r.*

Proof. Let $X' = \{x_{i_1}, x_{i_2}, \cdots, x_{i_r}\}$ $(1 \leq i_1 < i_2 < \cdots < i_r \leq m)$. We construct the matching M as follows. For each j $(1 \leq j \leq r)$, add the following pairs to M: $(a_{i_j}^k, b_{i_j}^{k+1})$ for $k = 1, \ldots, j-1$; $(a_{i_j}^k, b_{i_j}^k)$ for $k = j+1, \ldots, r$; $(a_{i_j}^j, d_j^{i_j})$; (c_j^{k+1}, d_j^k) for $k = 1, \ldots, i_j - 1$; (c_j^k, d_j^k) for $k = i_j + 1, \ldots, m$; and $(x_{i_j}, b_{i_j}^1)$. (Fig. 3.2 gives an example for a specific i_j.) Also, for each i such that $x_i \in X \setminus X'$, add (a_i^k, b_i^k) for $k = 1, \ldots, r$ to M. It is easy to see that (i) and (ii) are satisfied. Also, it is straightforward to check that blocking pairs are only $(c_j^{i_j}, d_j^{i_j})$ $(1 \leq j \leq r,\ i_j \neq 1)$, and hence there are at most r blocking pairs. $\square$

Lemma 2. *Let X be a set of men and r be an integer such that $0 < r \leq |X|$. Let M be any matching for X and $\mathcal{C}(X, r)$ that matches all members of $\mathcal{C}(X, r)$. Then the number of single men in X is $|X| - r$.*

Proof. This is obvious because any member in $\mathcal{C}(X, r)$ includes only persons in $\mathcal{C}(X, r)$ and X in the preference list, and there are r more women than men in $\mathcal{C}(X, r)$. □

When X is a set of women, we similarly define the $\binom{m}{r}$–gadget by exchanging the roles of men and women.

3.1.2 Main Part of the Reduction

Let $I' = (G, K)$ be an instance of EXACT–MM, where G is a subdivision graph of some cubic graph and K is a positive integer. Since G is a bipartite graph, we can write it as $G = (U, W, E)$ such that $U = \{u_1, \cdots, u_{n_1}\}$ and $W = \{w_1, \cdots, w_{n_2}\}$, where each vertex in U (W, respectively) has degree exactly 2 (3, respectively). (Hence n_1 and n_2 are related as $2n_1 = 3n_2$.) Without loss of generality, we may assume that $K < \min(|U|, |W|)$ and that G has a matching of size K.

As in [BMM10], we give the following definitions: For each $u_i \in U$, let w_{p_i} and w_{q_i} be the two neighbors of u_i in G, where $p_i < q_i$, and for each $w_j \in W$, let u_{r_j}, u_{s_j}, and u_{t_j} be the three neighbors of w_j in G, where $r_j < s_j < t_j$. Also, for each $u_i \in U$ and $w_j \in W$ such that $(u_i, w_j) \in E$, define $\sigma_{j,i} = 1, 2$ according to whether w_j is w_{p_i} or w_{q_i} respectively, and define $\tau_{i,j} = 1, 2, 3$ according to whether u_i is u_{r_j} or u_{s_j} or u_{t_j} respectively. For a given $\varepsilon > 0$, define $B = \lceil \frac{3}{\varepsilon} \rceil$ and $C = (n_1 + n_2)^{B+1}$.

For each vertex $u_i \in U$, we construct $2C + 3$ men and $2C + 2$ women, whose preference lists are given in Fig. 3.3, where men's lists are given in the left and women's lists are given in the right of the figure. We denote $\mathcal{U}(u_i)$ the set of these men and women. Define the set $U^0 = \{u_1^0, \cdots, u_{n_1}^0\}$ (consisting of men, one from each $\mathcal{U}(u_i)$ ($1 \le i \le n_1$)). We then construct $\binom{n_1}{n_1 - K}$–gadget $\mathcal{C}(U^0, n_1 - K)$.

Similarly, for each $w_j \in W$, we construct $3C + 3$ men and $3C + 4$ women, whose preference lists are given in Fig. 3.4. We denote $\mathcal{W}(w_j)$ the set of these men and women. Define the set $W^0 = \{w_1^0, \cdots, w_{n_2}^0\}$ (consisting of women, one from each $\mathcal{W}(w_j)$ ($1 \le j \le n_2$)), and construct $\binom{n_2}{n_2 - K}$–gadget $\mathcal{C}(W^0, n_2 - K)$.

The reduction is now completed. The resulting instance I contains the same number $n = (2 + 2C + 2n_1 - 2K)n_1 + (3 + 3C + 2n_2 - 2K)n_2 + K$ of men and women. Note that each person's preference list is of length at most three. It is not hard to see that the reduction can be performed in time polynomial in the size of I'.

3.1.3 Properties of Gadgets

Before proceeding to the correctness proof, we prove useful lemmas:

Lemma 3. *For any edge $(u_i, w_j) \in E$, we can form a matching M restricted to people in $\mathcal{U}(u_i) \cup \mathcal{W}(w_j)$ so that (i) all people in $\mathcal{U}(u_i) \cup \mathcal{W}(w_j)$ are matched, (ii) M contains at most 2 blocking pairs, and (iii) for any extension of M to a complete matching of I, no person in $\mathcal{U}(u_i) \cup \mathcal{W}(w_j)$ will create a blocking pair with a person not in $\mathcal{U}(u_i) \cup \mathcal{W}(w_j)$.*

Proof. We construct a matching M as follows. Since $(u_i, w_j) \in E$, there are integers k and l such that $\sigma_{j,i} = k$ and $\tau_{i,j} = l$, by the definition of σ and τ. We first add (u_i^k, w_j^l) to M. Next, we consider people in $\mathcal{U}(u_i)$. Add the following pairs to M: $(g_{i,1}^1, z_i^2)$; $(g_{i,1}^s, e_{i,1}^{s-1})$ for $s = 2, \ldots, C$; $(g_{i,2}^1, e_{i,1}^C)$; $(g_{i,2}^s, e_{i,2}^{s-1})$ for $s = 2, \ldots, C$; and $(u_i^0, e_{i,2}^C)$. If $k = 1$, then add (u_i^2, z_i^1), otherwise, add (u_i^1, z_i^1). Finally, we consider people in $\mathcal{W}(w_j)$. Add the following pairs to M: $(v_j^3, h_{j,1}^1)$; $(f_{j,1}^s, h_{j,1}^{s+1})$ for $s = 1, \ldots, C-1$; $(f_{j,1}^C, h_{j,2}^1)$; $(f_{j,2}^s, h_{j,2}^{s+1})$ for $s = 1, \ldots, C-1$; $(f_{j,2}^C, h_{j,3}^1)$; $(f_{j,3}^s, h_{j,3}^{s+1})$ for $s = 1, \ldots, C-1$; and $(f_{j,3}^C, w_j^0)$. If $l = 1$, add (v_j^1, w_j^2) and (v_j^2, w_j^3); if $l = 2$, add (v_j^1, w_j^1) and (v_j^2, w_j^3); if $l = 3$, add (v_j^1, w_j^1) and (v_j^2, w_j^2).

It is straightforward to verify that Condition (i) is satisfied. To see that Conditions (ii) and (iii) hold, observe the following: In $\mathcal{U}(u_i)$, all men of the form $g_{i,t}^s$ for any t, s, and u_i^0 are matched with their first choices. Clearly, these men do not form a blocking pair. Also, women who include only these men in their preference lists cannot form a blocking pair. So, only u_i^1, u_i^2, z_i^1, and z_i^2 can form a blocking pair. If we check the cases of $k = 1$ and $k = 2$, we can verify that at most one blocking pair is possible. Similarly, in $\mathcal{W}(w_j)$, all women of the form $h_{j,t}^s$ for any t, s, and w_j^0 are matched with their first choices. So, only v_j^1, v_j^2, v_j^3, w_j^1, w_j^2, and w_j^3 can be a part of a blocking pair. We may conclude that there is at most one blocking pair by checking cases $l = 1, 2, 3$. □

Lemma 4. *In any matching of I that matches all members of $\mathcal{U}(u_i)$ ($\mathcal{W}(w_j)$, respectively), all people in $\mathcal{U}(u_i)$ ($\mathcal{W}(w_j)$, respectively), except for one man (woman, respectively), are matched among themselves.*

Proof. This is true because any woman in $\mathcal{U}(u_i)$ includes only men in $\mathcal{U}(u_i)$ in her preference list. The case for $\mathcal{W}(w_j)$ can be proved similarly. □

Lemma 5. *Suppose that $(u_i, w_j) \in E$. Let M be any matching of I such that all people in $\mathcal{U}(u_i)$ and $\mathcal{W}(w_j)$ are matched by M and both $(u_i^0, \mathcal{C}(U^0, n_1 - K)[u_i^0])$ and $(w_j^0, \mathcal{C}(W^0, n_2 - K)[w_j^0])$ are in M. Then there are at least C blocking pairs for M (formed by only people in $\mathcal{U}(u_i) \cup \mathcal{W}(w_j)$).*

Proof. Since $(u_i^0, \mathcal{C}(U^0, n_1 - K)[u_i^0]) \in M$ and all people in $\mathcal{U}(u_i)$ are matched in M, by tracing the women's preference lists, the partners of women in $\mathcal{U}(u_i)$ are uniquely determined, namely, $(g_{i,t}^s, e_{i,t}^s) \in M$ for any t, s, and $(u_i^t, z_i^t) \in M$ for $t = 1, 2$. Similarly, we may uniquely determine the pairs within $\mathcal{W}(w_j)$, namely, $(f_{j,t}^s, h_{j,t}^s) \in M$ for any t, s, and $(v_j^t, w_j^t) \in M$ for $t = 1, 2, 3$.

Since $(u_i, w_j) \in E$, there are integers k and l such that $\sigma_{j,i} = k$ and $\tau_{i,j} = l$ by the definition of σ and τ. Then, all $(g_{i,k}^s, h_{j,l}^s)$ $(1 \leq s \leq C)$ are blocking pairs for M. □

3.1.4 Correctness of the Reduction

We first show that I admits a perfect matching. As we have assumed that G has a matching of size K, let it be M'. For each edge $(u_i, w_j) \in M'$, we match people in $\mathcal{U}(u_i)$ and $\mathcal{W}(w_j)$ as in the proof of Lemma 3. There are exactly $n_1 - K$ unmatched vertices in U. Let $\tilde{U^0} (\subseteq U^0)$ consist of men corresponding to these unmatched vertices, i.e. $\tilde{U^0} = \{u_i^0 : u_i \in U \text{ is unmatched in } M'\}$. We match people in $\tilde{U^0}$ and $\binom{n_1}{n_1 - K}$–gadget $\mathcal{C}(U^0, n_1 - K)$ as in the proof of Lemma 1. Also, for each i such that $u_i^0 \in \tilde{U^0}$, match every woman in $\mathcal{U}(u_i)$ to her first choice man. Similarly, there are exactly $n_2 - K$ unmatched vertices in W. Define $\tilde{W^0} (\subseteq W^0)$ as $\tilde{W^0} = \{w_j^0 : w_j \in W \text{ is unmatched in } M'\}$. Again, using the proof of Lemma 1, we match people in $\tilde{W^0}$ and $\binom{n_2}{n_2 - K}$–gadget $\mathcal{C}(W^0, n_2 - K)$. Finally, for each j such that $w_j^0 \in \tilde{W^0}$, match every man in $\mathcal{W}(w_j)$ to his first choice woman. By a careful observation, together with Lemma 1 (i) and (ii) and Lemma 3 (i), it can be verified that the above constriction yields a perfect matching.

Now suppose that G has a maximal matching M' of size K. We construct a perfect matching M of I from M' as described above. We will count the number of blocking pairs for M. By Lemma 1 (iii), $\mathcal{C}(U^0, n_1 - K)$ and $\mathcal{C}(W^0, n_2 - K)$ contain at most $n_1 - K$ and $n_2 - K$ blocking pairs, respectively, and people in these gadgets do not create blocking pairs with people outside respective gadgets. Next we look at gadgets corresponding to vertices. For a pair of vertices u_i and w_j such that $(u_i, w_j) \in M'$, there are at most 2 blocking pairs formed by people in $\mathcal{U}(u_i)$ and $\mathcal{W}(w_j)$ by Lemma 3 (ii). Since $|M'| = K$, there are at most $2K$ such blocking pairs. Also, by Lemma 3 (iii), people in $\mathcal{U}(u_i)$ and $\mathcal{W}(w_j)$ do not form blocking pairs with people outside $\mathcal{U}(u_i) \cup \mathcal{W}(w_j)$. Finally, we consider the gadgets corresponding to the vertices unmatched in M'. Consider the gadget $\mathcal{U}(u_i)$ where u_i is unmatched in M'. By the construction of M, all women in $\mathcal{U}(u_i)$ are matched with their first choices, and cannot form a blocking pair. Hence only the possibility is that a man $g^s_{i,\sigma_{j,i}}$ forms a blocking pair with a woman $h^s_{j,\tau_{i,j}}$ for some j and s. If this is the case, then $(u_i, w_j) \in E$, but by the maximality of M', w_j is matched in M'. Then, by the construction of M, $h^s_{j,\tau_{i,j}}$ must be matched with her first choice and hence $(g^s_{i,\sigma_{j,i}}, h^s_{j,\tau_{i,j}})$ cannot be a blocking pair. Similarly, no people in $\mathcal{W}(w_j)$ where w_j is unmatched in M' cannot form a blocking pair. In summary, the total number of blocking pairs is at most $(n_1 - K) + (n_2 - K) + 2K = n_1 + n_2$.

Conversely, suppose that there is a perfect matching M of I that contains less than C blocking pairs. By Lemma 4, for each $u_i \in U$, all people in $\mathcal{U}(u_i)$, except for one man (which we call a *free-man*), are matched among themselves. Hence there are exactly n_1 free-men. By Lemma 2, M matches exactly $n_1 - K$ men from U^0 with women in $\mathcal{C}(U^0, n_1 - K)$. Clearly, all these men are free-men. So, there are K remaining free-men. We will define *free-women* similarly, and by a similar argument, there are K remaining free-women. Since M is a perfect matching, these men and women are matched together.

Define M' as $M' = \{(u_i, w_j) : (x, y) \in M, x \in \mathcal{U}(u_i), y \in \mathcal{W}(w_j)\}$. If $(x, y) \in M$ for some x ($\in \mathcal{U}(u_i)$) and y ($\in \mathcal{W}(w_j)$), then $(u_i, w_j) \in E$ by the construction of preference lists of I. Also, it is easy to see that x and y are one of K free-men and free-women, respectively, mentioned above. Hence, M' is a matching of G of size K. We show that M' is maximal. For suppose not. Then, there is an edge $(u_i, w_j) \in E$ both of whose endpoints are unmatched in M'. By the construction of M', $u^0_i \in \mathcal{U}(u_i)$ is matched with the woman $\mathcal{C}(U^0, n_1 - K)[u^0_i]$ and $w^0_j \in \mathcal{W}(w_j)$ is matched with the man $\mathcal{C}(W^0, n_2 - K)[w^0_j]$, in M. But then by Lemma 5, M contains at least C blocking pairs, a contradiction. Hence M' is maximal, and we can conclude that if G has no maximal matching of size K, then there is no perfect matching of I with less than $C(= (n_1 + n_2)^{B+1})$ blocking pairs.

Hence, the existence of $(n_1 + n_2)^B$-approximation algorithm for MAX SIZE MIN BP $(3, 3)$-SMI implies a polynomial-time algorithm for EXACT–MM, which implies P=NP. We will show that $(n_1 + n_2)^B \geq n^{1-\varepsilon}$. Recall that

$$n = (2 + 2C + 2n_1 - 2K)n_1 + (3 + 3C + 2n_2 - 2K)n_2 + K, \tag{3.1}$$

by which we obtain $n \leq 5(n_1 + n_2)^{B+2}$, and hence

$$(n_1 + n_2)^B \geq 5^{-\frac{B}{B+2}} n^{\frac{B}{B+2}}. \tag{3.2}$$

We may assume without loss of generality that $n_1 \geq 3$. Since each vertex in U and

W has degree 2 and 3, respectively, $2n_1 = 3n_2$. So, we have $n_1 + n_2 \geq 5$. Also, $K < \min(n_1, n_2)$ by hypothesis. Thus, Equation (3.1) implies that $n \geq 5^B$ and hence, $5^{-\frac{B}{B+2}} \geq n^{-\frac{1}{B+2}}$. Since $B + 2 \geq \frac{3}{\varepsilon}$, Inequality (3.2) implies that $(n_1 + n_2)^B \geq n^{1-\varepsilon}$, which completes the proof of Theorem 1.

3.2 Concluding Remarks

In this chapter, we proved that MAX SIZE MIN BP SMI is not approximable within $n^{1-\varepsilon}$ for any $\varepsilon > 0$ unless P=NP, even when all preference lists are of length at most 3, where n is the number of men in an input.

Our inapproximability proof is artificial; it used a very long chain of preference dependencies to produce a large gap in the number of blocking pairs. Since such long chains seem to occur rarely in the real world, it is interesting future work to consider the computational tractability under the assumption that such long chains do not exist.

$$
\begin{array}{lllll}
a_{i_j}^1 & : & d_1^{i_j} & \boxed{b_{i_j}^2} & b_{i_j}^1 \\
a_{i_j}^2 & : & d_2^{i_j} & \boxed{b_{i_j}^3} & b_{i_j}^2 \\
a_{i_j}^3 & : & d_3^{i_j} & \boxed{b_{i_j}^4} & b_{i_j}^3 \\
\vdots & & & & \\
a_{i_j}^{j-1} & : & d_{j-1}^{i_j} & \boxed{b_{i_j}^j} & b_{i_j}^{j-1} \\
a_{i_j}^j & : & \boxed{d_j^{i_j}} & b_{i_j}^{j+1} & b_{i_j}^j \\
a_{i_j}^{j+1} & : & d_{j+1}^{i_j} & b_{i_j}^{j+2} & \boxed{b_{i_j}^{j+1}} \\
\vdots & & & & \\
a_{i_j}^{r-1} & : & d_{r-1}^{i_j} & b_{i_j}^r & \boxed{b_{i_j}^{r-1}} \\
a_{i_j}^r & : & d_r^{i_j} & \boxed{b_{i_j}^r} &
\end{array}
\qquad
\begin{array}{llll}
b_{i_j}^1 & : & a_{i_j}^1 & \boxed{x_{i_j}} \\
b_{i_j}^2 & : & a_{i_j}^2 & \boxed{a_{i_j}^1} \\
b_{i_j}^3 & : & a_{i_j}^3 & \boxed{a_{i_j}^2} \\
\vdots & & & \\
b_{i_j}^{j-1} & : & a_{i_j}^{j-1} & \boxed{a_{i_j}^{j-2}} \\
b_{i_j}^j & : & a_{i_j}^j & \boxed{a_{i_j}^{j-1}} \\
b_{i_j}^{j+1} & : & \boxed{a_{i_j}^{j+1}} & a_{i_j}^j \\
\vdots & & & \\
b_{i_j}^{r-1} & : & \boxed{a_{i_j}^{r-1}} & a_{i_j}^{r-2} \\
b_{i_j}^r & : & \boxed{a_{i_j}^r} & a_{i_j}^{r-1}
\end{array}
$$

$$
\begin{array}{lllll}
c_j^2 & : & d_j^2 & \boxed{d_j^1} \\
c_j^3 & : & d_j^3 & \boxed{d_j^2} \\
c_j^4 & : & d_j^4 & \boxed{d_j^3} \\
\vdots & & & \\
c_j^{i_j-1} & : & d_j^{i_j-1} & \boxed{d_j^{i_j-2}} \\
c_j^{i_j} & : & d_j^{i_j} & \boxed{d_j^{i_j-1}} \\
c_j^{i_j+1} & : & \boxed{d_j^{i_j+1}} & d_j^{i_j} \\
\vdots & & & \\
c_j^{m-1} & : & \boxed{d_j^{m-1}} & d_j^{m-2} \\
c_j^m & : & \boxed{d_j^m} & d_j^{m-1}
\end{array}
\qquad
\begin{array}{lllll}
d_j^1 & : & \boxed{c_j^2} & a_1^j & \\
d_j^2 & : & \boxed{c_j^3} & c_j^2 & a_2^j \\
d_j^3 & : & \boxed{c_j^4} & c_j^3 & a_3^j \\
d_j^4 & : & \boxed{c_j^5} & c_j^4 & a_4^j \\
\vdots & & & & \\
d_j^{i_j-1} & : & \boxed{c_j^{i_j}} & c_j^{i_j-1} & a_{i_j-1}^j \\
d_j^{i_j} & : & c_j^{i_j+1} & c_j^{i_j} & \boxed{a_{i_j}^j} \\
d_j^{i_j+1} & : & c_j^{i_j+2} & \boxed{c_j^{i_j+1}} & a_{i_j+1}^j \\
\vdots & & & & \\
d_j^{m-1} & : & c_j^m & \boxed{c_j^{m-1}} & a_{m-1}^j \\
d_j^m & : & \boxed{c_j^m} & a_m^j &
\end{array}
$$

Fig. 3.2 A part of the matching described in the proof of Lemma 1

$$
\begin{array}{lllll}
u_i^1 & : & z_i^1 & w_{p_i}^{\tau_{i,p_i}} & \\
u_i^2 & : & z_i^1 & z_i^2 & w_{q_i}^{\tau_{i,q_i}} \\
g_{i,1}^1 & : & z_i^2 & h_{p_i,\tau_{i,p_i}}^1 & e_{i,1}^1 \\
g_{i,1}^2 & : & e_{i,1}^1 & h_{p_i,\tau_{i,p_i}}^2 & e_{i,1}^2 \\
g_{i,1}^3 & : & e_{i,1}^2 & h_{p_i,\tau_{i,p_i}}^3 & e_{i,1}^3 \\
\vdots & & & & \\
g_{i,1}^{C-1} & : & e_{i,1}^{C-2} & h_{p_i,\tau_{i,p_i}}^{C-1} & e_{i,1}^{C-1} \\
g_{i,1}^{C} & : & e_{i,1}^{C-1} & h_{p_i,\tau_{i,p_i}}^{C} & e_{i,1}^{C} \\
g_{i,2}^1 & : & e_{i,1}^C & h_{q_i,\tau_{i,q_i}}^1 & e_{i,2}^1 \\
g_{i,2}^2 & : & e_{i,2}^1 & h_{q_i,\tau_{i,q_i}}^2 & e_{i,2}^2 \\
g_{i,2}^3 & : & e_{i,2}^2 & h_{q_i,\tau_{i,q_i}}^3 & e_{i,2}^3 \\
\vdots & & & & \\
g_{i,2}^{C-1} & : & e_{i,2}^{C-2} & h_{q_i,\tau_{i,q_i}}^{C-1} & e_{i,2}^{C-1} \\
g_{i,2}^{C} & : & e_{i,2}^{C-1} & h_{q_i,\tau_{i,q_i}}^{C} & e_{i,2}^{C} \\
u_i^0 & : & e_{i,2}^C & \mathcal{C}(U^0, n_1 - K)[u_i^0] &
\end{array}
\qquad
\begin{array}{llll}
z_i^1 & : & u_i^1 & u_i^2 \\
z_i^2 & : & u_i^2 & g_{i,1}^1 \\
e_{i,1}^1 & : & g_{i,1}^1 & g_{i,1}^2 \\
e_{i,1}^2 & : & g_{i,1}^2 & g_{i,1}^3 \\
e_{i,1}^3 & : & g_{i,1}^3 & g_{i,1}^4 \\
\vdots & & & \\
e_{i,1}^{C-1} & : & g_{i,1}^{C-1} & g_{i,1}^C \\
e_{i,1}^{C} & : & g_{i,1}^{C} & g_{i,2}^1 \\
e_{i,2}^1 & : & g_{i,2}^1 & g_{i,2}^2 \\
e_{i,2}^2 & : & g_{i,2}^2 & g_{i,2}^3 \\
e_{i,2}^3 & : & g_{i,2}^3 & g_{i,2}^4 \\
\vdots & & & \\
e_{i,2}^{C-1} & : & g_{i,2}^{C-1} & g_{i,2}^C \\
e_{i,2}^{C} & : & g_{i,2}^{C} & u_i^0
\end{array}
$$

Fig. 3.3 Preference lists of $\mathcal{U}(u_i)$

$$
\begin{array}{llll}
v_j^1 & : & w_j^1 & w_j^2 \\
v_j^2 & : & w_j^2 & w_j^3 \\
v_j^3 & : & w_j^3 & h_{j,1}^1 \\
f_{j,1}^1 & : & h_{j,1}^1 & h_{j,1}^2 \\
f_{j,1}^2 & : & h_{j,1}^2 & h_{j,1}^3 \\
f_{j,1}^3 & : & h_{j,1}^3 & h_{j,1}^4 \\
\vdots & & & \\
f_{j,1}^{C-1} & : & h_{j,1}^{C-1} & h_{j,1}^C \\
f_{j,1}^C & : & h_{j,1}^C & h_{j,2}^1 \\
f_{j,2}^1 & : & h_{j,2}^1 & h_{j,2}^2 \\
f_{j,2}^2 & : & h_{j,2}^2 & h_{j,2}^3 \\
f_{j,2}^3 & : & h_{j,2}^3 & h_{j,2}^4 \\
\vdots & & & \\
f_{j,2}^{C-1} & : & h_{j,2}^{C-1} & h_{j,2}^C \\
f_{j,2}^C & : & h_{j,2}^C & h_{j,3}^1 \\
f_{j,3}^1 & : & h_{j,3}^1 & h_{j,3}^2 \\
f_{j,3}^2 & : & h_{j,3}^2 & h_{j,3}^3 \\
f_{j,3}^3 & : & h_{j,3}^3 & h_{j,3}^4 \\
\vdots & & & \\
f_{j,3}^{C-1} & : & h_{j,3}^{C-1} & h_{j,3}^C \\
f_{j,3}^C & : & h_{j,3}^C & w_j^0
\end{array}
\qquad
\begin{array}{lllll}
w_j^1 & : & v_j^1 & u_{r_j}^{\sigma_{j,r_j}} & \\
w_j^2 & : & v_j^1 & v_j^2 & u_{s_j}^{\sigma_{j,s_j}} \\
w_j^3 & : & v_j^2 & v_j^3 & u_{t_j}^{\sigma_{j,t_j}} \\
h_{j,1}^1 & : & v_j^3 & g_{r_j,\sigma_{j,r_j}}^1 & f_{j,1}^1 \\
h_{j,1}^2 & : & f_{j,1}^1 & g_{r_j,\sigma_{j,r_j}}^2 & f_{j,1}^2 \\
h_{j,1}^3 & : & f_{j,1}^2 & g_{r_j,\sigma_{j,r_j}}^3 & f_{j,1}^3 \\
\vdots & & & & \\
h_{j,1}^{C-1} & : & f_{j,1}^{C-2} & g_{r_j,\sigma_{j,r_j}}^{C-1} & f_{j,1}^{C-1} \\
h_{j,1}^C & : & f_{j,1}^{C-1} & g_{r_j,\sigma_{j,r_j}}^C & f_{j,1}^C \\
h_{j,2}^1 & : & f_{j,1}^C & g_{s_j,\sigma_{j,s_j}}^1 & f_{j,2}^1 \\
h_{j,2}^2 & : & f_{j,2}^1 & g_{s_j,\sigma_{j,s_j}}^2 & f_{j,2}^2 \\
h_{j,2}^3 & : & f_{j,2}^2 & g_{s_j,\sigma_{j,s_j}}^3 & f_{j,2}^3 \\
\vdots & & & & \\
h_{j,2}^{C-1} & : & f_{j,2}^{C-2} & g_{s_j,\sigma_{j,s_j}}^{C-1} & f_{j,2}^{C-1} \\
h_{j,2}^C & : & f_{j,2}^{C-1} & g_{s_j,\sigma_{j,s_j}}^C & f_{j,2}^C \\
h_{j,3}^1 & : & f_{j,2}^C & g_{t_j,\sigma_{j,t_j}}^1 & f_{j,3}^1 \\
h_{j,3}^2 & : & f_{j,3}^1 & g_{t_j,\sigma_{j,t_j}}^2 & f_{j,3}^2 \\
h_{j,3}^3 & : & f_{j,3}^2 & g_{t_j,\sigma_{j,t_j}}^3 & f_{j,3}^3 \\
\vdots & & & & \\
h_{j,3}^{C-1} & : & f_{j,3}^{C-2} & g_{t_j,\sigma_{j,t_j}}^{C-1} & f_{j,3}^{C-1} \\
h_{j,3}^C & : & f_{j,3}^{C-1} & g_{t_j,\sigma_{j,t_j}}^C & f_{j,3}^C \\
w_j^0 & : & f_{j,3}^C & \multicolumn{2}{l}{\mathcal{C}(W^0, n_2 - K)[w_j^0]}
\end{array}
$$

Fig. 3.4 Preference lists of $\mathcal{W}(w_j)$

Chapter 4

The Hospitals/Residents Problem with Lower Quotas

In this chapter, we study an extension of HR where each hospital declares not only an upper bound but also a *lower* bound on the number of residents it accepts. Consequently, a feasible matching must satisfy the condition that the number of residents assigned to each hospital is between its upper and lower quotas. We call this problem *HR with lower quota* (*HRLQ*). In HRLQ, stable matchings do not always exist. However, it is easy to decide whether or not there is a stable matching for a given instance, since in HR the number of students a specific hospital h receives is identical for any stable matching (this is a part of the well-known *Rural Hospitals Theorem* [GS85]). Namely, if this number satisfies the upper and lower bound conditions of all the hospitals, it is a feasible (and stable) matching, and otherwise, no stable matching exists. In case there is no stable matching, it is natural to seek for a matching that is "as stable as possible".

We first consider the problem of minimizing the number of blocking pairs, which is quite popular in the literature (e.g., [KMV94, ABM05, BMM10]). As we will show in Section 4.1, it seems that the introduction of the lower quota intrinsically increases the difficulty of the problem. Actually, we show that this problem is NP-hard and cannot be approximated within a factor of $(|H| + |R|)^{1-\varepsilon}$ for any positive constant ε unless P=NP, where H and R denote the sets of hospitals and residents, respectively. This inapproximability result holds even if all the preference lists are complete (i.e., include all the members of the other side), all the hospitals have the same preference list, (e.g., determined by scores of exams and known as the *master list* [IMS08]), and all the hospitals have an upper quota of one. On the positive side, we give a polynomial-time $(|H| + |R|)$–approximation algorithm, which shows that the above inapproximability result is almost tight.

We then tackle this hardness from two different angles. First, we give an exponential-time exact algorithm with running time $O((|H||R|)^{t+1})$, where t is the number of blocking pairs in an optimal solution. Note that this is a polynomial-time algorithm when t is a constant. Second, we consider another measure for optimization criteria, i.e., the number of residents who are involved in blocking pairs. We show that this problem is still NP-hard, but give a quadratic improvement, i.e., we give a polynomial-time $\sqrt{|R|}$-approximation algorithm. We also give an instance showing that our analysis is tight up to a constant factor. Furthermore, we show that if our problem has a constant approximation factor, then the Densest k-Subgraph Problem (DkS) has a constant approximation factor also. Note that the best known approximation factor of DkS has long been $|V|^{1/3}$ [FKP01] in spite of extensive studies, and was improved to $|V|^{1/4+\epsilon}$ for an arbitrary positive constant ϵ [BCC+10]. The reduction is somewhat tricky; it is done through a third problem, called the Minimum Coverage Problem (MinC), and exploits the best approximation algorithm for DkS. MinC is relatively less studied and only NP-hardness was previously known for its complexity [Vin07]. As a by-product, our proof gives a similar inapproximability result for MinC (Lemma 17), which is of independent interest.

4.1 Preliminaries

An instance of HRLQ consists of a set R of residents and a set H of hospitals. Each hospital h has lower and upper quotas, p and q ($p \leq q$), respectively. We sometimes say that the quota of h is $[p, q]$, or h is a $[p, q]$-hospital. For simplicity, we also write the name of a hospital with its quotas, such as $h[p, q]$. Each member (resident or hospital) has a preference list that orders a subset of the members of the other party.

Minimum-blocking-pair hospitals/residents problem with lower quota (*Min-BP HRLQ* for short) is the problem of finding a feasible matching with the minimum number of blocking pairs. *Min-BP 1ML-HRLQ* ("1ML" standing for "1 master list") is the restriction of Min-BP HRLQ so that in a given instance, preference lists of all the hospitals are identical. *0-1 Min-BP HRLQ* is the restriction of Min-BP HRLQ where a quota of each hospital is either $[0, 1]$ or $[1, 1]$. *0-1 Min-BP 1ML-HRLQ* is Min-BP HRLQ with both "1ML" and "0-1" restrictions.

Minimum-blocking-resident hospitals/residents problem with lower quota (*Min-BR HRLQ* for short) is the problem of finding a feasible matching with the minimum number of blocking residents. *Min-BR 1ML-HRLQ*, *0-1 Min-BR HRLQ*, and *0-1 Min-BR 1ML-HRLQ* are defined similarly.

We assume without loss of generality that the number of residents is at least the sum of the lower quotas of all the hospitals, since otherwise there is no feasible matching. We call this assumption the *number of residents assumption* (or the *NR-assumption* for short). Also, we impose the following restriction, the *complete list restriction* (or the *CL-restriction* for short), to guarantee existence of a feasible solution: every hospital with a positive lower quota must have a complete preference list, and every resident's list must include all such hospitals. (We remark in Section 4.4 that allowing arbitrarily incomplete preference lists makes the problem extremely hard.)

As a starting example, consider n residents and $n+1$ hospitals, whose preference lists and quotas are as follows. Here, "$\cdots$" in the residents' preference lists denotes an arbitrary order of the remaining hospitals.

$$
\begin{array}{lllll}
r_1 & : & h_1 & h_{n+1} & \cdots \\
r_2 & : & h_1 & h_2 & h_n & \cdots \\
r_3 & : & h_2 & h_1 & h_3 & \cdots \\
r_4 & : & h_3 & h_1 & h_4 & \cdots \\
\vdots & & & & \\
r_i & : & h_{i-1} & h_1 & h_i & \cdots \\
\vdots & & & & \\
r_n & : & h_{n-1} & h_1 & h_n & \cdots
\end{array}
\qquad
\begin{array}{rllll}
h_1[0,1] & : & r_1 & r_2 & \cdots & r_n \\
h_2[1,1] & : & r_1 & r_2 & \cdots & r_n \\
\vdots & & & & & \\
h_n[1,1] & : & r_1 & r_2 & \cdots & r_n \\
h_{n+1}[1,1] & : & r_1 & r_2 & \cdots & r_n
\end{array}
$$

Note that we have n $[1,1]$-hospitals all of which have to be filled by the n residents. Therefore, let us modify the instance by removing the $[0,1]$-hospital h_1 and apply the Gale-Shapley algorithm (in this chapter we always use RGS shown in Section 2.2, which is the resident-oriented version). Then the resulting matching is $M_1 = \{(r_1,h_{n+1}),(r_2,h_2),(r_3,h_3),\cdots,(r_n,h_n)\}$, which contains at least n blocking pairs (between h_1 and every resident). However, the matching $M_2 = \{(r_1,h_{n+1}),(r_2,h_n),(r_3,h_2),(r_4,h_3),\ldots,(r_n,h_{n-1})\}$ contains only three blocking pairs, namely (r_1,h_1), (r_2,h_1), and (r_2,h_2).

4.2 Minimum-Blocking-Pair HRLQ

In this section, we consider the problem of minimizing the number of blocking pairs.

4.2.1 Inapproximability

We first prove a strong inapproximability result for the restricted subclass.

Theorem 2. *For any positive constant ε, there is no polynomial-time $(|H|+|R|)^{1-\varepsilon}$-approximation algorithm for 0-1 Min-BP 1ML-HRLQ unless P=NP, even if all the preference lists are complete.*

Proof. We demonstrate a polynomial-time reduction from the well-known NP-complete problem *Vertex Cover* (*VC* for short) [GJ79]. In VC, we are given a graph $G = (V, E)$ and a positive integer $K \leq |V|$, and asked if there is a subset C of vertices of G such that $|C| \leq K$, which contains at least one endpoint of each edge. Let $I_0 = (G_0, K_0)$ be an instance of VC where $G_0 = (V_0, E_0)$ and K_0 is a positive integer. Define $n = |V_0|$. For a constant ε, define $c = \lceil \frac{8}{\varepsilon} \rceil$, $B_1 = n^c$, and $B_2 = n^c - |E_0|$.

We construct the instance I of 0-1 Min-BP 1ML-HRLQ from I_0. The set of residents is $R = C \cup F \cup S$, and the set of hospitals is $H = V \cup T \cup X$. Each set is defined as follows:

$$
\begin{aligned}
C &= \{c_i \mid 1 \leq i \leq K_0\} \\
F &= \{f_i \mid 1 \leq i \leq n - K_0\} \\
S^{i,j} &= \{s^{i,j}_{0,a} \mid 1 \leq a \leq B_2\} \cup \{s^{i,j}_{1,a} \mid 1 \leq a \leq B_2\} \quad ((v_i, v_j) \in E_0, i < j) \\
S &= \bigcup S^{i,j} \\
V &= \{v_i \mid 1 \leq i \leq n\} \\
T^{i,j} &= \{t^{i,j}_{0,a} \mid 1 \leq a \leq B_2\} \cup \{t^{i,j}_{1,a} \mid 1 \leq a \leq B_2\} \quad ((v_i, v_j) \in E_0, i < j) \\
T &= \bigcup T^{i,j} \\
X &= \{x_i \mid 1 \leq i \leq B_1\}
\end{aligned}
$$

Each hospital in X has a quota [0,1], and other hospitals have a quota [1,1]. Note that $|C| + |F| = |V| (= n)$ and $|S| = |T| (= 2|E_0|B_2)$. Since any hospital in $V \cup T$ has a quota [1,1], any feasible matching is a one-to-one correspondence between R and $V \cup T$, and every hospital in X must be empty. Note that $|H| = n + 2|E_0|B_2 + B_1$ and $|R| = n + 2|E_0|B_2$; hence $|H| + |R| = 2n + 4|E_0|B_2 + B_1 = 2n - 4|E_0|^2 + (4|E_0| + 1)n^c < n^2 + 4n^{c+2} + n^c \leq 6n^{c+2}$, which is polynomial in n.

Next, we construct preference lists. Fig. 4.1 shows preference lists of residents, where $[[V]]$ (respectively $[[X]]$) denotes a total order of elements in V (respectively X) in an increasing order of indices. The symbol "$\cdots$" denotes an arbitrarily ordered list of all the other hospitals that do not explicitly appear in the list.

$$
\begin{array}{llllllll}
c_i & : & [[V]] & [[X]] & \cdots & & & (1 \le i \le K_0) \\
f_i & : & [[V]] & [[X]] & \cdots & & & (1 \le i \le n - K_0) \\
s^{i,j}_{0,1} & : & t^{i,j}_{0,1} & v_i & t^{i,j}_{1,1} & [[X]] & \cdots & ((v_i, v_j) \in E_0, i < j) \\
s^{i,j}_{0,2} & : & t^{i,j}_{0,2} & v_i & t^{i,j}_{0,3} & [[X]] & \cdots & ((v_i, v_j) \in E_0, i < j) \\
 & & \vdots & & & & & \\
s^{i,j}_{0,B_2-1} & : & t^{i,j}_{0,B_2-1} & v_i & t^{i,j}_{0,B_2} & [[X]] & \cdots & ((v_i, v_j) \in E_0, i < j) \\
s^{i,j}_{0,B_2} & : & t^{i,j}_{0,B_2} & v_i & t^{i,j}_{0,1} & [[X]] & \cdots & ((v_i, v_j) \in E_0, i < j) \\
s^{i,j}_{1,1} & : & t^{i,j}_{0,2} & v_j & t^{i,j}_{1,2} & [[X]] & \cdots & ((v_i, v_j) \in E_0, i < j) \\
s^{i,j}_{1,2} & : & t^{i,j}_{1,2} & v_j & t^{i,j}_{1,3} & [[X]] & \cdots & ((v_i, v_j) \in E_0, i < j) \\
 & & \vdots & & & & & \\
s^{i,j}_{1,B_2-1} & : & t^{i,j}_{1,B_2-1} & v_j & t^{i,j}_{1,B_2} & [[X]] & \cdots & ((v_i, v_j) \in E_0, i < j) \\
s^{i,j}_{1,B_2} & : & t^{i,j}_{1,B_2} & v_j & t^{i,j}_{1,1} & [[X]] & \cdots & ((v_i, v_j) \in E_0, i < j)
\end{array}
$$

Fig. 4.1 Preference lists of residents

Preference lists of hospitals are identical and are obtained from the master list "$[[C]]$ $[[S]]$ $[[F]]$". Here, $[[C]]$ and $[[F]]$ are as before a total order of all the residents in C and F, respectively, in an increasing order of indices. $[[S]]$ is a total order of $[[S^{i,j}]]$ $((v_i, v_j) \in E_0, i < j)$ in any order, where $[[S^{i,j}]] = s^{i,j}_{1,1}\ s^{i,j}_{0,1}\ s^{i,j}_{0,2}\ \cdots\ s^{i,j}_{0,B_2}\ s^{i,j}_{1,2}\ \cdots\ s^{i,j}_{1,B_2}$.

Now the reduction is completed. Before showing the correctness proof, we will see some properties of the reduced instance. For a resident r and a hospital h, if h appears to the right of the $[[X]]$-part of r's list, we call (r, h) a *prohibited pair*.

Lemma 6. *If a matching M contains a prohibited pair, then the number of blocking pairs in M is at least B_1.*

Proof. Suppose that a matching M contains a prohibited pair (r, h). By the definition of prohibited pairs, r prefers any hospital $x \in X$ to h. On the other hand, recall that any hospital $x \in X$ is empty in any feasible matching, and hence, under-subscribed.

Hence, (r, x) is a blocking pair for every $x \in X$. Since $|X| = B_1$, the proof is completed. □

Now, recall that for each edge $(v_i, v_j) \in E_0$ $(i < j)$, there are the set of residents $S^{i,j}$ and the set of hospitals $T^{i,j}$. We call this pair of sets a $g_{i,j}$-*gadget*, and write it as $g_{i,j} = (S^{i,j}, T^{i,j})$. For each gadget $g_{i,j}$, let us define two perfect matchings between $S^{i,j}$ and $T^{i,j}$ as follows:

$$\begin{aligned} M^0_{i,j} = \{ & (s^{i,j}_{0,1}, t^{i,j}_{0,1}), (s^{i,j}_{0,2}, t^{i,j}_{0,2}), \ldots, (s^{i,j}_{0,a}, t^{i,j}_{0,a}), \ldots, (s^{i,j}_{0,B_2-1}, t^{i,j}_{0,B_2-1}), \\ & (s^{i,j}_{0,B_2}, t^{i,j}_{0,B_2}), (s^{i,j}_{1,1}, t^{i,j}_{1,2}), (s^{i,j}_{1,2}, t^{i,j}_{1,3}), \ldots, \\ & (s^{i,j}_{1,a}, t^{i,j}_{1,a+1}), \ldots, (s^{i,j}_{1,B_2-1}, t^{i,j}_{1,B_2}), (s^{i,j}_{1,B_2}, t^{i,j}_{1,1})\}, \text{ and} \end{aligned}$$

$$\begin{aligned} M^1_{i,j} = \{ & (s^{i,j}_{0,1}, t^{i,j}_{1,1}), (s^{i,j}_{0,2}, t^{i,j}_{0,3}), \ldots, (s^{i,j}_{0,a}, t^{i,j}_{0,a+1}), \ldots, (s^{i,j}_{0,B_2-1}, t^{i,j}_{0,B_2}), \\ & (s^{i,j}_{0,B_2}, t^{i,j}_{0,1}), (s^{i,j}_{1,1}, t^{i,j}_{0,2}), (s^{i,j}_{1,2}, t^{i,j}_{1,2}), \ldots, \\ & (s^{i,j}_{1,a}, t^{i,j}_{1,a}), \ldots, (s^{i,j}_{1,B_2-1}, t^{i,j}_{1,B_2-1}), (s^{i,j}_{1,B_2}, t^{i,j}_{1,B_2})\}. \end{aligned}$$

Fig. 4.2 shows $M^0_{i,j}$ and $M^1_{i,j}$ on preference lists of $S^{i,j}$, where the [[X]]-part and thereafter are omitted.

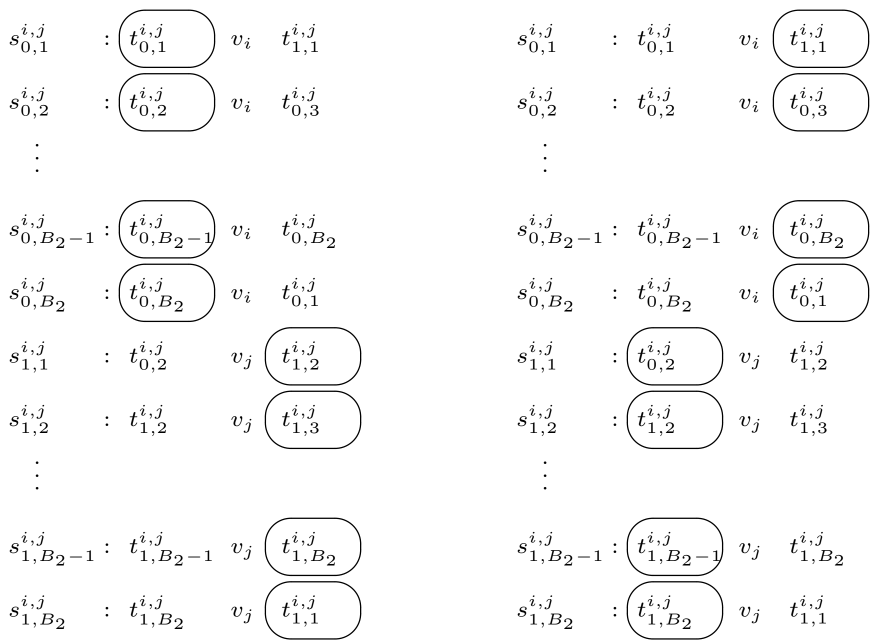

Fig. 4.2 Matchings $M^0_{i,j}$ (left) and $M^1_{i,j}$ (right)

Lemma 7. *For a gadget $g_{i,j} = (S^{i,j}, T^{i,j})$, $M^0_{i,j}$ and $M^1_{i,j}$ are the only perfect matchings between $S^{i,j}$ and $T^{i,j}$ that do not include a prohibited pair. Furthermore, each of*

$M^0_{i,j}$ and $M^1_{i,j}$ contains only one blocking pair (r,h) such that $r \in S^{i,j}$ and $h \in T^{i,j}$. (Hereafter, we simply state this as a "blocking pair between $S^{i,j}$ and $T^{i,j}$".)

Proof. Construct a bipartite graph $G_{i,j}$, where each vertex set is $S^{i,j}$ and $T^{i,j}$, and there is an edge between $r(\in S^{i,j})$ and $h(\in T^{i,j})$ if and only if (r,h) is not a prohibited pair. One can see that $G_{i,j}$ is a cycle of length $4B_2$. Hence there are only two perfect matchings between $S^{i,j}$ and $T^{i,j}$, and they are actually $M^0_{i,j}$ and $M^1_{i,j}$. Also, it is easy to check that $M^0_{i,j}$ contains only one blocking pair $(s^{i,j}_{1,1}, t^{i,j}_{0,2})$, and $M^1_{i,j}$ contains only one blocking pair $(s^{i,j}_{0,1}, t^{i,j}_{0,1})$. □

We are now ready to show the gap for inapproximability.

Lemma 8. *If I_0 is a "yes" instance of VC, then I has a solution with at most $n^2+|E_0|$ blocking pairs.*

Proof. Suppose that G_0 has a vertex cover of size at most K_0. If its size is less than K_0, add arbitrary vertices to make the size exactly K_0, which is, of course, still a vertex cover. Let this vertex cover be $V_{0c}(\subseteq V_0)$, and let $V_{0f} = V_0 \setminus V_{0c}$. For convenience, we use V_{0c} and V_{0f} also to denote the sets of corresponding hospitals.

We construct a matching M of I according to V_{0c}. First, match each resident in C with each hospital in V_{0c}, and each resident in F with each hospital in V_{0f}, in an arbitrary way. Since $|C \cup F| = |V| = n$, there are at most n^2 blocking pairs between $C \cup F$ and V.

For each gadget $g_{i,j} = (S^{i,j}, T^{i,j})$ $((v_i, v_j) \in E_0, i < j)$, we use one of the two matchings in Lemma 7. Since V_{0c} is a vertex cover, either v_i or v_j is included in V_{0c}. If v_i is in V_{0c}, use $M^1_{i,j}$, otherwise, use $M^0_{i,j}$. It is then easy to see that there is no blocking pair between $S^{i,j}$ and $H \setminus T^{i,j}$ or $R \setminus S^{i,j}$ and $T^{i,j}$. Also, as proved in Lemma 7, there is only one blocking pair between $S^{i,j}$ and $T^{i,j}$ in either case.

Therefore, the number of blocking pairs is at most n^2 between $C \cup F$ and V, and exactly $|E_0|$ within $g_{i,j}$-gadgets, and hence $n^2 + |E_0|$ in total, which completes the proof. □

Lemma 9. *If I_0 is a "no" instance of VC, then any solution of I has at least B_1 blocking pairs.*

Proof. Suppose that I admits a matching M with less than B_1 blocking pairs. We show that I_0 has a vertex cover of size K_0.

First, recall that any feasible matching must be a one-to-one correspondence between R and $V \cup T$. Also, by Lemma 6, if M contains a prohibited pair then there are at least B_1 blocking pairs, contradicting the assumption. Thus, M does not contain a prohibited pair. Since $|C \cup F| = |V|$ and any resident $r \in C \cup F$ includes only V to the left of the $[[X]]$-part in the preference list, M must include a perfect matching between $C \cup F$ and V.

Next, consider a gadget $g_{i,j} = (S^{i,j}, T^{i,j})$ and observe the preference lists of $S^{i,j}$. Since v_i and v_j are matched with residents in $C \cup F$, for M to contain no prohibited pairs, all residents in $S^{i,j}$ must be matched with hospitals in $T^{i,j}$. By Lemma 7, there are only two possibilities, namely, $M^0_{i,j}$ and $M^1_{i,j}$, and either matching admits one blocking pair within each $g_{i,j}$. Hence there are $|E_0|$ such blocking pairs for all $g_{i,j}$-gadgets.

Suppose that the matching between $S^{i,j}$ and $T^{i,j}$ is $M^0_{i,j}$. Then, if the hospital v_j is matched with a resident in F, there are B_2 blocking pairs between v_j and $s^{i,j}_{1,1}, \ldots, s^{i,j}_{1,B_2}$. Then, we have $|E_0| + B_2 = B_1$ blocking pairs, contradicting the assumption. Hence, v_j must be matched with a resident in C. On the other hand, suppose that the matching for $g_{i,j}$ is $M^1_{i,j}$. If the hospital v_i is matched with a resident in F, again there are B_2 blocking pairs, between v_i and $s^{i,j}_{0,1}, \ldots, s^{i,j}_{0,B_2}$. Therefore, v_i must be matched with a resident in C. Namely, for each edge (v_i, v_j), either v_i or v_j is matched with a resident in C. Hence, the collection of vertices whose corresponding hospitals are matched with residents in C is a vertex cover of size K_0. This completes the proof. □

Finally, we estimate the gap obtained by Lemmas 8 and 9. As observed previously, $n^c < |H| + |R| \leq 6n^{c+2}$. Hence, $B_1/(n^2 + |E_0|) \geq n^c/2n^2 = 8n^{c+2}2^{-4}n^{-4} \geq 8n^{c+2}n^{-8} > (|H|+|R|)^{1-\frac{8}{c}} \geq (|H|+|R|)^{1-\varepsilon}$. Hence a polynomial-time $(|H|+|R|)^{1-\varepsilon}$-approximation algorithm for 0-1 Min-BP 1ML-HRLQ solves VC, implying P=NP. □

4.2.2 Approximability

The following theorem shows that an almost tight upper bound can be achieved by a simple approximation algorithm for the general class.

Theorem 3. *There is a polynomial-time* $(|H| + |R|)$*–approximation algorithm for Min-BP HRLQ.*

Proof. Before showing an algorithm, we introduce some terms used to describe the algorithm. In a matching M, define a *deficiency of a hospital* $h_i[p_i, q_i]$ to be $\max\{p_i - |M(h_i)|, 0\}$. We say that a hospital $h_i[p_i, q_i]$ has *surplus* if h_i satisfies $|M(h_i)| - p_i > 0$. The following simple algorithm (Algorithm 2) achieves the approximation ratio of $|H| + |R|$.

Algorithm 2 An $(|H| + |R|)$–approximation algorithm for Min-BP HRLQ

1: Consider an instance I of Min-BP HRLQ as an instance of HR by ignoring lower quotas. Then apply the Gale-Shapley algorithm to I and obtain a matching M.
2: If there is an unassigned resident in M, output M.
3: Move residents from hospitals with surplus to the hospitals with positive deficiencies in an arbitrary way (but so as not to create new positive deficiency) to fill all the deficiencies. Then output the modified matching.

Obviously, Algorithm 2 runs in polynomial time. Note that because of the NR-assumption and the CL-restriction, line 3 is executable, namely, there are sufficiently many residents in hospitals with surplus to fill all the deficiencies.

We first show that if a matching M is returned at line 2, M is an optimal solution. Let r be a resident unassigned in M. Then r must have been rejected by all the hospitals with a positive lower quota, since r includes all such hospitals in the list because of the CL-restriction. Therefore, any such hospital is full in M, that is, M is a feasible matching. Hence, we obtain a feasible stable matching, which is clearly an optimal solution.

In the following, we assume that all the residents are assigned in M. Let k be the sum of the deficiencies over all the hospitals. Then, k residents are moved. Suppose that resident r is moved from hospital h to another hospital. Then, it is easy to see that a new blocking pair includes either r or h since only they can become worse off. Hence, there arise at most $|H| + |R|$ new blocking pairs per resident movement and there are at most $k(|H| + |R|)$ blocking pairs in total. On the other hand, we show in the following that if there are k deficiencies in M, an optimal solution contains at least k blocking pairs. These observations give an $(|H| + |R|)$-approximation upper bound.

Let M_{opt} be an optimal solution. For convenience, we think that a hospital $h_i[p_i, q_i]$ has q_i distinct positions, each of which can receive at most one resident. Define the bipartite graph $G_{M,M_{opt}} = (V_R, V_H, E)$ as follows: Each vertex in V_R corresponds to a resident in I, and each vertex in V_H to a position (so, $|V_H| = \sum q_i$). If resident r is assigned by M to hospital h, then in $G_{M,M_{opt}}$, we include an edge (called an M-edge) between $r \in V_R$ and some position $p \in V_H$ of h, and similarly, if resident r is assigned by M_{opt} to hospital h, then we include an edge (called an M_{opt}-edge) between r and some position p of h, so that a single vertex p receives at most one M-edge and at most one M_{opt}-edge. Without loss of generality, we may assume that if a resident r is assigned to the same hospital by M and M_{opt}, r is assigned to the same position p. (In this case, we have parallel edges between r and p.) Hence, if a resident is assigned to different positions by M and M_{opt}, then he/she is assigned to different hospitals. Note that each vertex of $G_{M,M_{opt}}$ has degree at most two.

Note that M_{opt} satisfies all the lower quotas, while M has k deficiencies. This means that there are at least k vertices in V_H that are matched in M_{opt} but not in M. It is easy to see that these k vertices are endpoints of k disjoint paths in $G_{M,M_{opt}}$, in which M_{opt}-edges and M-edges appear alternately. By a standard argument (for example, see the proof of Lemma 4.2 of [HIMY07]), we can show that each such path contains at least one blocking pair for M or M_{opt}, but all of them are for M_{opt} because M is stable. This completes the proof. □

4.2.3 Exponential-Time Exact Algorithm

Our goal in this section is to design non-trivial exponential-time algorithms by using the parameter t denoting the optimal cost, i.e., the number of blocking pairs in an optimal solution. Perhaps a natural idea is to set the number c_i of residents ($p_i \leq c_i \leq q_i$) assigned to each hospital $h_i[p_i, q_i]$, so that the sum of c_i's over all the hospitals is equal to the number of residents. However, there is no obvious way of setting such c_i's rather than exhaustive search, which will result in blow-ups of the computation time even if t is small. Furthermore, even if we would be able to find suitable setting of c_i's, we are still not sure how to assign the residents to hospitals optimally (see the example of Section 4.1).

However, once we guess a set of blocking pairs included in a matching, we can easily test whether it is a correct guess or not by using the Gale-Shapley algorithm and the Rural Hospitals theorem. Based on this observation, we will show an $O((|H||R|)^{t+1})$-time exact algorithm for Min-BP HRLQ.

Theorem 4. *There is an $O((|H||R|)^{t+1})$-time exact algorithm for Min-BP HRLQ, where t is the number of blocking pairs in an optimal solution of a given instance.*

Proof. For a given integer $k > 0$, the following procedure $A(k)$ finds a solution (i.e., a matching between residents and hospitals) whose cost (i.e., the number of blocking pairs) is at most k if any. Starting from $k = 1$, our algorithm (Algorithm E) runs $A(k)$ until it finds a solution, by increasing the value of k one by one. $A(k)$ is quite simple, for which the following informal discussion suffices.

Let I be a given instance. First, we guess a set B of k blocking pairs. Since there are at most $|H||R|$ pairs, there are at most $(|H||R|)^k$ choices of B. For each $(r, h) \in B$, we remove h from r's preference list (and r from h's list). Let I' be the modified instance. We then apply the Gale-Shapley algorithm to I'. If all the lower quotas are satisfied, then it is a desired solution, otherwise, we fail and proceed to the next guess.

We show that Algorithm E runs correctly. Consider any optimal solution M_{opt} and consider the execution of $A(k)$ for $k = t$ for which our current guess B contains exactly the t blocking pairs of M_{opt}. Then, it is not hard to see that M_{opt} is stable in I' and satisfies all the lower quotas. Then by the Rural Hospitals theorem, any stable matching for I' satisfies all the lower quotas. Hence if we apply the Gale-Shapley algorithm to I', we find a matching M that satisfies all the lower quotas. Note that M has no blocking pair in I'. Then, M has at most t blocking pairs in the original instance I because, when a removed hospital h is returned back to the preference list of r, only (r, h) can be a new blocking pair.

Finally, we bound the time-complexity of Algorithm E. For each k, we apply the Gale-Shapley algorithm to at most $(|H||R|)^k$ instances, where each execution can be done in time $O(|H||R|)$. Therefore, the time-complexity is $O((|H||R|)^{k+1})$ for each k. Since we find a solution when k is at most t, the whole time-complexity is at most $\Sigma_{k=1}^{t} O((|H||R|)^{k+1}) = O((|H||R|)^{t+1})$. □

4.3 Minimum-Blocking-Resident HRLQ

In this section, we consider the problem of minimizing the number of blocking residents.

4.3.1 NP-hardness

We first show a hardness result.

Theorem 5. *Min-BR 1ML-HRLQ is NP-hard even if all the preference lists are* *complete.*

Proof. We will show a polynomial-time reduction from the NP-complete problem CLIQUE [GJ79]. In CLIQUE, we are given a graph $G = (V, E)$ and a positive integer $K \leq |V|$, and asked if G contains a complete graph with K vertices as a subgraph.

Let $I_0 = (G_0, K_0)$ be an instance of CLIQUE where $G_0 = (V_0, E_0)$ and $0 < K_0 \leq |V_0|$. We will construct an instance I of Min-BR 1ML-HRLQ. Let $n = |V_0|$, $m = |E_0|$, and B be a positive integer such that $B > 2K_0$. Let $R = C \cup E$ be the set of residents and $H = V \cup \{x\}$ be the set of hospitals of I. Each set is defined as $C = \{c_i \mid 1 \leq i \leq K_0\}$, $E = \{e^k_{i,j} \mid (v_i, v_j) \in E_0, 1 \leq k \leq B\}$, and $V = \{v_i \mid 1 \leq i \leq n\}$. (There is a one-to-one correspondence between the set V of hospitals and the set V_0 of vertices, so we use the same symbol v_i to refer to both vertex and the corresponding hospital.)

Corresponding to each edge $(v_i, v_j) \in E_0$, there are B residents $e^k_{i,j} (1 \leq k \leq B)$. We will call them *residents associated with* (v_i, v_j). Preference lists and quotas are given in Fig. 4.3. For a set X, "$[X]$" means an arbitrarily (but fixed) ordered list of the members in X, and "..." means an arbitrarily ordered list of all the other hospitals that do not appear explicitly in the list. Note that all the preference lists are complete, and all the hospitals have the same preference list.

c_i	:	$[V]$	x			$(1 \leq i \leq K_0)$
$e^k_{i,j}$	:	v_i	v_j	x	$\dots$	$((v_i, v_j) \in E_0, 1 \leq k \leq B)$
$v_i[0,1]$	:	$[C]$	$[E]$			$(1 \leq i \leq n)$
$x[mB, mB]$	:	$[C]$	$[E]$			

Fig. 4.3 Preference lists of residents and preference lists and quotas of hospitals

Lemma 10. *If I_0 is a "yes" instance of CLIQUE, then there is a feasible matching of I having at most $(m - \binom{K_0}{2})B + K_0$ blocking residents.*

Proof. Suppose that G_0 has a clique V'_0 of size K_0. We will construct a matching M of I from V'_0. We assign all the residents in C to the hospitals in V'_0 in an arbitrary way, and all the residents in E to the hospital x. Since V'_0 is a clique, $(v_i, v_j) \in E_0$ for any pair of $v_i, v_j \in V'_0 (i \neq j)$. There are B residents $e^k_{i,j}$ $(1 \leq k \leq B)$ associated with the edge (v_i, v_j). These residents are assigned to the hospital x inferior to the hospitals v_i and v_j in M, but the hospitals v_i and v_j are assigned residents in C, better than $e^k_{i,j}$. Hence all $e^k_{i,j}$ are non-blocking residents. There are $B\binom{K_0}{2}$ such residents $e^k_{i,j}$ and the total number of residents is $mB + K_0$. Hence there are at most $(m - \binom{K_0}{2})B + K_0$ blocking residents in M. □

Lemma 11. *For a matching X of I, let cost(X) be the number of blocking residents of X. For an arbitrary feasible matching M of I, there is a feasible matching M' of I such that (i) M' assigns every resident in C to a hospital in V and (ii) cost$(M') \leq$ cost$(M) + K_0$.*

Proof. First, if some residents are unassigned in M, we modify M by assigning them to arbitrary hospitals. This is possible because all the preference lists are complete and the number of residents is at most the sum of the upper quotas. Clearly, this does not increase the cost. Let $C_x = \{c \mid c \in C, M(c) = x\}$ and $E_v = \{e \mid e \in E, M(e) \in V\}$. Then, $|C_x| = |E_v|$ since $|M(x)| = |C_x| + (|E| - |E_v|)$ and $|M(x)| = mB = |E|$ by the lower quota of x. If C_x is empty, we are done because we can let $M' = M$. Hence, suppose that C_x is nonempty. Let M' be a matching obtained by M by exchanging assigned hospitals between C_x and E_v arbitrarily. Then M' is feasible and the following (1)–(3) are easy to verify:

(1) Any resident in $C \setminus C_x$ does not change its assigned hospital, and no hospital in V becomes worse off. Therefore, no new blocking resident arises from $C \setminus C_x$. (2) Any resident r in C_x is a blocking resident in M because r is assigned to x and there is a hospital in V that receives a resident from E_v. Therefore, no new blocking resident arises from C_x. (3) For the same reason as (1), no new blocking resident arises from $E \setminus E_v$.

Hence, only residents in E_v can newly become blocking residents. Since $|E_v| = |C_x| \leq |C| = K_0$, we have that cost$(M') \leq$ cost$(M) + K_0$. □

Lemma 12. *If I_0 is a “no” instance of CLIQUE, then any feasible matching of I contains at least $(m - \binom{K_0}{2} + 1)B - K_0$ blocking residents.*

Proof. Suppose that there is a matching M of I that contains less than $(m - \binom{K_0}{2} + 1)B - K_0$ blocking residents. We will show that G_0 contains a clique of size K_0. We first construct a matching M' using Lemma 11. Then M' contains less than $(m - \binom{K_0}{2} + 1)B$ blocking residents, and any resident in C is assigned to a hospital in V. Note that every resident in E is now assigned to x since x's lower quota is

$mB = |E|$. Define $V_0' \subseteq V_0$ be the set of vertices corresponding to the assigned hospitals in V. Clearly, $|V_0'| = K_0$. We claim that V_0' is a clique.

Recall that there are $mB + K_0$ residents. Since we assume that there are less than $(m - \binom{K_0}{2} + 1)B$ blocking residents, there are more than $K_0 + \binom{K_0}{2}B - B$ non-blocking residents, and since $|C| = K_0$, there are more than $\binom{K_0}{2}B - B$ non-blocking residents in E. Consider the following partition of E into B subsets: $E_k = \{e^k_{i,j} \mid (v_i, v_j) \in E_0\}$ $(1 \leq k \leq B)$. Then the above observation on the number of non-blocking residents in E implies that there is a k such that E_k contains at least $\binom{K_0}{2}$ non-blocking residents. Since every resident in E is assigned to x, only $e^k_{i,j}$ such that both v_i and v_j are in V_0' can be non-blocking. This means that any pair of vertices in V_0' causes such a non-blocking resident, implying that V_0' is a clique. □

Because $B > 2K_0$, we have $(m - \binom{K_0}{2} + 1)B - K_0 > (m - \binom{K_0}{2})B + K_0$. Hence by Lemmas 10 and 12, Min-BR 1ML-HRLQ is NP-hard. □

We can prove the NP-hardness for more restricted case using the following Lemma 13. Since the same reduction will be used in the approximability part (Section 4.3.2), we state the lemma in a stronger form than is needed here.

Lemma 13. *If there is a polynomial-time α–approximation algorithm for 0-1 Min-BR HRLQ, then there is a polynomial-time α–approximation algorithm for Min-BR HRLQ.*

Proof. We give a polynomial-time approximation preserving reduction from Min-BR HRLQ to 0-1 Min-BR HRLQ. Let I be an instance of Min-BR HRLQ. We construct an instance I' of 0-1 Min-BR HRLQ in polynomial time: The set of residents of I' is the same as that of I. Corresponding to each hospital $h_i[p_i, q_i]$ of I, I' contains p_i hospitals $h_{i,1}, \cdots, h_{i,p_i}$ with quota $[1, 1]$, and $q_i - p_i$ hospitals $h_{i,p_i+1}, \cdots, h_{i,q_i}$ with quota $[0, 1]$. For any j, the preference list of a hospital $h_{i,j}$ of I' is the same as that of a hospital h_i of I. The preference list of a resident r of I' is constructed from the preference list of the corresponding resident in I by replacing h_i by $h_{i,1} \ \cdots \ h_{i,q_i}$ for each hospital h_i of I. Without loss of generality, we can assume that $q_i \leq |R|$ for each i. Hence I' can be constructed in polynomial time.

From a feasible matching M' for I', it is easy to construct a feasible matching M for I; just adding (r, h_i) to M for each $(r, h_{i,j}) \in M'$. Let *cost*, *cost*$'$, *opt* and *opt*$'$ be the costs of M, M', the optimal costs of I and I', respectively. In order to complete the proof, we must show that $\frac{cost}{opt} \leq \frac{cost'}{opt'}$. To this end, it is enough to show (i) *cost* $\leq$ *cost*$'$, and (ii) *opt*$'$ $\leq$ *opt*. For (i), it is easy to verify that if r is a blocking resident for M, then so is r for M' too. For (ii), we show that from (any) matching X for I, we can construct a matching X' for I' without increasing the cost. Consider a hospital h_j. Let $r_{j,1}, r_{j,2}, \cdots, r_{j,|X(h_j)|}$ be the residents in $X(h_j)$ and suppose that h_j prefers these residents in this order. We construct a matching X' by adding $(r_{j,k}, h_{j,k})$ to X' for all k and j. Again, it is easy to see that X' is feasible for I' and if r is a blocking resident for X', then r is also a blocking resident for X. □

Corollary 1. *0-1 Min-BR 1ML-HRLQ is NP-hard even if all the preference lists are complete.*

Proof. Note that the reduction in the proof of Lemma 13 preserves the "1ML" property and the completeness of the preference lists. Then the corollary is immediate from Theorem 5 and Lemma 13. □

4.3.2 Approximability

For the approximability, we note that Algorithm 2 in the proof of Theorem 3 does not work. For example, consider the instance introduced in Section 4.1. If we apply the Gale-Shapley algorithm, resident r_i is assigned to h_i for each i, and we need to move r_1 to h_{n+1}. However since h_1 becomes empty, all the residents become blocking residents. On the other hand, the optimal cost is 2 as we have seen there. Thus the approximation ratio becomes as bad as $\Omega(|R|)$.

Theorem 6. *There is a polynomial-time* $\sqrt{|R|}$*–approximation algorithm for Min-BR HRLQ.*

We know by Lemma 13 that it is enough to attack 0-1 Min-BR HRLQ. Hence we give a $\sqrt{|R|}$–approximation algorithm for 0-1 Min-BR HRLQ (Lemma 15) to prove Theorem 6. In 0-1 Min-BR HRLQ, the number of residents assigned to each hospital is at most one. Hence, for a matching M, we sometimes abuse the notation $M(h)$ to denote the resident assigned to h (if any) although it was originally defined as the *set* of residents assigned to h.

Algorithm

To describe the idea behind our algorithm, recall again Algorithm 2 presented in the proof of Theorem 3: First, apply the Gale-Shapley algorithm to a given instance

I and obtain a matching M. Next, move residents arbitrarily from assigned $[0,1]$-hospitals to empty $[1,1]$-hospitals. Suppose that in the course of the execution of Algorithm 2, we move a resident r from a $[0,1]$-hospital h to an empty $[1,1]$-hospital. Then, of course r creates a blocking pair with h, but some other residents may also create blocking pairs with h because h becomes empty. Hence, consider the following modification. First, set the upper quota of a $[0,1]$-hospital h to ∞ and apply the Gale-Shapley algorithm. Then, all residents who "wish" to go to h actually go there. Hence, even if we move all such residents to other hospitals, only the moved residents can become blocking residents. By doing this, we can bound the number of blocking residents by the number (given by the function g introduced below) of those moving residents. In the above example, we extended the upper quota of only one hospital, but in fact, we may need to select two or more hospitals to select sufficiently many residents to be sent to other hospitals so as to make the matching feasible. However, at the same time, this number should be kept minimum to guarantee the quality of the solution.

As mentioned above, we define $g(h,h)$: For an instance I of HR, suppose that we extend the upper quota of hospital h to ∞ and find a stable matching of this new instance. Define $g(h,h)$ as the number of residents who are assigned to h in this stable matching. Recall that this quantity does not depend on the choice of the stable matching by the Rural Hospitals theorem [GS85]. Extend $g(h,h)$ to $g(A,B)$ for $A, B \subseteq H$ such that $g(A,B)$ denotes the number of residents assigned to hospitals in A when we change upper quotas of all the hospitals in B to ∞.

We now propose Algorithm 3 for 0-1 Min-BR HRLQ. The idea is to find a small number of residents (*victims*) to be moved, and construct a feasible matching M^* in which only the victims are blocking. First we apply the Gale-Shapley algorithm to a given instance I while ignoring the lower quotas of I and obtain a matching M_s. The matching M_s is used to find non-empty $[0,1]$-hospitals (denoted $H'_{0,1}$ in the description of Algorithm 3) from which the victims will be selected. Next, we estimate the popularity of the hospital h in $H'_{0,1}$ using $g(h,h)$ defined above, and select a certain number of least popular hospitals S from $H'_{0,1}$ (we will later show that $H'_{0,1}$ is large enough to select S). We then apply the Gale-Shapley algorithm again while setting the upper quotas of hospitals in S to ∞ and obtain a matching M_∞. The residents who came to hospitals in S are victims and we move these residents to the empty $[1,1]$-hospitals to obtain the final solution M^* (we will later show that there are enough number of victims to fill the empty $[1,1]$-hospitals). We can show that the number of victims is small enough because we have selected less popular hospitals to S.

We will introduce notations used to describe Algorithm 3 formally. Let I be a given instance. Define $H_{p,q}$ to be the set of $[p,q]$-hospitals of I. Recall from Section 4.2 that the deficiency of a hospital is the shortage of the assigned residents from its lower quota (with respect to the matching obtained by the Gale-Shapley algorithm). Now define the *deficiency of the instance* I as the sum of the deficiencies of all the hospitals of I, and denote it $D(I)$. Since we are considering 0-1 Min-BR HRLQ, $D(I)$ is exactly the number of empty $[1,1]$-hospitals.

Algorithm 3 A $\sqrt{|R|}$–approximation algorithm for 0-1 Min-BR HRLQ

1: Apply the Gale-Shapley algorithm to I by ignoring the lower quotas. Let M_s be the obtained matching. Compute the deficiency $D(I)$.

2: $H'_{0,1} := \{h \mid M_s(h) \neq \emptyset, h \in H_{0,1}\}$. (If $M_s(h) = \emptyset$, then residents never come to h in the following lines 3 and 4.)

3: Compute $g(h,h)$ for each $h \in H'_{0,1}$ by using the Gale-Shapley algorithm.

4: From $H'_{0,1}$, select $D(I)$ hospitals with smallest $g(h,h)$ values (ties are broken arbitrarily). Let S be the set of these hospitals. Extend the upper quotas of all hospitals in S to ∞, and run the Gale-Shapley algorithm. Let M_∞ be the obtained matching.

5: In M_∞, move residents who are assigned to hospitals in S arbitrarily to empty hospitals to make the matching feasible. (We first make $[1,1]$-hospitals full. This is possible because of the NR-assumption and the CL-restriction. If there is a hospital in S still having two or more residents, then send surplus residents arbitrarily to empty $[0,1]$-hospitals, or simply make them unassigned if there is no $[0,1]$-hospital to send them to.) Output the resulting matching M^*.

We first prove the following property of the original HR problem.

Lemma 14. *Let I_0 be an instance of HR, and h be any hospital. Let I_1 be a modification of I_0 so that only the upper quota of h is increased by 1. Let M_i be a stable matching of I_i for each $i \in \{0,1\}$. Then, (i) $|M_0(h)| \leq |M_1(h)|$, and (ii) $\forall h' \in H \backslash \{h\}$, $|M_0(h')| \geq |M_1(h')|$.*

Proof. If M_0 is stable for I_1, then we are done, so suppose not. We will construct a stable matching for I_1 by successive modifications starting from M_0. Because M_0 is stable for I_0, if M_0 has blocking pairs for I_1, then all of them involve h. Let r be the resident such that (r, h) is a blocking pair and there is no blocking pair (r', h) such that h prefers r' to r. If we assign r to h (possibly by canceling the previous assignment of r if r was assigned in M_0), all the blocking pairs including h are removed. If no new blocking pairs arise, again we are done. Otherwise, r must be previously assigned to some hospital, say h', and all the new blocking pairs involve h'. We then choose the resident r', most preferred by h' among all the blocking residents, and assign r' to h'. We continue this operation until there arise no new blocking pairs. This procedure eventually terminates because each iteration improves exactly one resident. By the termination condition, the resulting matching is stable for I_1. Note that by this procedure, only h can gain one more resident, and at most one hospital may lose one resident. By the Rural Hospitals theorem, the number of residents assigned to each hospital is the same in M_1 and the current matching. This completes the proof. □

Obviously, Algorithm 3 runs in polynomial time. We show that Algorithm 3 runs correctly, namely that the output matching M^* satisfies the quotas. To do so, it suffices to show the following conditions

$$|H'_{0,1}| \geq D(I) \tag{4.1}$$

and

$$|\{r \mid M_\infty(r) \in S\}| \geq |\{h \mid h \in H_{1,1}, M_\infty(h) = \emptyset\}| \tag{4.2}$$

so that lines 4 and 5 are executable, respectively.

For Equation (4.1), let N_1 be the number of residents assigned to hospitals in $H_{1,1}$ in M_s. Then $|M_s| = |H'_{0,1}| + N_1$ and $D(I) = |H_{1,1}| - N_1$. We can assume that all the residents are assigned in M_s since otherwise, we already have a feasible stable matching (as explained in the proof of Theorem 3) and therefore $|M_s| = |R|$. From these equations, we have $|H'_{0,1}| = D(I)+|R|-|H_{1,1}|$. By the NR-assumption, it follows that $|R| \geq |H_{1,1}|$, from which we have $|H'_{0,1}| \geq D(I)$ as required. For Equation (4.2), it suffices to show that the number N_2 of residents assigned to $S \cup H_{1,1}$ in M_∞ is at least the number of hospitals in $H_{1,1}$, i.e., $|H_{1,1}|$. Note that empty hospitals in M_s are also empty in M_∞ by Lemma 14. Therefore, the number $\overline{N_2}$ of residents assigned to hospitals in $H \setminus (S \cup H_{1,1})$ in M_∞ is at most the number of hospitals in $H'_{0,1} \setminus S$. Thus $\overline{N_2} \leq |H'_{0,1}| - |S|$ and $N_2 = |R| - \overline{N_2} \geq |R| - (|H'_{0,1}| - |S|)$. By the definition of $D(I)$, we have that $|H'_{0,1}|+|H_{1,1}| = |R|+D(I)$. Thus, $N_2 \geq |R|-(|R|+D(I)-|H_{1,1}|-|S|) = |H_{1,1}|$ (recall that $|S| = D(I)$).

Analysis of the Approximation Ratio

Lemma 15. *The approximation ratio of Algorithm 3 is at most* $\sqrt{|R|}$.

Proof. Let I be a given instance of 0-1 Min-BR HRLQ and let f_{opt} and f_{alg} be the costs of an optimal solution and the solution obtained by Algorithm 3, respectively. First, note that any resident r who is assigned to a hospital $h \in H \setminus S$ in M_∞ prefers no hospital in S to h, since otherwise, r and such a hospital (in S) form a blocking pair for M_∞, a contradiction (recall that the upper quota of any hospital in S is ∞). Therefore, even if we move residents from hospitals in S at line 5, no unmoved resident becomes a blocking resident. Thus only moved residents can be blocking residents and

$$f_{alg} \leq g(S,S). \tag{4.3}$$

We then give a lower bound on the optimal cost. To do so, recall the proof of Theorem 3, where it is shown that any optimal solution for instance I of Min-BP HRLQ has at least $D(I)$ blocking pairs. It should be noted that those $D(I)$ blocking pairs do not have any common resident. Thus we have

$$f_{opt} \geq D(I). \tag{4.4}$$

Now here is our key lemma to evaluate the approximation ratio.

Lemma 16. *In line 3 of Algorithm 3, there are at least $D(I)$ different hospitals $h \in H'_{0,1}$ such that $g(h,h) \leq f_{opt}$.*

The proof will be given in a moment. By this lemma, we have $g(h,h) \leq f_{opt}$ for any $h \in S$, since at line 4 of Algorithm 3, we select $D(I)$ hospitals with the smallest $g(h,h)$ values. This implies that

$$\sum_{h \in S} g(h,h) \leq D(I) f_{opt}. \tag{4.5}$$

Also, by Lemma 14, we have

$$g(h,S) \leq g(h,h) \tag{4.6}$$

for any $h \in S$. Hence, by Equations (4.3), (4.6), (4.5) and (4.4), we have

$$f_{alg} \leq g(S,S) = \sum_{h \in S} g(h,S) \leq \sum_{h \in S} g(h,h) \leq D(I) f_{opt} \leq (f_{opt})^2.$$

Therefore, we have that $\sqrt{f_{alg}} \leq f_{opt}$, and hence $\frac{f_{alg}}{f_{opt}} \leq \sqrt{f_{alg}} \leq \sqrt{|R|}$, completing the proof of Lemma 15. □

Proof of Lemma 16. Let h be a hospital satisfying the condition of the lemma. In order to calculate $g(h, h)$ in line 3, we construct a stable matching, say M_h for the instance $I_\infty(h)$ in which the upper quota of h is changed to ∞. We do not know what kind of matching M_h is, but in the following, we show that there is a stable matching, say M_2, for $I_\infty(h)$ such that $|M_2(h)| \leq f_{opt}$. Matchings M_h and M_2 may be different matchings, but we can guarantee that $|M_h(h)| = |M_2(h)| \leq f_{opt}$ by the Rural Hospitals theorem. A bit trickily, we construct M_2 from an optimal matching.

Let M_{opt} be an optimal solution of I (which of course we do not know). Let R_b and R_n be the sets of blocking residents and non-blocking residents for M_{opt}, respectively. Then $|R_b| = f_{opt}$ by definition. We modify M_{opt} as follows: Take any resident $r \in R_b$. If r is unassigned, we do nothing. Otherwise, force r to be unassigned. Then there may arise new blocking pairs involving residents in R_n. Let BP_1 be the set of such new blocking pairs. Note that all of the blocking pairs in BP_1 include the hospital h' to which r was assigned. Among the residents involved in BP_1, we select the resident r' who is most preferred by h' and assign r' to h'. Then, all the blocking pairs in BP_1 disappear. However, there may arise new blocking pairs (BP_2) involving residents in R_n, and all the blocking pairs in BP_2 include the hospital h'' to which r' was assigned. In a similar way as the proof of Lemma 14, we continue to move residents until no new blocking resident arises from R_n (but this time, we move only residents in R_n as explained above). We do this for all the residents in R_b, and let M_1 be the resulting matching.

The following properties (4.7) and (4.8) are immediate:

$$\text{There are at least } f_{opt} \text{ unassigned residents in } M_1, \tag{4.7}$$

since residents in R_b are unassigned in M_1.

$$\text{Residents in } R_n \text{ are non-blocking for } M_1. \tag{4.8}$$

We prove the following properties:

$$\text{There are at most } f_{opt} \text{ empty } [1,1]\text{-hospitals in } M_1. \tag{4.9}$$

Define $H' = \{h \mid h \in H'_{0,1} \text{ and } h \text{ is empty in } M_1\}$. Then

$$|H'| \geq D(I). \tag{4.10}$$

For (4.9), note that all the $[1,1]$-hospitals are full in M_{opt}. It is easy to see that, in the above procedure for each $r \in R_b$, at most one assigned hospital is made empty. Since $|R_b| = |f_{opt}|$, the number of such hospitals is at most $|f_{opt}|$ and hence the claim holds.

For (4.10), let H_1 be the set of hospitals assigned in M_1. We have that

$$H' = H'_{0,1} \setminus (H_1 \cap H_{0,1}) \tag{4.11}$$

by the definition of H', and that

$$|H'_{0,1}| = |R| + D(I) - |H_{1,1}| \tag{4.12}$$

by the definition of $D(I)$. Also, the above property (4.7) implies that $|R| - |H_1| \geq f_{opt}$ and (4.9) implies that $|H_{1,1}| - |H_1 \cap H_{1,1}| \leq f_{opt}$, from which we have that

$$\begin{aligned} |H_1 \cap H_{0,1}| &= |H_1| - |H_1 \cap H_{1,1}| \\ &\leq (|R| - f_{opt}) + (f_{opt} - |H_{1,1}|) \\ &= |R| - |H_{1,1}|. \end{aligned} \tag{4.13}$$

From Equations (4.11) to (4.13), we have $|H'| \geq |H'_{0,1}| - |H_1 \cap H_{0,1}| \geq (|R| + D(I) - |H_{1,1}|) - (|R| - |H_{1,1}|) = D(I)$, as required.

Let h be an arbitrary hospital in H'. We show that $g(h,h) \leq f_{opt}$. Then, this completes the proof of Lemma 16 because $H' \subseteq H'_{0,1}$ and (4.10). Since h is empty in M_1, residents in R_n are still non-blocking for M_1 in $I_\infty(h)$ (whose definition is in the beginning of this proof) by the property (4.8). Now, choose any resident r from R_b, and apply the Gale-Shapley algorithm to $I_\infty(h)$ starting from M_1. This execution starts from the proposal by r, and at the end, nobody in $R_n \cup \{r\}$ is a blocking resident for $I_\infty(h)$. Since hospitals assigned in M_1 never become empty, and since unassigned residents in R_n never become assigned, h receives at most one resident. If we do this for all the residents in R_b, the resulting matching M_2 is stable for $I_\infty(h)$, and h is assigned at most $|R_b| = f_{opt}$ residents. As mentioned previously, this implies $g(h,h) \leq f_{opt}$. □

Tightness of the Analysis

We give an instance of 0-1 Min-BR HRLQ for which Algorithm 3 produces a solution of cost $|R| - \sqrt{|R|}$ but the optimal cost is at most $2\sqrt{|R|}$. Namely, the analysis of Lemma 15 is tight up to a constant factor.

Let $R = C \cup D \cup E$ and $H = A \cup B \cup X$, where $C = \{c_i \mid 1 \le i \le n\}$, $D = \{d_{i,j} \mid 1 \le i \le n, 1 \le j \le n-2\}$, $E = \{e_i \mid 1 \le i \le n\}$, $A = \{a_i \mid 1 \le i \le n\}$, $B = \{b_i \mid 1 \le i \le n\}$, and $X = \{x_i \mid 1 \le i \le n^2 - n\}$. The preference lists of residents are

$$\begin{array}{llllll} c_i & : & a_i \quad b_i & [[X]] & \cdots & (1 \le i \le n) \\ d_{i,j} & : & b_i \quad [[X]] & \cdots & & (1 \le i \le n, 1 \le j \le n-2) \\ e_i & : & b_i \quad [[A]] & [[X]] & \cdots & (1 \le i \le n) \end{array}$$

and the preference lists and quotas of hospitals are

$$\begin{array}{lllll} a_i[0,1] & : & c_i & \cdots & (1 \le i \le n) \\ b_i[0,1] & : & d_{i,1} & \cdots & (1 \le i \le n) \\ x_i[1,1] & : & \cdots & & (1 \le k \le n^2 - n) \end{array}$$

where $[[X]]$ denotes $x_1 \ \cdots \ x_{n^2-n}$ and $[[A]]$ denotes $a_1 \ \cdots \ a_n$. "$\cdots$" denotes an arbitrarily ordered list of the members that do not appear explicitly. Note that all the preference lists are complete. The deficiency of this instance is n. If we set the upper quota of a_i to ∞, then $n+1$ residents $c_i, e_1, e_2, \ldots, e_n$ are assigned to a_i, so $g(a_i, a_i) = n+1$ for all $1 \le i \le n$. If we set the upper quota of b_i to ∞, then $n-1$ residents $e_i, d_{i,1}, d_{i,2}, \ldots, d_{i,n-2}$ are assigned to b_i, so $g(b_i, b_i) = n-1$. Thus, Algorithm 3 constructs $S = \{b_1, \cdots, b_n\}$ at line 4 and the solution has $n^2 - n = |R| - \sqrt{|R|}$ blocking residents. However, consider the following matching: First, apply the Gale-Shapley algorithm for D and $B \cup X$. Then, assign the residents in $C \cup E$ to the empty hospitals in X arbitrarily. Then, nobody in D can be a blocking resident. Hence the cost is at most $2n = 2\sqrt{|R|}$. Therefore, the approximation ratio is at least $(|R| - \sqrt{|R|})/(2\sqrt{|R|}) = \Omega(\sqrt{|R|})$.

4.3.3 Inapproximability

For the hardness of Min-BR HRLQ, we have only NP-hardness, but we can give a strong evidence for its inapproximability. The *Densest k-Subgraph Problem* (DkS) is the problem of finding, given a graph G and a positive integer k, an induced subgraph of G with k vertices that contains as many edges as possible. This problem is NP-hard because it is a generalization of Max CLIQUE. Its approximability has been studied intensively but there still remains a large gap between approximability and inapproximability: The best known approximation ratio is $|V|^{1/4+\epsilon}$ [BCC$^+$10], while there is no PTAS under reasonable assumptions [Fei02, Kho06]. The following

Theorem 7 shows that approximating Min-BR HRLQ within a constant ratio implies the same for DkS.

Theorem 7. *If Min-BR 1ML-HRLQ has a polynomial-time c-approximation algorithm, then DkS has a polynomial-time $(1+\epsilon)c^4$-approximation algorithm for any positive constant ϵ.*

Proof. The proof uses another problem called *Minimum Coverage Problem* (MinC) [Vin07]. In MinC, we are given a family $\mathcal{P}$ of subsets of a base set $\mathcal{U}$ and a positive integer t, and asked to select t sets from $\mathcal{P}$ so that their union is minimized. Theorem 7 can be easily proved by combining the following two lemmas, whose proofs will be given shortly:

Lemma 17. *If MinC admits a polynomial-time c-approximation algorithm, then DkS admits a polynomial-time $(1+\epsilon)c^4$-approximation algorithm for any positive constant ϵ.*

Lemma 18. *If Min-BR 1ML-HRLQ admits a polynomial-time d-approximation algorithm, then MinC admits a polynomial-time $(1+\epsilon)d$-approximation algorithm for any positive constant ϵ.*

Suppose that Min-BR 1ML-HRLQ admits a polynomial-time c-approximation algorithm. Given an arbitrary positive constant ϵ, we choose ϵ' such that $\epsilon' \leq (1+\epsilon)^{\frac{1}{5}} - 1$ in Lemmas 17 and 18. By Lemma 18, MinC admits a polynomial-time $(1+\epsilon')c$-approximation algorithm and then by Lemma 17, DkS admits a polynomial-time $(1+\epsilon')^5c^4$-approximation algorithm. By the choice of ϵ', we have $(1+\epsilon')^5c^4 \leq (1+\epsilon)c^4$, and hence the proof of Theorem 7 is completed. □

Proof of Lemma 17. We will construct a polynomial-time $(1+\epsilon)c^4$-approximation algorithm for DkS using a c-approximation algorithm A for MinC. Suppose that we are given a graph $G = (V, E)$ and an integer k as an instance I of DkS. We regard each vertex in V as an element and each edge in E as a set of size two containing its two endpoints, and consider it as an instance of MinC. Recall that in MinC, we are given a positive integer t which specifies the number of sets we must select. We repeatedly apply algorithm A to this instance by increasing the target value of t one by one from one, until A outputs a solution of cost $c(k+1)$ or more for the first time. Let $\tilde{t}$ be the value of t at this point and $\tilde{s}$ be the value output by A. (If A never outputs such a solution even when $t = |E|$, it means that $|V| < c(k+1)$ in the given graph. This is more desirable case for us, as shown below.) Then, $\tilde{s} \geq c(k+1)$ by the above condition, and the optimal value of MinC when the target value is $\tilde{t}$ is at least $k+1$ since A is a c-approximation algorithm. This means that there is no subset of k vertices in G containing $\tilde{t}$ edges; in other words, the optimal value of the DkS instance I is less than $\tilde{t}$.

Note that when the target values in MinC differ by one, the two corresponding optimal values differ by at most two because adding one edge increases the number of vertices by at most two. Therefore, $\tilde{s} \leq c^2(k+1)+c$ since otherwise, $\tilde{s} > c^2(k+1)+c$ and the optimal value of MinC when the target value is $\tilde{t}$ is more than $c(k+1)+1$, namely at least $c(k+1)+2$, because A is a c-approximation algorithm. Then, when the target value is $\tilde{t}-1$, the optimal value of MinC is at least $c(k+1)$ by the above observation, and hence A must have already output a solution of value at least $c(k+1)$, a contradiction.

We now have a subgraph G' of G with $\tilde{s}$ vertices and at least $\tilde{t}$ edges. We then solve DkS approximately for G' (with the same k) using the greedy algorithm given in [AITT00]. We can find a subgraph of G' with k vertices and at least $\frac{k(k-1)}{\tilde{s}(\tilde{s}-1)}\tilde{t}$ edges, which is a $\frac{\tilde{s}(\tilde{s}-1)}{k(k-1)}$-approximate solution of the original problem I (recall that the optimal value of I is less than $\tilde{t}$). Since $\tilde{s} \leq c^2(k+1)+c$ as proved above,

$$\frac{\tilde{s}(\tilde{s}-1)}{k(k-1)} \leq c^4 + \frac{(3k+1)c^4 + 2(k+1)c^3 - kc^2 - c}{k(k-1)}.$$

Note that for any fixed constants c and ϵ, we can find a constant k_0 such that $\frac{(3k+1)c^4+2(k+1)c^3-kc^2-c}{k(k-1)} \leq \epsilon c^4$ for all $k \geq k_0$. Also, note that DkS when k is a constant is solvable in polynomial time. Thus, given a DkS instance, solving optimally when $k < k_0$, and using the above reduction otherwise, is a desirable $(1+\epsilon)c^4$-approximation algorithm.

If A does not output a solution when determining $\tilde{s}$, we know that $|V| < c(k+1)$ as discussed previously. In this case we simply apply the above greedy algorithm to G itself instead of G'. The optimal cost is at most $|E|$ and the algorithm's cost is at least $\frac{k(k-1)}{|V|(|V|-1)}|E|$, so the approximation ratio is at most $\frac{|V|(|V|-1)}{k(k-1)}$. By a similar argument as above, we can show that this is bounded by $(1+\epsilon)c^2$ for any positive ϵ for large enough k. This completes the proof. □

Proof of Lemma 18. We give a polynomial-time reduction from MinC to Min-BR 1ML-HRLQ. Suppose that a given instance I_0 of MinC consists of the base set

$\mathcal{U}= \{u_1, u_2, \ldots, u_n\}$, a collection $\mathcal{P}= \{P_1, P_2, \ldots, P_m\}$ of subsets of $\mathcal{U}$, and a positive integer t (the number of subsets to be selected). We construct an instance I of Min-BR 1ML-HRLQ.

Let $R = C \cup U$ be the set of residents and $H = P \cup \{x\}$ be the set of hospitals, where each set is defined as follows: $C = \{c_i \mid 1 \le i \le m-t\}$, $U = \{u_i^j \mid 1 \le i \le n, 1 \le j \le B\}$, and $P = \{p_i \mid 1 \le i \le m\}$. Note that $|R| = nB + m - t$. Here, B is a positive integer determined later. Preference lists and quotas are defined in Fig. 4.4. For each i ($1 \le i \le n$), residents u_i^j ($1 \le j \le B$) correspond to the element u_i of the base set $\mathcal{U}$ of MinC. Each $[0,1]$-hospital p_i corresponds to the subset P_i of MinC instance I_0. For each resident u_i^j, the set $P(i)$ contains the hospital p_k if and only if the element u_i is contained in the set P_k in I_0. For a set S, "$[S]$" denotes an arbitrarily ordered list of the members in S. Note that all the preference lists of hospitals are identical. It is easy to see that the reduction can be performed in polynomial time.

$$
\begin{array}{lllll}
c_i & : [P] & x & & (1 \le i \le m-t) \\
u_i^j & : [P(i)] & x & [P \setminus P(i)] & (1 \le i \le n, 1 \le j \le B) \\
 & & & & \\
p_i[0,1] & : [C] & [U] & & (1 \le i \le m) \\
x[nB, nB] & : [C] & [U] & &
\end{array}
$$

Fig. 4.4 Preference lists of residents and hospitals

Let $opt(I_0)$ and $opt(I)$ be the optimal costs of I_0 and I, respectively. In the following, we show that (i) $opt(I) \le B \cdot opt(I_0) + (m-t)$, and (ii) from a solution of I of cost a, we can construct a solution of I_0 of cost at most $(a+m-t)/B$ in polynomial time.

Hence, if there is a polynomial-time d-approximation algorithm for Min-BR 1ML-HRLQ, namely, if $\frac{a}{opt(I)} \le d$, then we can obtain

$$
\begin{aligned}
\frac{(a+m-t)/B}{opt(I_0)} &\le d + \frac{(d+1)(m-t)}{B \cdot opt(I_0)} \\
&\le d + \frac{2md}{B \cdot opt(I_0)} \\
&\le (1 + \frac{2m}{B})d.
\end{aligned}
$$

Now, if we take $B \ge \frac{2}{\epsilon}m$, then $(1+\frac{2m}{B})d \le (1+\epsilon)d$, as desired.

We first prove (i). Let $\mathcal{P}^*$ be an optimal solution (a subset of size t) for I_0. We will construct a solution M of I as follows: Let $M(u_i^j) = x$ for all i and j. Assign residents in C to hospitals corresponding to subsets in $\mathcal{P} \setminus \mathcal{P}^*$ in an arbitrary way. For each $P_j \in \mathcal{P}^*$, let the hospital p_j be empty. Consider a resident u_i^j and consider a subset P_k of I_0 that contains the element u_i. Note that u_i^j prefers the hospital p_k to x. If $P_k \notin \mathcal{P}^*$, then p_k receives a resident better than u_i^j in M and hence (u_i^j, p_k) is not a blocking pair. If $P_k \in \mathcal{P}^*$, then p_k is empty in M and hence (u_i^j, p_k) is a blocking pair. Hence, $\mathcal{P}^*$ does not include any P_k that contains u_i (in other words, the element u_i does not contribute to the cost of $\mathcal{P}^*$) if and only if u_i^j is not a blocking resident. There are $(m-t)+nB$ residents and among them $B(n-opt(I_0))$ are non-blocking as observed. Thus the number of blocking residents for M is at most $(m-t)+nB-B(n-opt(I_0)) = B \cdot opt(I_0) + (m-t)$, which completes the proof of (i).

We then prove (ii). Consider a feasible matching M of cost a. We may assume without loss of generality that all the residents are assigned in M because if not, we can assign unassigned residents to under-subscribed hospitals arbitrarily without increasing the cost. Let $C_x = \{c \mid c \in C, M(c) = x\}$ and $U_p = \{u \mid u \in U, M(u) \in P\}$. Then, $|C_x| = |U_p|$ since $|M(x)| = |C_x| + (|U| - |U_p|)$ and $|M(x)| = nB = |U|$ by the lower quota of x.

Let M' be a matching obtained by M by exchanging assigned hospitals between C_x and U_p arbitrarily. The following (1)–(3) are easy to verify: (1) Any resident in $C \setminus C_x$ does not change its assigned hospital, and no hospital in P becomes worse off. Therefore, no new blocking resident arises from $C \setminus C_x$. (2) Any resident r in C_x is a blocking resident in M because r is assigned to x and there is a hospital in P that receives a resident from U_p. Therefore, no new blocking resident arises from C_x. (3) For the same reason as (1), no new blocking resident arises from $U \setminus U_p$. Hence, only residents in U_p can newly become blocking residents. Since $|U_p| = |C_x| \leq |C| = m-t$, the number of blocking residents for M' is at most $a + (m-t)$.

Construct a solution $\mathcal{P}'$ of I_0 from M' such that $\mathcal{P}' = \{P_i \mid \text{ hospital } p_i \text{ is empty in } M'\}$. Clearly, $|\mathcal{P}'| = t$. We show that the cost of $\mathcal{P}'$ is at most $(a+m-t)/B$. Partition U into B subsets $U_j = \{u_i^j \mid 1 \leq i \leq n\}$ $(1 \leq j \leq B)$. Then there is an integer j such that U_j contains at most $(a+m-t)/B$ blocking residents. If u_i^j is non-blocking, all the hospitals superior to x for u_i^j are assigned in M', and hence by the construction of $\mathcal{P}'$, no subset containing u_i is selected in $\mathcal{P}'$, i.e., the element u_i

does not contribute to the cost of $\mathcal{P}'$. Hence, only elements u_i whose corresponding residents u_i^j are blocking can contribute to the cost of $\mathcal{P}'$. Therefore, the cost of $\mathcal{P}'$ is at most $(a+m-t)/B$. □

4.4 Concluding Remarks

In this chapter, we defined HRLQ. Then, we showed that Min-BP HRLQ is hard to approximate within the ratio of $(|H|+|R|)^{1-\epsilon}$ for any positive constant ϵ where H and R are the sets of hospitals and residents, respectively. We then gave an exponential-time exact algorithm whose running time is $O((|H||R|)^{t+1})$, where t is the number of blocking pairs in an optimal solution. We also considered another measure for optimization criteria, i.e., the number of residents who are involved in blocking pairs. We showed that Min-BR HRLQ is still NP-hard but has a polynomial-time $\sqrt{|R|}$-approximation algorithm.

A future research is to obtain lower bounds on the approximation factor for Min-BR HRLQ (we even do not know its APX-hardness at this moment). Since this problem is harder than the Densest k-Subgraph Problem, which is a problem of finding an induced subgraph with k vertices that contains as many edges as possible, it should be a reasonable challenge.

As for Min-BP HRLQ, it is interesting to consider a decision variant, namely, the problem of asking whether an optimal solution contains at most k blocking pairs for a given integer k. In Theorem 2, we have shown that the problem of determining whether the optimal cost is at most n^δ or at least $n^{1-\delta}$ is NP-hard for any constant $\delta(>0)$, where $n=|H|+|R|$. This implies that the decision problem is NP-hard if $k=O(n^\delta)$ for any δ. On the other hand, Theorem 4 implies that the problem is solvable in polynomial time when k is a constant. It is interesting to consider the complexity of the problem when k is between them, e.g., $k=$polylog(n).

Another direction was to develop an FPT algorithm (parameterized by the optimal cost t) for Min-BP HRLQ, improving Theorem 4. Recently, this was solved negatively by Mnich and Schlotter [MS20]. As a special case of a theorem shown in [MS20], it is proved that Min-BP HRLQ is not fixed-parameter tractable parameterized by t.

Finally, we remark on the possibility of generalization of instances: In this chapter, we guarantee existence of feasible matchings by the CL-restriction (Section 4.1). However, even if we allow arbitrarily incomplete lists (and even ties), it is decidable in polynomial time if the given instance admits a feasible matching [Gab83]. Thus, it might be interesting to seek approximate solutions for instances without the CL-restriction. Unfortunately, however, we can easily imply its $|R|^{1-\epsilon}$-approximation hardness in the following way.

Consider the problem of finding a maximum cardinality matching with the fewest blocking pairs, given a stable marriage instance with incomplete preference lists (call it *Min-BP SMI* for short). Its approximation hardness of $n^{1-\epsilon}$ for any positive constant ϵ is already known [BMM10], where n is the number of men in an input. The reduction given in Chapter 3, whose idea was taken from [BMM10], constructs an instance of Min-BP SMI having a perfect matching and creates a large gap on the number of blocking pairs between "yes" instances and "no" instances. We can verify that this gap holds also for the number of men involved in blocking pairs. If we regard instances produced by this reduction as ones of Min-BR HRLQ, by considering men and women as residents and hospitals, respectively, and setting the quotas to $[1, 1]$ for all the hospitals, then we can show $|R|^{1-\epsilon}$-approximation hardness of 0-1 Min-BR HRLQ.

Chapter 5

Algorithms for Noncrossing Matchings

In this chapter, we give algorithms and a hardness result for problems of finding a noncrossing matching.

Ruangwises and Itoh [RI19] incorporated the notion of noncrossing matchings [Ata85, CLW15, KT86, MOP93, WW85] to SMI. In their model, there are two parallel lines where n men are aligned on one line and n women are aligned on the other line. A matching is *noncrossing* if no two edges of it cross each other. A *stable noncrossing matching* is a matching which is simultaneously stable and noncrossing. They defined two notions of stability: In a *strongly stable noncrossing matching* (*SSNM*), the definition of a blocking pair is the same as that of the standard stable marriage problem. Thus the set of SSNMs is exactly the intersection of the set of stable matchings and that of noncrossing matchings. In a *weakly stable noncrossing matching* (*WSNM*), a blocking pair has an additional condition that it must not cross matching edges. Ruangwises and Itoh [RI19] proved that a WSNM exists for any instance, and presented an $O(n^2)$-time algorithm for the problem of finding a WSNM (denoted FIND_WSNM). They also showed that the same results hold for the weak stability when ties are present in preference lists. Furthermore, they demonstrated that an SSNM does not always exist, and that there can be WSNMs of different sizes. Concerning these observations, they posed open questions on the complexities of the problems of determining the existence of an SSNM (denoted EXIST_SSNM) and finding a WSNM of maximum cardinality (denoted MAX_WSNM).

Table 5.1 summarizes previous and our results, where our results are described in bold. We first show that both the above mentioned open problems are solvable in polynomial time. Specifically, EXIST_SSNM is solved in $O(n^2)$-time by exploiting the well-known Rural Hospitals theorem (Proposition 1) and MAX_WSNM is solved in $O(n^4)$-time by an algorithm based on dynamic programming (Theorem 10).

We then consider SMTI where preference lists may include ties. SMTI has three stability notions, *super-*, *strong*, and *weak* stability [Irv94]. We show that our algorithm for solving MAX_WSNM is applicable to all of the three stability notions with slight modifications (Corollary 4). We also show that our algorithm for solving EXIST_SSNM can be applied to super- and strong stabilities without any modification (Corollaries 2 and 3). In contrast, we show that EXIST_SSNM is NP-complete for the weak stability (Theorem 8).

This NP-completeness holds even for a restricted case where the length of each person's preference list is at most two and ties appear in only men's preference lists. To complement this intractability, we show that if each man's preference list contains at most one woman (but women's preference lists may be of unbounded length), the problem is solvable in $O(n)$-time (Theorem 9). If we parameterize this problem by two positive integers p and q that bound the lengths of preference lists of men and women, respectively, Theorem 8 shows that the problem is NP-complete even if $p \leq 2$ and $q \leq 2$, while Theorem 9 shows that the problem is solvable in polynomial time if $p = 1$ or $q = 1$ (by symmetry of men and women). Thus the computational complexity of the problem is completely solved in terms of the length of preference lists. We remark that this is a rare case since many NP-hard variants of the stable marriage problem can be solved in polynomial time if the length of preference lists of one side is bounded by two [IMO09, BMM10, BMM12, MO19].

5.1 Preliminaries

A pair in a matching can be seen as an edge on the plane, so we may use "pair" and "edge" interchangeably. Two edges (m_i, w_j) and (m_x, w_y) are said to *cross* each other if they share an interior point, or formally, if $(x - i)(y - j) < 0$ holds. A matching is *noncrossing* if it contains no pair of crossing edges.

For a matching M, a *noncrossing blocking pair* for M is a blocking pair for M that does not cross any edge of M. A matching M is a *weakly stable noncrossing matching* (*WSNM*) if M is noncrossing and does not admit any noncrossing blocking pair. A matching M is a *strongly stable noncrossing matching* (*SSNM*) if M is noncrossing and does not admit any blocking pair. Note that an SSNM is always a WSNM by definition but the converse is not true.

Table 5.1 Previous and our results (our results in bold)

		EXIST_SSNM	FIND_WSNM	MAX_WSNM
SMI		$\boldsymbol{O(n^2)}$ **[Proposition 1]**	$O(n^2)$ [RI19]	$\boldsymbol{O(n^4)}$ **[Theorem 10]**
SMTI	super-	$\boldsymbol{O(n^2)}$ **[Corollary 2]**		$\boldsymbol{O(n^4)}$ **[Corollary 4]**
	strong	$\boldsymbol{O(n^3)}$ **[Corollary 3]**		$\boldsymbol{O(n^4)}$ **[Corollary 4]**
	weak	**NPC***1 **[Theorem 8]** $\boldsymbol{O(n)}$*2 **[Theorem 9]**	$O(n^2)$ [RI19]	$\boldsymbol{O(n^4)}$ **[Corollary 4]**

*1 even if each person's preference list contains at most two persons and ties appear in only men's preference lists.
*2 if each man's preference list contains at most one woman.

We then extend the above definitions to the case where preference lists may contain ties. When ties are present, there are three possible definitions of blocking pairs and three stability notions as described in Section 2.1.3. With these definitions of blocking pairs, the terms "noncrossing blocking pair", "WSNM", and "SSNM" for each stability notion can be defined analogously. In the SMTI case, we extend the names of stable noncrossing matchings using the type of stability as a prefix. For example, a WSNM in the super-stability is denoted *super-WSNM*.

Note that, in this chapter, the terms "weak" and "strong" are used in two different meanings. This might be confusing but we decided not to change these terms, respecting previous literature.

For implementation of our algorithms, we use ranking arrays described in Section 1.2.3 of [GI89]. Although in [GI89] ranking arrays are defined for complete preference lists without ties, they can easily be modified for incomplete lists and/or with ties. Then, by the aid of ranking arrays, we can determine, given persons p, q_1, and q_2, whether $q_1 \succ_p q_2$ or $q_2 \succ_p q_1$ or $q_1 =_p q_2$ in constant time. Also we can determine, given m and w, if (m, w) is an acceptable pair or not in constant time.

5.2 Strongly Stable Noncrossing Matchings

5.2.1 Algorithm for SMI

In SMI, an easy observation shows that existence of an SSNM can be determined in $O(n^2)$ time:

Proposition 1. *There exists an $O(n^2)$-time algorithm to find an SSNM or to report that none exists, given an SMI-instance.*

Proof. Note that an SSNM is a stable matching in the original sense. In SMI, there always exists at least one stable matching [GI89], and due to the Rural Hospitals theorem [GS85, Rot84, Rot86], the set of matched agents is the same in any stable matching. These agents can be determined in $O(n^2)$ time by using the Gale-Shapley algorithm [GS62]. There is only one way of matching them in a noncrossing manner. Hence the matching constructed in this way is the unique candidate for an SSNM. All we have to do is to check if it is stable, which can be done in $O(n^2)$ time. □

5.2.2 Algorithms and Hardness Result for SMTI

In the presence of ties, super-stable and strongly stable matchings do not always exist. However, there is an $O(n^2)$-time ($O(n^3)$-time, respectively) algorithm that finds a super-stable (strongly stable, respectively) matching or reports that none exists [Irv94, KMMP07]. Also, the Rural Hospitals theorem takes over to the super-stability [IMS00] and strong stability [IMS03]. Therefore, the same algorithm as in Section 5.2.1 applies for these cases, implying the following corollaries:

Corollary 2. *There exists an $O(n^2)$-time algorithm to find a super-SSNM or to report that none exists, given an SMTI-instance.*

Corollary 3. *There exists an $O(n^3)$-time algorithm to find a strong-SSNM or to report that none exists, given an SMTI-instance.*

In contrast, the problem becomes NP-complete for the weak stability even for a highly restricted case:

Theorem 8. *The problem of determining if a weak-SSNM exists, given an SMTI-instance, is NP-complete, even if each person's preference list contains at most two persons and ties appear in only men's preference lists.*

Proof. Membership in NP is obvious. We show NP-hardness by a reduction from 3SAT [Coo71]. An instance of 3SAT consists of a set of variables and a set of clauses. Each variable takes either true (1) or false (0). A *literal* is a variable or its negation. A *clause* is a disjunction of at most three literals. A clause is *satisfied* if at least one of its literals takes the value 1. A 0/1 assignment to variables that satisfies all the clauses is called a *satisfying assignment*. An instance f of 3SAT is *satisfiable* if it has at least one satisfying assignment. 3SAT asks if there exists a satisfying assignment. 3SAT is NP-complete even if each variable appears exactly four times, exactly twice positively and exactly twice negatively, and each clause contains exactly three literals [BKS03]. We use 3SAT instances restricted in this way.

Now we show the reduction. Let f be an instance of 3SAT having n variables x_i ($1 \leq i \leq n$) and m clauses C_j ($1 \leq j \leq m$). For each variable x_i, we construct a *variable gadget*. It consists of six men $p_{i,1}$, $p_{i,2}$, $p_{i,3}$, $p_{i,4}$, $a_{i,1}$, and $a_{i,2}$, and four women $q_{i,1}$, $q_{i,2}$, $q_{i,3}$, and $q_{i,4}$. A variable gadget corresponding to x_i is called an *x_i-gadget*. For each clause C_j, we construct a *clause gadget*. It consists of seven men $y_{j,k}$ ($1 \leq k \leq 7$) and nine women $v_{j,k}$ ($1 \leq k \leq 6$) and $z_{j,k}$ ($1 \leq k \leq 3$). A clause gadget corresponding to C_j is called a *C_j-gadget*. Additionally, we create a man s and a woman t, who constitute a gadget called the *separator*.

Thus, there are $6n + 7m + 1$ men and $4n + 9m + 1$ women in the created SMTI-instance, denoted $I(f)$. Finally, we add dummy persons who have empty preference lists to make the numbers of men and women equal. They do not play any role in the following arguments, so we omit them.

Suppose that x_i's kth positive occurrence ($k = 1, 2$) is in the $d_{i,k}$th clause $C_{d_{i,k}}$ as the $e_{i,k}$th literal ($1 \leq e_{i,k} \leq 3$). Similarly, suppose that x_i's kth negative occurrence ($k = 1, 2$) is in the $g_{i,k}$th clause $C_{g_{i,k}}$ as the $h_{i,k}$th literal ($1 \leq h_{i,k} \leq 3$). The preference lists of ten persons in the x_i-gadget are constructed as shown in Fig. 5.1. Here, each preference list is described as a sequence from left to right according to preference, i.e., the leftmost person is the most preferred and the rightmost person is the least preferred. Tied persons (i.e., persons with the equal preference) are included in parentheses. Men are aligned in the order of $p_{i,1}$, $p_{i,3}$, $a_{i,1}$, $a_{i,2}$, $p_{i,2}$, and $p_{i,4}$ from top to bottom, and women are aligned in the order of $q_{i,1}$, $q_{i,3}$, $q_{i,2}$, and $q_{i,4}$. (See Fig. 5.2. Edges depicted in the figure are those within the variable gadget.)

It might be helpful to explain here intuition behind a variable gadget. People there are partitioned into two groups, $\{p_{i,1}, a_{i,1}, p_{i,2}, q_{i,1}, q_{i,2}\}$ and $\{p_{i,3}, a_{i,2}, p_{i,4}, q_{i,3}, q_{i,4}\}$.

$p_{i,1}$:	$q_{i,1}$	$z_{g_{i,1},h_{i,1}}$		$q_{i,1}$:	$a_{i,1}$	$p_{i,1}$
$a_{i,1}$:	$(q_{i,1}$	$q_{i,2})$		$q_{i,2}$:	$a_{i,1}$	$p_{i,2}$
$p_{i,2}$:	$q_{i,2}$	$z_{d_{i,1},e_{i,1}}$				
$p_{i,3}$:	$q_{i,3}$	$z_{g_{i,2},h_{i,2}}$		$q_{i,3}$:	$a_{i,2}$	$p_{i,3}$
$a_{i,2}$:	$(q_{i,3}$	$q_{i,4})$		$q_{i,4}$:	$a_{i,2}$	$p_{i,4}$
$p_{i,4}$:	$q_{i,4}$	$z_{d_{i,2},e_{i,2}}$				

Fig. 5.1 Preference lists of persons in x_i-gadget

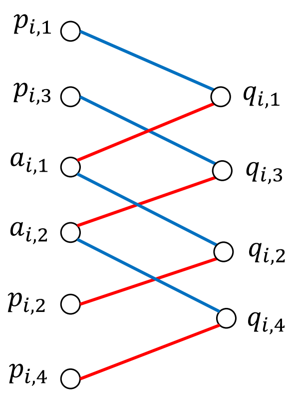

Fig. 5.2 Alignment of agents in a variable gadget. This gadget admits two non-crossing stable matchings highlighted in blue and red, associated with assignment $x_i = 0$ and $x_i = 1$, respectively.

The first group corresponds to the first positive occurrence and the first negative occurrence of x_i. It has two stable matchings $\{(p_{i,1}, q_{i,1}), (a_{i,1}, q_{i,2})\}$ (blue in Fig. 5.2) and $\{(a_{i,1}, q_{i,1}), (p_{i,2}, q_{i,2})\}$ (red). We associate the former with the assignment $x_i = 0$ and the latter with the assignment $x_i = 1$. The second group corresponds to the second positive occurrence and the second negative occurrence of x_i. It has two stable matchings $\{(p_{i,3}, q_{i,3}), (a_{i,2}, q_{i,4})\}$ (blue) and $\{(a_{i,2}, q_{i,3}), (p_{i,4}, q_{i,4})\}$ (red). We associate the former with $x_i = 0$ and the latter with $x_i = 1$. Entanglement of two groups as in Fig. 5.2 plays a role of ensuring consistency of assignments between the first and the second group. Depending on the choice of the matching in the first group, edges with the same color must be chosen from the second group to avoid edge-crossing.

Let us continue the reduction. We then construct preference lists of clause gadgets. Consider a clause C_j, and suppose that its kth literal is of a variable x_{j_k}. Define $\ell_{j,k}$ as

$$\ell_{j,k} = \begin{cases} 1 \text{ if this is the 1st negative occurrence of } x_{j_k} \\ 2 \text{ if this is the 1st positive occurrence of } x_{j_k} \\ 3 \text{ if this is the 2nd negative occurrence of } x_{j_k} \\ 4 \text{ if this is the 2nd positive occurrence of } x_{j_k}. \end{cases}$$

The preference lists of persons in the C_j-gadget are as shown in Fig. 5.3. The alignment order of persons in each clause gadget is the same as in Fig. 5.3. Since a clause gadget is complicated, we show a structure in the leftmost figure of Fig. 5.4 (three matchings $N_{j,1}$, $N_{j,2}$, and $N_{j,3}$ will be used later).

$y_{j,1}$:	$(v_{j,1} \quad v_{j,3})$	$v_{j,1}$:	$y_{j,1}$	
$y_{j,2}$:	$(v_{j,2} \quad z_{j,1})$	$v_{j,2}$:	$y_{j,2}$	
$y_{j,3}$:	$(v_{j,3} \quad v_{j,4})$	$v_{j,3}$:	$y_{j,1}$	$y_{j,3}$
$y_{j,4}$:	$(z_{j,2} \quad v_{j,5})$	$z_{j,1}$:	$y_{j,2}$	$p_{j_1,\ell_{j,1}}$
$y_{j,5}$:	$(v_{j,4} \quad v_{j,6})$	$z_{j,2}$:	$y_{j,4}$	$p_{j_2,\ell_{j,2}}$
$y_{j,6}$:	$(v_{j,5} \quad z_{j,3})$	$v_{j,4}$:	$y_{j,5}$	$y_{j,3}$
$y_{j,7}$:	$v_{j,6}$	$v_{j,5}$:	$y_{j,6}$	$y_{j,4}$
		$v_{j,6}$:	$y_{j,5}$	$y_{j,7}$
		$z_{j,3}$:	$y_{j,6}$	$p_{j_3,\ell_{j,3}}$

Fig. 5.3 Preference lists of persons in C_j-gadget

Finally, each of the man and the woman in the separator includes only the other in the list (Fig. 5.5). They are guaranteed to be matched together in any stable matching.

Alignment of the whole instance is depicted in Fig. 5.6. Variable gadgets are placed top, then followed by the separator, clause gadgets come bottom. The separator plays a role of prohibiting a person of a variable gadget to be matched with a person of a clause gadget; if they are matched, then the corresponding edge crosses the separator.

Now the reduction is completed. It is not hard to see that the reduction can be performed in polynomial time and the conditions on the preference lists stated in the theorem are satisfied.

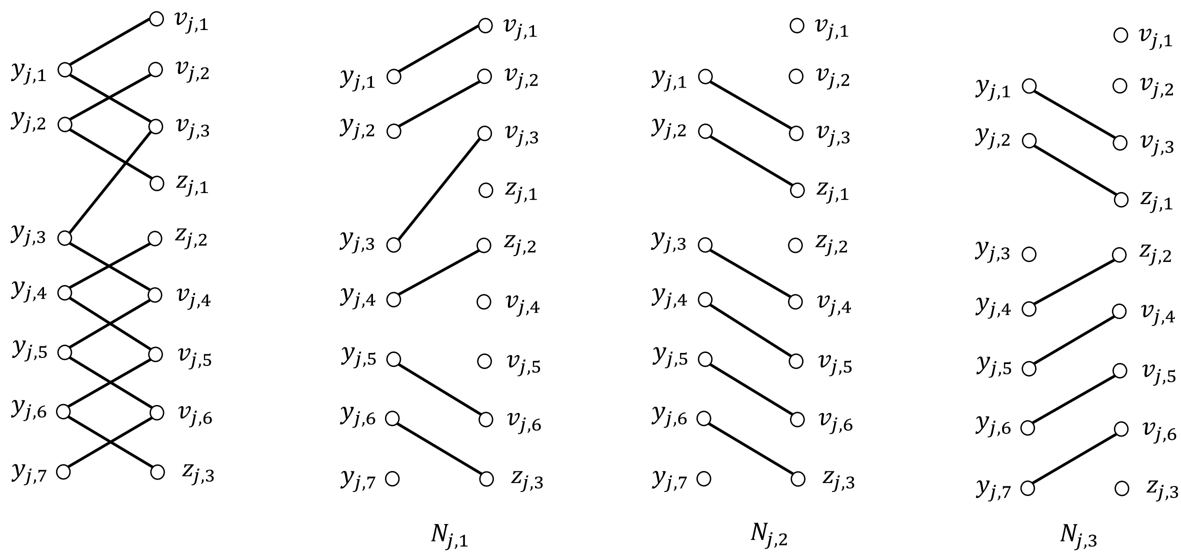

Fig. 5.4 Acceptability graph of a clause gadget C_j and its matchings $N_{j,1}$, $N_{j,2}$, and $N_{j,3}$

s: t t: s

Fig. 5.5 Preference lists of the man and the woman in the separator

We then show the correctness. First, suppose that f is satisfiable and let A be a satisfying assignment. We construct a weak-SSNM M of $I(f)$ from A. For an x_i-gadget, define two matchings

- $M_{i,0} = \{(p_{i,1}, q_{i,1}), (a_{i,1}, q_{i,2}), (p_{i,3}, q_{i,3}), (a_{i,2}, q_{i,4})\}$ (blue in Fig. 5.2) and
- $M_{i,1} = \{(a_{i,1}, q_{i,1}), (p_{i,2}, q_{i,2}), (a_{i,2}, q_{i,3}), (p_{i,4}, q_{i,4})\}$ (red in Fig. 5.2).

If $x_i = 0$ under A, then add $M_{i,0}$ to M; otherwise, add $M_{i,1}$ to M. For a C_j-gadget, we define three matchings

- $N_{j,1} = \{(y_{j,1}, v_{j,1}), (y_{j,2}, v_{j,2}), (y_{j,3}, v_{j,3}), (y_{j,4}, z_{j,2}), (y_{j,5}, v_{j,6}), (y_{j,6}, z_{j,3})\}$,
- $N_{j,2} = \{(y_{j,1}, v_{j,3}), (y_{j,2}, z_{j,1}), (y_{j,3}, v_{j,4}), (y_{j,4}, v_{j,5}), (y_{j,5}, v_{j,6}), (y_{j,6}, z_{j,3})\}$, and
- $N_{j,3} = \{(y_{j,1}, v_{j,3}), (y_{j,2}, z_{j,1}), (y_{j,4}, z_{j,2}), (y_{j,5}, v_{j,4}), (y_{j,6}, v_{j,5}), (y_{j,7}, v_{j,6})\}$,

that are depicted in Fig. 5.4. Note that, for each $k \in \{1, 2, 3\}$, only $z_{j,k}$ (among $z_{j,1}$, $z_{j,2}$, and $z_{j,3}$) is single in $N_{j,k}$. If C_j is satisfied by the kth literal ($k \in \{1, 2, 3\}$), then add $N_{j,k}$ to M. (If C_j is satisfied by more than one literal, then choose one arbitrarily.) Finally add the pair (s, t) to M.

It is not hard to see that M is noncrossing. We show that it is weakly stable.

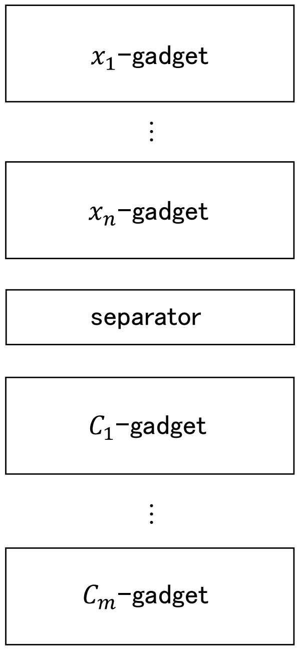

Fig. 5.6 Alignment of agents

Clearly, neither s nor t in the separator forms a blocking pair. Next, consider the x_i-gadget. In $M_{i,0}$, women $q_{i,2}$ and $q_{i,4}$ are matched with the first-choice man. The woman $q_{i,1}$ is matched with the second-choice man $p_{i,1}$ but her first-choice man $a_{i,1}$ is matched with a first-choice woman $q_{i,2}$. Similarly, $q_{i,3}$'s first-choice man $a_{i,2}$ is matched with a first-choice woman $q_{i,4}$. Men $p_{i,1}$, $a_{i,1}$, $p_{i,3}$, and $a_{i,2}$ are matched with a first-choice woman. Hence these persons cannot be a part of a blocking pair; only $p_{i,2}$ and $p_{i,4}$ may participate in a blocking pair. Similarly, we can argue that, in $M_{i,1}$, only $p_{i,1}$ and $p_{i,3}$ may participate in a blocking pair.

Consider a C_j-gadget. In $N_{j,1}$, all the men except for $y_{j,7}$ are matched with a first-choice woman. $y_{j,7}$'s unique choice $v_{j,6}$ is matched with the first-choice man $y_{j,5}$. Hence no man in this gadget can participate in a blocking pair, and so no blocking pair exists within this gadget. Since $z_{j,2}$ and $z_{j,3}$ are matched with their respective first-choice woman, only the possibility is that $z_{j,1}$ forms a blocking pair with $p_{j_1,\ell_{j,1}}$ of a variable gadget. The same observation applies for $N_{j,2}$ and $N_{j,3}$ and we can see that for each $k \in \{1,2,3\}$ only $z_{j,k}$ can participate in a blocking pair in $N_{j,k}$.

To summarize, if there exists a blocking pair, it must be of the form $(p_{i,\ell}, z_{j,k})$ for some i, ℓ, j, and k, and both $p_{i,\ell}$ and $z_{j,k}$ are single in M. Suppose that $\ell = 1$. The reason for $(p_{i,1}, z_{j,k})$ being an acceptable pair is that C_j's kth literal is $\neg x_i$, a negative occurrence of x_i. Since $p_{i,1}$ is single, $M_{i,1} \subset M$ and hence $x_i = 1$ under A. Since

$z_{j,k}$ is single, $N_{j,k} \subset M$ and hence C_j is satisfied by its kth literal $\neg x_i$, but this is a contradiction. The other cases $\ell = 2, 3, 4$ can be argued in the same manner, and we can conclude that M is stable.

Conversely, suppose that $I(f)$ admits a weak-SSNM M. We construct a satisfying assignment A of f. Before giving construction, we observe structural properties of M in two lemmas:

Lemma 19. *For each i ($1 \leq i \leq n$), either $M_{i,0} \subset M$ or $M_{i,1} \subset M$.*

Proof. Note that preference lists of the ten persons of the x_i-gadget include persons of the same x_i-gadget or some persons from clause gadgets. Hence, due to the separator, persons of the x_i-gadget can only be matched within this gadget to avoid edge-crossings.

Note that a stable matching is a maximal matching. With regard to $p_{i,1}$, $a_{i,1}$, $p_{i,2}$, $q_{i,1}$, and $q_{i,2}$, there are three maximal matchings $\{(p_{i,1}, q_{i,1}), (a_{i,1}, q_{i,2})\}$, $\{(a_{i,1}, q_{i,1}), (p_{i,2}, q_{i,2})\}$, and $\{(p_{i,1}, q_{i,1}), (p_{i,2}, q_{i,2})\}$, but the last one is blocked by $(a_{i,1}, q_{i,1})$ and $(a_{i,1}, q_{i,2})$. Hence either the first or the second one must be in M. With regard to $p_{i,3}$, $a_{i,2}$, $p_{i,4}$, $q_{i,3}$, and $q_{i,4}$, there are three maximal matchings $\{(p_{i,3}, q_{i,3}), (a_{i,2}, q_{i,4})\}$, $\{(a_{i,2}, q_{i,3}), (p_{i,4}, q_{i,4})\}$, and $\{(p_{i,3}, q_{i,3}), (p_{i,4}, q_{i,4})\}$, but the last one is blocked by $(a_{i,2}, q_{i,3})$ and $(a_{i,2}, q_{i,4})$. Hence either the first or the second one must be in M.

If we choose $\{(p_{i,1}, q_{i,1}), (a_{i,1}, q_{i,2})\}$, then we must choose $\{(p_{i,3}, q_{i,3}), (a_{i,2}, q_{i,4})\}$ to avoid edge-crossing, which constitute $M_{i,0}$. If we choose $\{(a_{i,1}, q_{i,1}), (p_{i,2}, q_{i,2})\}$, then we must choose $\{(a_{i,2}, q_{i,3}), (p_{i,4}, q_{i,4})\}$, which constitute $M_{i,1}$. Hence either $M_{i,0}$ or $M_{i,1}$ must be a part of M. □

Lemma 20. *For a C_j-gadget, at least one of $z_{j,1}$, $z_{j,2}$, and $z_{j,3}$ is unmatched in M.*

Proof. Note that preference lists of the persons of the C_j-gadget include persons of the same C_j-gadget or some persons from variable gadgets. To avoid edge-crossing, persons must be matched within the same C_j-gadget.

For contradiction, suppose that all $z_{j,1}$, $z_{j,2}$, and $z_{j,3}$ are matched in M. Then $(y_{j,2}, z_{j,1})$, $(y_{j,4}, z_{j,2})$, and $(y_{j,6}, z_{j,3})$ are in M (Fig. 5.7(1)). To avoid edge-crossing, $(y_{j,3}, v_{j,3})$, $(y_{j,3}, v_{j,4})$, and $(y_{j,7}, v_{j,6})$ must not be in M (Fig. 5.7(2)). The pair $(y_{j,5}, v_{j,4})$ must be in M as otherwise $(y_{j,3}, v_{j,4})$ is a blocking pair (Fig. 5.7(3)). For M to be a matching, $(y_{j,5}, v_{j,6})$ must not be in M (Fig. 5.7(4)). Then $(y_{j,7}, v_{j,6})$ is a blocking pair, a contradiction. □

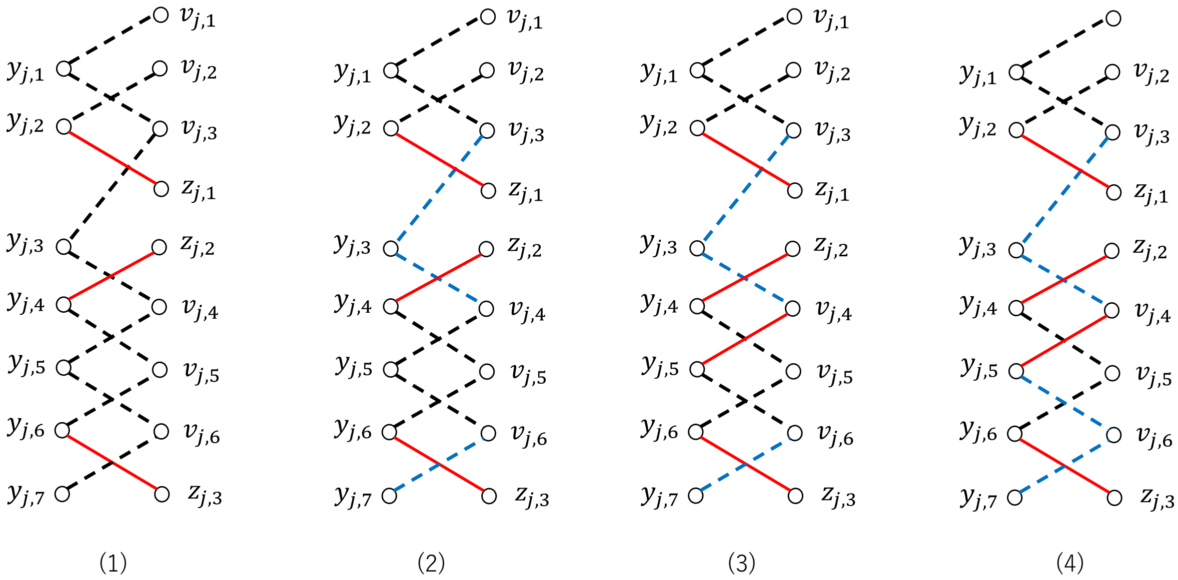

Fig. 5.7 Situation in the proof of Lemma 20. Red solid edges are those confirmed to be in M, blue dashed edges are those confirmed not to be in M, and black dashed edges are uncertain.

For each i, either $M_{i,0} \subset M$ or $M_{i,1} \subset M$ holds by Lemma 19. If $M_{i,0} \subset M$ holds then we set $x_i = 0$ in A, and if $M_{i,1} \subset M$ holds then we set $x_i = 1$ in A. We show that A satisfies f. Let C_j be an arbitrary clause. By Lemma 20, at least one of $z_{j,1}$, $z_{j,2}$, and $z_{j,3}$ is unmatched in M. If there are two or more unmatched women, then choose one arbitrarily and let this woman be $z_{j,k}$. We show that C_j is satisfied by its kth literal. Suppose not.

First suppose that the kth literal of C_j is the first positive occurrence of x_i. Then, by construction of preference lists, $(p_{i,2}, z_{j,k})$ is an acceptable pair. If $x_i = 0$ under A, then $M_{i,0} \subset M$ by construction of A, and hence $p_{i,2}$ is single in M. Thus $(p_{i,2}, z_{j,k})$ is a blocking pair, which contradicts stability of M. Hence $x_i = 1$ under A and C_j is satisfied by x_i. When the kth literal of C_j is the second positive occurrence of x_i, the same argument holds if we replace $p_{i,2}$ by $p_{i,4}$.

Next suppose that the kth literal of C_j is the first negative occurrence of x_i. Then, by construction of preference lists, $(p_{i,1}, z_{j,k})$ is an acceptable pair. If $x_i = 1$ under A, then $M_{i,1} \subset M$ by construction of A, and hence $p_{i,1}$ is single in M. Thus $(p_{i,1}, z_{j,k})$ is a blocking pair, which contradicts stability of M. Hence $x_i = 0$ under A and C_j is satisfied by $\neg x_i$. If the kth literal of C_j is the second negative occurrence of x_i, the same argument holds if we replace $p_{i,1}$ by $p_{i,3}$. Thus A is a satisfying assignment of f and the proof is completed. □

Next we give a positive result.

Theorem 9. *The problem of determining if a weak-SSNM exists, given an SMTI-instance, is solvable in $O(n)$-time if each man's preference list contains at most one woman.*

Proof. Let I be an input SMTI-instance. First, we construct the bipartite graph $G_I = (U_I, V_I, E_I)$, where U_I and V_I correspond to the sets of men and women in I, respectively, and $(m, w) \in E_I$ if and only if m is a first-choice of w. For a vertex $v \in V_I$, let $d(v)$ denote its degree in G_I. Since acceptability is mutual, if a woman w's preference list in I is nonempty, $d(w) \geq 1$ holds. Note that it can happen that $d(w) \geq 2$ because preference lists may contain ties. In the following lemma, we characterize (not necessarily noncrossing) stable matchings of I.

Lemma 21. *M is a stable matching of I if and only if $M \subseteq E_I$ and each woman $w \in V_I$ such that $d(w) \geq 1$ is matched in M.*

Proof. Suppose that M is stable. If $M \not\subseteq E_I$, there is an edge $(m, w) \in M \setminus E_I$. The fact $(m, w) \notin E_I$ means that m is not w's first-choice so there is an edge $(m', w) \in E_I$ such that $m' \succ_w m$. Since $(m, w) \in M$, m' is single in M. Therefore, (m', w) is a blocking pair for M, a contradiction. If there is a woman $w \in V_I$ such that $d(w) \geq 1$ but w is single in M, then any man m such that (m, w) is an acceptable pair is a blocking pair because m is also single in M, a contradiction.

Conversely, suppose that M satisfies the conditions of the right hand side. Then each woman who has a nonempty list is matched with a first-choice man, so there cannot be a blocking pair. □

By Lemma 21, our task is to select from E_I one edge per woman w such that $d(w) \geq 1$, in such a way that the resulting matching is noncrossing. We do this greedily. M is initially empty, and we add edges to M by processing vertices of V_I from top to bottom. At w_i's turn, if $d(w_i) \geq 1$, then choose the topmost edge that does not cross any edge in M, and add it to M. If there is no such edge, then we immediately conclude that I admits no weak-SSNM. If we can successfully process all the women, we output the final matching M, which is a weak-SSNM.

In the following, we formalize the above idea. A pseudo-code of the whole algorithm Weak-SSNM-1 is given in Algorithm 4.

We show the correctness. Suppose that Weak-SSNM-1 outputs a matching M.

Algorithm 4 WEAK-SSNM-1

Require: An SMTI-instance I.

Ensure: A weak-SSNM M or “No” if none exists.

```
1:  Construct the bipartite graph G_I = (U_I, V_I, E_I).
2:  Let M := ∅.
3:  for i = 1 to n do
4:    if d(w_i) ≥ 1 then
5:      Let j* (if any) be the smallest j such that (m_j, w_i) ∈ E_I and M ∪ {(m_j, w_i)}
        is a noncrossing matching.
6:      Let M := M ∪ {(m_{j*}, w_i)}.
7:      if no such j* exists then
8:        Output “No” and halt.
9:      end if
10:   end if
11: end for
12: Output M.
```

M is noncrossing by the condition of line 5, and M is stable because the construction of M follows the condition of Lemma 21.

Conversely, suppose that I admits a weak-SSNM M^*. We show that WEAK-SSNM-1 outputs a matching. Suppose not, and suppose that WEAK-SSNM-1 failed when processing woman (vertex) w_k. Let $\bar{M}$ be the matching constructed so far by WEAK-SSNM-1. Then for each i $(1 \leq i \leq k-1)$, w_i is single in M^* if and only if she is single in $\bar{M}$. Also, since $M^* \subseteq E_I$ by Lemma 21, we can show by a simple induction that for each i $(1 \leq i \leq k-1)$, if $M^*(w_i) = m_p$ and $\bar{M}(w_i) = m_q$, then $q \leq p$. Then, at line 5, we could have chosen $(M^*(w_k), w_k)$ to add to $\bar{M}$, a contradiction.

Finally, we consider time-complexity. Since the preference list of each man contains at most one woman, the graph G_I at line 1 can be constructed in $O(n)$-time and contains at most n edges. The **for**-loop can be executed in $O(n)$-time because each edge is scanned at most once in the loop; whether or not an edge crosses edges of M at line 5 can be done in constant time by keeping the maximum index of the matched men in M at any stage. □

5.3 Maximum Cardinality Weakly Stable Noncrossing Matchings

In this section, we present an algorithm to find a maximum cardinality WSNM. For an instance I, let $opt(I)$ denote the size of the maximum cardinality WSNM.

5.3.1 Algorithm for SMI

Let I' be a given instance with men $m_1, \ldots, m_n$ and women $w_1, \ldots, w_n$. To simplify the description of the algorithm, we translate I' to an instance I by adding a man m_0 and a woman w_0, each of whom includes only the other in the preference list, and similarly a man m_{n+1} and a woman w_{n+1}, each of whom includes only the other in the preference list. It is easy to see that, for a WSNM M' of I', $M = M' \cup \{(m_0, w_0), (m_{n+1}, w_{n+1})\}$ is a WSNM of I. Conversely, any WSNM M of I includes the pairs (m_0, w_0) and (m_{n+1}, w_{n+1}), and $M' = M \setminus \{(m_0, w_0), (m_{n+1}, w_{n+1})\}$ is a WSNM of I'. Thus we have that $opt(I) = opt(I') + 2$. Hence, without loss of generality, we assume that a given instance I has $n + 2$ men and $n + 2$ women, with m_0, w_0, m_{n+1}, and w_{n+1} having the above mentioned preference lists.

Let $M = \{(m_{i_1}, w_{j_1}), (m_{i_2}, w_{j_2}), \ldots, (m_{i_k}, w_{j_k})\}$ be a noncrossing matching of I such that $i_1 < i_2 \cdots < i_k$ and $j_1 < j_2 \cdots < j_k$. We call (m_{i_k}, w_{j_k}) the *maximum pair* of M. Suppose that (m_x, w_y) is the maximum pair of a noncrossing matching M. We call M a *semi-WSNM* if each of its noncrossing blocking pairs (m_i, w_j) (if any) satisfies $x \le i \le n+1$ and $y \le j \le n+1$. Intuitively, a semi-WSNM is a WSNM up to its maximum pair. Note that any semi-WSNM must contain (m_0, w_0), as otherwise it is a noncrossing blocking pair. For $0 \le i \le n + 1$ and $0 \le j \le n + 1$, we define $X(i, j)$ as the maximum size of a semi-WSNM of I whose maximum pair is (m_i, w_j); if I does not admit a semi-WSNM with the maximum pair (m_i, w_j), $X(i, j)$ is defined to be $-\infty$.

Lemma 22. *$opt(I) = X(n + 1, n + 1)$.*

Proof. Note that any WSNM of I includes (m_{n+1}, w_{n+1}), as otherwise it is a noncrossing blocking pair. Hence it is a semi-WSNM with the maximum pair (m_{n+1}, w_{n+1}). Conversely, any semi-WSNM with the maximum pair (m_{n+1}, w_{n+1}) does not include a noncrossing blocking pair and hence is also a WSNM. Therefore, the set of WSNMs is equivalent to the set of semi-WSNMs with the maximum pair (m_{n+1}, w_{n+1}). This completes the proof. □

To compute $X(n+1, n+1)$, we shortly define quantity $Y(i,j)$ $(0 \le i \le n+1, 0 \le j \le n+1)$ using recursive formulas, and show that $Y(i,j) = X(i,j)$ for all i and j. We then show that these recursive formulas allow us to compute $Y(i,j)$ in polynomial time using dynamic programming.

We say that two noncrossing edges (m_i, w_j) and (m_x, w_y) $(i < x, j < y)$ are *conflicting* if they admit a noncrossing blocking pair between them; precisely speaking, (m_i, w_j) and (m_x, w_y) are conflicting if the matching $\{(m_i, w_j), (m_x, w_y)\}$ admits a blocking pair (m_s, w_t) such that $i \le s \le x$ and $j \le t \le y$. Otherwise, they are *nonconflicting*. Intuitively, two conflicting edges cannot be consecutive elements of a semi-WSNM.

Now $Y(i,j)$ is defined in Equations (5.1) to (5.4). For convenience, we assume that $-\infty + 1 = -\infty$. In Equation (5.4), $Y(i', j')$ in $\max\{\}$ is taken among all (i', j') such that (i) $0 \le i' \le i-1$, (ii) $0 \le j' \le j-1$, (iii) $(m_{i'}, w_{j'})$ is an acceptable pair, and (iv) (m_i, w_j) and $(m_{i'}, w_{j'})$ are nonconflicting. If no such (i', j') exists, $\max\{Y(i', j')\}$ is defined as $-\infty$ and as a result $Y(i,j)$ is also computed as $-\infty$.

$$Y(0,0) = 1 \tag{5.1}$$

$$Y(0,j) = -\infty \quad (1 \le j \le n+1) \tag{5.2}$$

$$Y(i,0) = -\infty \quad (1 \le i \le n+1) \tag{5.3}$$

$$Y(i,j) = \begin{cases} 1 + \max\{Y(i', j')\} & \text{(if } (m_i, w_j) \text{ is an acceptable pair)} \\ -\infty & \text{(otherwise)} \end{cases}$$

$$(1 \le i \le n+1, 1 \le j \le n+1) \tag{5.4}$$

Lemma 23. *$Y(i,j) = X(i,j)$ for any i and j such that $0 \le i \le n+1$ and $0 \le j \le n+1$.*

Proof. We prove the claim by induction. We first show that $Y(0,0) = X(0,0)$. The matching $\{(m_0, w_0)\}$ is the unique semi-WSNM with the maximum pair (m_0, w_0), so $X(0,0) = 1$ by definition. Also, $Y(0,0) = 1$ by Equation (5.1). Hence we are done. We then show that $Y(0,j) = X(0,j)$ for $1 \le j \le n+1$. Since (m_0, w_j) is an unacceptable pair, there is no semi-WSNM with the maximum pair (m_0, w_j), so $X(0, j) = -\infty$ by definition. Also, $Y(0, j) = -\infty$ by Equation (5.2). We can show that $Y(i, 0) = X(i, 0)$ for $1 \leq i \leq n+1$ by a similar argument.

Next we show that $Y(i, j) = X(i, j)$ holds for $1 \leq i \leq n+1$ and $1 \leq j \leq n+1$. As an induction hypothesis, we assume that $Y(a, b) = X(a, b)$ holds for $0 \leq a \leq i-1$ and $0 \leq b \leq j-1$. First, observe that if $X(i, j) \neq -\infty$, then $X(i, j) \geq 2$. This is because two pairs (m_0, w_0) and (m_i, w_j) must be present in any semi-WSNM having the maximum pair (m_i, w_j).

We first consider the case that $X(i, j) \geq 2$. Let $X(i, j) = k$. Then, there is a semi-WSNM M with the maximum pair (m_i, w_j) such that $|M| = k$. Let $M' = M \setminus \{(m_i, w_j)\}$ and (m_x, w_y) be the maximum pair of M'. It is not hard to see that M' is a semi-WSNM with the maximum pair (m_x, w_y) and that $|M'| = k-1$. Therefore, $X(x, y) \geq k-1$ by the definition of X, and $Y(x, y) = X(x, y) \geq k-1$ by the induction hypothesis. Since M is a semi-WSNM, (m_i, w_j) and (m_x, w_y) are nonconflicting, so (x, y) satisfies the condition for (i', j') in Equation (5.4). Hence $Y(i, j) \geq 1 + Y(x, y) \geq k$. Suppose that $Y(i, j) \geq k+1$. By the definition of Y, this means that there is (i', j') that satisfies conditions (i)–(iv) for Equation (5.4), and $Y(i', j') \geq k$. By the induction hypothesis, $X(i', j') = Y(i', j') \geq k$. Then there is a semi-WSNM M' with the maximum pair $(m_{i'}, w_{j'})$ such that $|M'| \geq k$. Since M' is a semi-WSNM, and $(m_{i'}, w_{j'})$ and (m_i, w_j) are nonconflicting, $M = M' \cup \{(m_i, w_j)\}$ is a semi-WSNM with the maximum pair (m_i, w_j) such that $|M| = |M'| + 1 \geq k+1$. This contradicts the assumption that $X(i, j) = k$. Hence $Y(i, j) \leq k$ and therefore $Y(i, j) = k$ as desired.

Finally, consider the case that $X(i, j) = -\infty$. If (m_i, w_j) is unacceptable, then the latter case of Equation (5.4) is applied and $Y(i, j) = -\infty$. So assume that (m_i, w_j) is acceptable. Then the former case of Equation (5.4) is applied. It suffices to show that for any (i', j') that satisfies conditions (i)–(iv) (if any), $Y(i', j') = -\infty$ holds. Assume on the contrary that there is such (i', j') with $Y(i', j') = k$. Then $X(i', j') = k$ by the induction hypothesis, and there is a semi-WSNM M' such that $|M'| = k$, $(m_{i'}, w_{j'})$ is the maximum pair of M', and $(m_{i'}, w_{j'})$ and (m_i, w_j) are nonconflicting. Then $M = M' \cup \{(m_i, w_j)\}$ is a semi-WSNM such that (m_i, w_j) is the maximum pair and $|M| = |M'| + 1 = k+1$, implying that $X(i, j) = k+1$. This contradicts the assumption that $X(i, j) = -\infty$ and the proof is completed. □

Now we analyze time-complexity of the algorithm. Computing each $Y(0, 0)$, $Y(0, j)$, and $Y(i,0)$ can be done in constant time. For computing one $Y(i,j)$ according to Equation (5.4), there are $O(n^2)$ candidates for (i',j'). For each (i',j'), checking if $(m_{i'}, w_{j'})$ and (m_i, w_j) are conflicting can be done in constant time with $O(n^4)$-time preprocessing described in subsequent paragraphs. Therefore one $Y(i,j)$ can be computed in time $O(n^2)$. Since there are $O(n^2)$ $Y(i,j)$s, the time-complexity for computing all $Y(i,j)$s is $O(n^4)$. Adding the $O(n^4)$-time for preprocessing mentioned above, the total time-complexity of the algorithm is $O(n^4)$.

In the preprocessing, we construct three tables S, A, and B.

- S is a $\Theta(n^4)$-sized four-dimensional table that takes logical values 0 and 1. For $0 \le i' \le i \le n+1$ and $0 \le j' \le j \le n+1$, $S(i',i,j',j) = 1$ if and only if there exists at least one acceptable pair (m,w) such that $m \in \{m_{i'}, m_{i'+1}, \ldots, m_i\}$ and $w \in \{w_{j'}, w_{j'+1}, \ldots, w_j\}$. Since $S(i,i,j,j) = 1$ if and only if (m_i, w_j) is an acceptable pair, it can be computed in constant time. In general, $S(i',i,j',j)$ can be computed in constant time as follows.

$$S(i',i,j',j) = \begin{cases} 1 \quad \text{(if } (m_i, w_j) \text{ is an acceptable pair)} \\ S(i',i-1,j',j) \vee S(i',i,j',j-1) \quad \text{(otherwise)} \end{cases}$$

  Hence S can be constructed in $O(n^4)$ time by a simple dynamic programming.
- A is a $\Theta(n^3)$-sized table. For convenience, we introduce an imaginary person λ who is acceptable to any person, where $q \succ_p \lambda$ holds for any person p and any person q acceptable to p. For $0 \le i \le n+1$ and $0 \le j' \le j \le n+1$, $A(i,j',j)$ stores the woman whom m_i most prefers among $\{w_{j'}, \ldots, w_j, \lambda\}$. Then, for i and j, $A(i,j,j) = w_j$ if (m_i, w_j) is an acceptable pair and $A(i,j,j) = \lambda$ otherwise. $A(i,j',j)$ can be computed as the better of $A(i,j',j-1)$ and $A(i,j,j)$ in m_i's list. By the above arguments, each element of A can be computed in constant time and hence A can be constructed in $O(n^3)$ time.
- B plays a symmetric role to A; for $0 \le i' \le i \le n+1$ and $0 \le j \le n+1$, $B(i',i,j)$ stores the man whom w_j most prefers among $\{m_{i'}, \ldots, m_i, \lambda\}$. B can also be constructed in $O(n^3)$ time.

It is easy to see that $(m_{i'}, w_{j'})$ and (m_i, w_j) are conflicting if and only if one of the following conditions hold. Condition 1 can be clearly checked in constant time. Thanks to the preprocessing, Conditions 2–4 can also be checked in constant time.

1. $(m_{i'}, w_j)$ or $(m_i, w_{j'})$ is a blocking pair for the matching $\{(m_{i'}, w_{j'}), (m_i, w_j)\}$.

2. $S(i'+1, i-1, j'+1, j-1) = 1$. (If this holds, there is a blocking pair (m, w) such that $m \in \{m_{i'+1}, m_{i'+2}, \ldots, m_{i-1}\}$ and $w \in \{w_{j'+1}, w_{j'+2}, \ldots, w_{j-1}\}$.)
3. m_i prefers $A(i, j'+1, j-1)$ to w_j or $m_{i'}$ prefers $A(i', j'+1, j-1)$ to $w_{j'}$. (If this holds, there exists a blocking pair (m, w) such that $m \in \{m_{i'}, m_i\}$ and $w \in \{w_{j'+1}, \ldots, w_{j-1}\}$.)
4. w_j prefers $B(i'+1, i-1, j)$ to m_i or $w_{j'}$ prefers $B(i'+1, i-1, j')$ to $m_{i'}$. (If this holds, there exists a blocking pair (m, w) such that $m \in \{m_{i'+1}, \ldots, m_{i-1}\}$ and $w \in \{w_{j'}, w_j\}$).

This completes the explanation on preprocessing, and from the discussion so far, we have the following theorem:

Theorem 10. *There exists an $O(n^4)$-time algorithm to find a maximum cardinality WSNM, given an SMI-instance.*

5.3.2 Algorithm for SMTI

Similarly to the SMI case, a weak-WSNM exists in any SMTI-instance, as remarked in page 415 of [RI19]: Given an SMTI-instance I, break all the ties arbitrarily and obtain an SMI-instance I'. Let M be a WSNM of I'. Then it is not hard to see that M is also a weak-WSNM of I. In contrast, there is a simple instance that admits neither a strong- nor a super-WSNM (Fig. 5.8). The empty matching is blocked by any acceptable pair. The matching $\{(m_1, w_1)\}$ is blocked by (m_2, w_2). The matching $\{(m_2, w_2)\}$ is blocked by (m_1, w_1). The matching $\{(m_1, w_1), (m_2, w_2)\}$ is blocked by (m_2, w_1). The matching $\{(m_2, w_1)\}$ is blocked by (m_1, w_1).

m_1:	w_1		w_1:	$(m_1 \quad m_2)$
m_2:	$w_1 \quad w_2$		w_2:	m_2

Fig. 5.8 An instance that admits neither a strong-WSNM nor a super-WSNM

Nevertheless, the algorithm in Section 5.3.1 can be applied to SMTI straightforwardly. Necessary modifications are summarized as follows:

- As mentioned above, existence of a WSNM is not guaranteed. If our algorithm outputs $Y(n+1, n+1) = -\infty$, then it means that no solution exists.
- The definition of two edges (m_i, w_j) and (m_x, w_y) being conflicting must be

modified depending on one of the three stability notions.

- The definitions of the tables A and B need to be modified as follows. $A(i, j', j)$ stores *one of* the women whom m_i most prefers among $\{w_{j'}, \ldots, w_j, \lambda\}$. Similarly, $B(i', i, j)$ stores *one of* the men whom w_j most prefers among $\{m_{i'}, \ldots, m_i, \lambda\}$.
- In the SMI case, $A(i, j', j)$ is computed as *the better* of $A(i, j', j-1)$ and $A(i, j, j)$ in m_i's list. But now it can happen that $A(i, j', j-1) =_{m_i} A(i, j, j)$, in which case $A(i, j', j)$ can be either $A(i, j', j-1)$ or $A(i, j, j)$. (Strictly speaking, this treatment was needed already in the SMI case because there can be a case that $A(i, j', j-1) = A(i, j, j) = \lambda$, but there we took simplicity.)
- Conditions 3 and 4 in checking confliction of two edges need to be modified as follows. In the super- and strong stabilities, "prefers" should be replaced by "weakly prefers". In the weak stability, "prefers" should be replaced by "strictly prefers".

With these modifications, whether two edges are conflicting or not can be checked in constant time. Therefore, we have the following corollary:

Corollary 4. *There exists an $O(n^4)$-time algorithm to find a maximum cardinality super-WSNM (strong-WSNM, weak-WSNM) or report that none exists, given an SMTI-instance.*

5.4 Concluding Remarks

In this chapter, we gave algorithms for determining existence of an SSNM and finding a largest WSNM. We showed that our algorithms are applicable to extensions where preference lists may include ties, except for one case which we show to be NP-complete. This NP-completeness holds even if each person's preference list is of length at most two and ties appear in only men's preference lists. To complement this intractability, we also showed that the problem is solvable in polynomial time if the length of preference lists of one side is bounded by one (but that of the other side is unbounded).

One of interesting future works is to consider optimization problems. For example, in SMI we have shown that it is easy to determine if there exists an SSNM with zero-crossing. What is the complexity of the problem of finding an SSNM with the minimum number of crossings, and if it is NP-hard, is there a good approximation algorithm for it? Also, it might be interesting to consider noncrossing stable matchings for other placements of agents, e.g., on a circle or on general position in 2-dimensional plane.

Chapter 6

Strategy-Proof Approximation Algorithms for SMTI

In this chapter, we consider the strategy-proofness in MAX SMTI, and investigate the trade-off between strategy-proofness and approximability.

In the case of incomplete preference lists, there may be unmatched (i.e., single) persons. Thus, we have to extend the definition of a person preferring one matching to another. We say that a person p prefers M' to M if either $M'(p) \succ_p M(p)$ holds or p is single in M but is matched in M' with some acceptable woman. Then the definition of strategy-proofness for SM naturally takes over to SMTI.

Since SMTI is a generalization of SM, Roth's impossibility theorem for SM [Rot82] holds also for MAX SMTI (regardless of approximability): That is, there is no strategy-proof stable mechanism for MAX SMTI. Therefore, we focus on *one-sided*-strategy-proofness. We first show that MAX SMTI admits a 2-approximate-stable mechanism, which is achieved by a simple extension of the GS algorithm. We also show that this result is tight. We next consider a restricted version, MAX SMTI-1T. Throughout the chapter, we assume that ties appear in men's lists only (and women's lists must be strict). In the following, we use the name *MAX SMTI-1TM* to stress that only men's preference lists may contain ties. As for woman-strategy-proofness, we obtain the same result as for MAX SMTI. That is, MAX SMTI-1TM admits a woman-strategy-proof 2-approximate-stable mechanism, and this result is tight. For man-strategy-proofness, we can reduce the approximation ratio to 1.5, which is the main result of this chapter.

We remark that no assumptions on running times are made for our negative results, while algorithms in our positive results run in linear time. Note also that the current best polynomial-time approximation algorithms for MAX SMTI and MAX SMTI-

1TM have the approximation ratios better than those in our negative results. Hence our results provide gaps between polynomial-time computation and strategy-proof computation.

6.1 Algorithms for MAX SMTI

In this section, we show that there is a 2-approximate-stable mechanism for MAX SMTI. It is achieved by a simple extension of the GS algorithm. We also show that this result is tight. We start with the positive part:

Lemma 24. *MAX SMTI admits both a man-strategy-proof* 2*-approximate-stable mechanism and a woman-strategy-proof* 2*-approximate-stable mechanism.*

Proof. Consider a mechanism S^* that is described by the following algorithm. Given a MAX SMTI instance I, S^* first breaks each tie so that persons in a tie are ordered increasingly in their indices, that is, if q_i and q_j are in the same tie of p's list, then after the tie break $q_i \succ_p q_j$ holds if and only if $i < j$. Let I' be the resulting instance. Its preference lists are incomplete but do not include ties. That is, I' is an SMI-instance. It then applies MGS for SMI to I' and obtains a stable matching M for I'. It is easy to see that M is stable for I. Also it is well-known that in MAX SMTI, any stable matching is a 2-approximate solution [MII+02]. Hence S^* is a 2-approximate-stable mechanism.

We then show that S^* is a man-strategy-proof mechanism. Suppose not. Then there is a MAX SMTI instance I and a man m who has a successful strategy in I. Let J be a MAX SMTI instance in which only m's preference list differs from I, and by using it m obtains a better outcome. Let M_I and M_J be the outputs of S^* on I and J, respectively. Then m prefers M_J to M_I, that is, either (i) $M_J(m) \succ_m M_I(m)$ with respect to m's true preference list in I, or (ii) m is single in M_I and matched in M_J, and $M_J(m)$ is acceptable to m in I. Let I' and J', respectively, be the SMI-instances constructed from I and J by breaking ties in the above mentioned manner. Then M_I and M_J are, respectively, the results of MGS applied to I' and J'. Since I' is the result of tie-breaking of I and m prefers M_J to M_I in I, m prefers M_J to M_I in I'. Note that, due to the tie-breaking rule, the preference lists of people except for m are same in I' and J'. This means that when MGS is used in SMI, m can have a successful strategy in I' (i.e., to change his list to that of J'), contradicting man-strategy-proofness of MGS for SMI (page 57 of [GI89]).

If we exchange the roles of men and women in S^*, we obtain a woman-strategy-proof 2-approximate-stable mechanism. □

We then show the negative part. We remark that ϵ is not necessarily a constant.

Lemma 25. *(1) For any positive ϵ, there is no man-strategy-proof $(2-\epsilon)$-approximate-stable mechanism for MAX SMTI, even if ties appear in only women's preference lists. (2) For any positive ϵ, there is no woman-strategy-proof $(2-\epsilon)$-approximate-stable mechanism for MAX SMTI, even if ties appear in only men's preference lists.*

Proof. (1) Consider the instance I_1 given in Fig. 6.1, where m_3's preference list is empty. It is straightforward to verify that I_1 has two stable matchings $M_1 = \{(m_1, w_1), (m_2, w_2)\}$ and $M_2 = \{(m_1, w_2), (m_2, w_3)\}$, both of which are of maximum size.

m_1:	w_2	w_1		w_1:	m_1	
m_2:	w_2	w_3		w_2:	$(m_1$	$m_2)$
m_3:				w_3:	m_2	

Fig. 6.1 A MAX SMTI instance I_1

Let S be an arbitrary $(2-\epsilon)$-approximate-stable mechanism for MAX SMTI. Since S is a stable mechanism, it must output either M_1 or M_2 on I_1. First suppose that it outputs M_1. Let I'_1 be the instance obtained from I_1 by deleting w_1 from m_1's preference list. Then since M_2 is still a stable matching for I'_1 and S is a $(2-\epsilon)$-approximate-stable mechanism, S must output a stable matching of size 2. But since M_2 is now the only stable matching of size 2, S outputs M_2 on I'_1. Thus m_1 can obtain a better partner by manipulating his preference list. On the other hand, suppose that S outputs M_2 on I_1. Then let I''_1 be the instance obtained from I_1 by deleting w_3 from m_2's preference list. By a similar argument, S must output M_1 on I''_1 and hence m_2 can obtain a better partner by manipulation. We have shown that, for any $(2-\epsilon)$-approximate-stable mechanism S, some man has a successful strategy in I_1 and hence S is not a man-strategy-proof mechanism.

(2) We use the instance I_2 given in Fig. 6.2, which is symmetric to I_1. By the same argument as above, we can show that for any $(2-\epsilon)$-approximate-stable mechanism S, some woman has a successful strategy in I_2 and hence S is not a woman-strategy-proof

m_1:	w_1			w_1:	m_2	m_1
m_2:	$(w_1$	$w_2)$		w_2:	m_2	m_3
m_3:	w_2			w_3:		

Fig. 6.2 A MAX SMTI instance I_2

mechanism. □

Thus, we have the following theorem:

Theorem 11. *MAX SMTI admits both a man-strategy-proof 2-approximate-stable mechanism and a woman-strategy-proof 2-approximate-stable mechanism. On the other hand, for any positive ϵ, MAX SMTI admits neither a man-strategy-proof $(2-\epsilon)$-approximate-stable mechanism nor a woman-strategy-proof $(2-\epsilon)$-approximate-stable mechanism.*

6.2 Non-strategy-proofness of Existing Algorithms

Since MGS is a man-strategy-proof stable mechanism for SM, such types of algorithms are good candidates for man-strategy-proof 1.5-approximation mechanism for MAX SMTI-1TM. Existing 1.5-approximation algorithms for MAX SMTI for one-sided ties are of GS-type, but in these algorithms, proposals are made from the side with no ties (women, in our case), so we cannot use them for our purpose. On the other hand, there are 1.5-approximation algorithms for the general MAX SMTI [McD09, Pal14, Kir13], which are fortunately of GS-type and can handle proposals from the side with ties (men, in our case). Hence one may expect that these algorithms will work. However, it is not the case.

Király [Kir13] presented a 1.5-approximation algorithm for general MAX SMTI (i.e., ties can appear on both sides), which is named "New Algorithm". We modify it in the following two respects.

1. Men's proposals do not get into the second round.
2. When there is arbitrarity, the person with the smallest index is prioritized.

Ideas behind these modifications are as follows: For item 1, since there is no ties in women's preference lists, executing the second round does not change the result.

The role of item 2 is to make the algorithm deterministic, so that the output is a function of an input (as we did in the proof of Lemma 24). For completeness, we give a pseudo-code of the algorithm, denoted M-KNA to stand for "Modified Király's New Algorithm", in Algorithm 5.

Each person takes one of three states, "free", "engaged", and "semi-engaged". Initially, all the persons are free. At lines 5, 10 and 14, man m proposes to woman w. Basically, the procedure is exactly the same as that of MGS. If w is free, then we let $M := M \cup \{(m, w)\}$ and both m and w be engaged (we say w *accepts* m). If w is engaged to m' (i.e., $(m', w) \in M$) and if $m \succ_w m'$, then we let $M := M \cup \{(m, w)\} \setminus \{(m', w)\}$, m be engaged, and m' be free. We also delete w from m''s preference list (we say w *accepts* m and *rejects* m'). If w is engaged to m' and $m' \succ_w m$, then we delete w from m's preference list (we say w *rejects* m).

There is an exception in the acceptance/rejection rule of a woman, when she receives the first and second proposals. This is actually the key for guaranteeing 1.5-approximation, but this rule is not used in the subsequent counter-example so we omit it here. Readers may consult to the original paper [Kir13] for the full description of the algorithm.

It is already proved that the (original) Király's algorithm always outputs a stable matching which is a 1.5-approximate solution, and it is not hard to see that the same results hold for the above M-KNA for MAX SMTI-1TM. However, as the example in Figs. 6.3 and 6.4 shows, it is not a man-strategy-proof mechanism.

m_1:	w_2	w_1	w_1:	m_2	m_4	m_1
m_2:	$(w_1$	$w_3)$	w_2:	m_4	m_1	
m_3:	w_3		w_3:	m_2	m_3	
m_4:	w_1	w_2	w_4:			

Fig. 6.3 A counter-example (true lists)

If M-KNA is applied to the true preference lists in Fig. 6.3, the obtained matching is $\{(m_2, w_1), (m_3, w_3), (m_4, w_2)\}$. Suppose that m_1 flips the order of w_1 and w_2 (Fig. 6.4). This time, M-KNA outputs $\{(m_1, w_2), (m_2, w_3), (m_4, w_1)\}$ and m_1 successfully obtains a partner w_2. By proposing to w_1 first, m_1 is able to let m_2 propose to w_3. This allows m_4 to obtain w_1, which prevents m_4 from proposing to w_2. This

Algorithm 5 Modified Király's New Algorithm (M-KNA) [Kir13]

```
Let M := ∅ and all people be free.
while there is a free man whose preference list is non-empty do
  Among those men, let m be the one with the smallest index.
  if the top of m's current preference list consists of only one woman w then
    Let m propose to w.
  end if
  if the top of m's current preference list is a tie then
    if all the women in the tie are engaged then
      Among those women, let w be the one with the smallest index.
      Let m propose to w.
    end if
    if there is a free woman in the tie then
      Among those free women, let w be the one with the smallest index.
      Let m propose to w.
    end if
  end if
end while
Output M.
```

m_1:	w_1	w_2	w_1:	m_2	m_4	m_1
m_2:	$(w_1$	$w_3)$	w_2:	m_4	m_1	
m_3:	w_3		w_3:	m_2	m_3	
m_4:	w_1	w_2	w_4:			

Fig. 6.4 A counter-example (manipulated by m_1)

eventually makes it possible for m_1 to obtain w_2.

We finally remark that the same example shows that the other two 1.5-approximation algorithms [McD09, Pal14] (with the tie-breaking rule 2 above) are not man-strategy-proof mechanisms either.

6.3 Algorithms for MAX SMTI-1TM

Recall that MAX SMTI-1TM is a restriction of MAX SMTI where ties can appear in men's preference lists only. Then the following corollary is immediate from Lemma 24 and Lemma 25(2).

Corollary 5. *MAX SMTI-1TM admits a woman-strategy-proof 2-approximate-stable mechanism, but no woman-strategy-proof* $(2-\epsilon)$*-approximate-stable mechanism for any positive* ϵ*.*

We then move to man-strategy-proofness. For man-strategy-proofness, we can reduce the approximation ratio to 1.5. We start with the negative part:

Lemma 26. *For any positive* ϵ*, there is no man-strategy-proof* $(1.5-\epsilon)$*-approximate-stable mechanism for MAX SMTI-1TM.*

Proof. The proof goes like that of Lemma 25. Consider the instance I_3 in Fig. 6.5. I_3 has four matchings of size 3, namely, $M_3 = \{(m_1, w_1), (m_2, w_2), (m_3, w_3)\}$, $M_4 = \{(m_1, w_1), (m_2, w_2), (m_3, w_4)\}$, $M_5 = \{(m_1, w_1), (m_2, w_3), (m_3, w_4)\}$, and $M_6 = \{(m_1, w_2), (m_2, w_3), (m_3, w_4)\}$. Among them, M_3 and M_6 are stable (M_4 is blocked by (m_3, w_3) and M_5 is blocked by (m_1, w_2)). Hence any $(1.5-\epsilon)$-approximate-stable mechanism outputs either M_3 or M_6, since a stable matching of size 2 is not a $(1.5-\epsilon)$-approximate solution.

m_1:	w_2	w_1		w_1:	m_1	
m_2:	$(w_2$	$w_3)$		w_2:	m_2	m_1
m_3:	w_3	w_4		w_3:	m_2	m_3
m_4:				w_4:	m_3	

Fig. 6.5 A MAX SMTI-1TM instance I_3

Consider an arbitrary $(1.5-\epsilon)$-approximate-stable mechanism S for MAX SMTI-1TM, and suppose that S outputs M_3 on I_3. Then if m_1 deletes w_1 from the list, M_6 is the unique maximum stable matching (of size 3); hence S must output M_6 and so m_1 can obtain a better partner w_2. Similarly, if S outputs M_6 on I_3, m_3 can force S to output M_3 by deleting w_4 from the list. In either case, some man has a successful strategy in I_3 and hence S is not a man-strategy-proof mechanism. □

Finally, we give a proof for the positive part, which is the main result of this chapter.

Lemma 27. *There exists a man-strategy-proof* 1.5*-approximate-stable mechanism for MAX SMTI-1TM.*

Proof. We give Algorithm 6 and show that it is a man-strategy-proof 1.5-approximate-stable mechanism by three subsequent lemmas (Lemmas 28 to 30). Algorithm 6 first translates an SMTI-1TM instance I to an SMI-instance I' using Algorithm 7, then applies MGS to I' and obtains a matching M', and finally constructs a matching M of I from M'. The new instance I' contains $2n$ men a_i and b_j ($1 \leq i \leq n$, $1 \leq j \leq n$) and $2n$ women s_j and t_j ($1 \leq j \leq n$) (lines 2 and 3 of Algorithm 7). It is important to note that a man a_i corresponds to a man m_i of I, while a man b_j and two women s_j and t_j correspond to a woman w_j of I. As will be seen later, b_j is definitely matched with s_j or t_j in M', and the other woman (i.e., either s_j or t_j who is not matched with b_j) plays a role of woman w_j of I: If she is single in M', then w_j is single in M. If she is matched with a_i in M', then w_j is matched with m_i in M.

Algorithm 6 An algorithm for MAX SMTI-1TM

Require: An instance I for MAX SMTI-1TM.

Ensure: A matching M for I.

1: Construct an SMI-instance I' from I using Algorithm 7.

2: Apply MGS to I' and obtain a matching M'.

3: Let $M := \{(m_i, w_j) \mid (a_i, s_j) \in M' \vee (a_i, t_j) \in M'\}$ and output M.

We briefly give a high-level idea behind Algorithm 6. Consider an application of MGS to I' at line 2. Since men's proposal order does not affect the outcome, it is convenient to first let b_j propose to his first choice woman s_j for each j. At this moment, there are n pairs (b_j, s_j) ($1 \leq j \leq n$). We regard this as an initial state, and as long as (b_j, s_j) is a pair, t_j acts as w_j. At some point, if s_j receives a proposal from some man a_i for the first time, s_j rejects b_j and b_j then proposes to his second choice woman t_j, which is accepted. We regard this as a change of the state, and the role of w_j is taken over to s_j. Once this happens, (b_j, t_j) remains a pair till the end of the algorithm. Recall that at line 4 of Algorithm 7, each man makes two copies of each tie. This is regarded as allowing a man to propose to woman w_j twice, first to t_j and second to s_j.

Algorithm 7 Translating instances

Require: An instance I for MAX SMTI-1TM.

Ensure: An instance I' for SMI.

1: Let X and Y be the sets of men and women of I, respectively.

2: Let $X' := \{a_i \mid m_i \in X\} \cup \{b_j \mid w_j \in Y\}$ be the set of men of I'.

3: Let $Y' := \{s_j \mid w_j \in Y\} \cup \{t_j \mid w_j \in Y\}$ be the set of women of I'.

4: Each a_i's list is constructed as follows: Consider a tie $(w_{j_1}\ w_{j_2}\ \cdots\ w_{j_k})$ in m_i's list in I. We assume without loss of generality that $j_1 < j_2 < \cdots < j_k$. (If not, just arrange the order, which does not change the instance.) Replace each tie $(w_{j_1}\ w_{j_2}\ \cdots\ w_{j_k})$ by a strict order of $2k$ women $t_{j_1}\ t_{j_2}\ \cdots\ t_{j_k}\ s_{j_1}\ s_{j_2}\ \cdots\ s_{j_k}$. A woman who is not included in a tie is considered as a tie of length one.

5: Each b_j's list is defined as "$b_j : s_j\ \ t_j$".

6: For each j, let $P(w_j)$ be the list of w_j in I, and $Q(w_j)$ be the list obtained from $P(w_j)$ by replacing each man m_i by a_i. Then s_j and t_j's lists are defined as follows:

$$\begin{array}{lll} s_j : & Q(w_j) & b_j \\ t_j : & b_j & Q(w_j) \end{array}$$

With these observations in mind, we can see that MGS for I' simulates the following GS-type algorithm for the original MAX SMTI instance I.

- Each free man proposes to a woman from the top of the list. When he encounters a tie T, he proposes to the women in T in a predetermined order (i.e., smaller index first). If he is rejected by all of them, he starts the second sequence of proposals to the women in T in the same order. If he is rejected by all the women in T again, then he proceeds to the next tie.
- Each woman's acceptance/rejection policy is as follows: If two proposals are first proposals, she respects her preference list. Similarly, if both are second proposals, she respects her preference list. If one is a first proposal and the other is a second proposal, she always chooses the second proposal (regardless of her list). Hence, once a woman receives a second proposal of some man, she never accepts a first proposal thereafter.

This algorithm achieves an approximation ratio of 1.5 for MAX SMTI, although we do not prove it here. A beneficial point of this algorithm is that a man's proposal order is predetermined and is not affected by other persons' states. As we explained in Section 6.2, absence of this property prevented existing algorithms from being man-strategy-proof.

The reason why we do not use this algorithm directly but translate it to an algorithm using MGS for SMI is to make the proof of man-strategy-proofness simpler; this translation allows us to attribute man-strategy-proofness of Algorithm 6 to that of MGS for SMI, as we did in the proof of Lemma 24.

Now we start formal proofs for the correctness.

Lemma 28. *Algorithm 6 always outputs a stable matching.*

Proof. Let M be the output of Algorithm 6 and M' be the matching obtained at line 2 of Algorithm 6. We first show that M is a matching. Since M' is a matching, a_i appears at most once in M', so m_i appears at most once in M. Observe that b_j is matched in M', as otherwise (b_j, t_j) blocks M', contradicting the stability of M' in I'. Hence at most one of s_j and t_j can be matched with a_i for some i, which implies that w_j appears at most once in M. Thus M is a matching.

We then show the stability of M. Since M' is the output of MGS, it is stable in I'. Now suppose that M is unstable in I and there is a blocking pair (m_i, w_j) for M. There are four cases:

Case (i): both $\boldsymbol{m_i}$ and $\boldsymbol{w_j}$ are single. Since m_i is single in M, line 3 of Algorithm 6 implies that a_i is single in M'. Since w_j is single in M, s_j is not matched in M' with anyone in $Q(w_j)$, i.e., s_j is single or matched with b_j. Note that (a_i, s_j) is a mutually acceptable pair because (m_i, w_j) is a blocking pair, and $a_i \succ_{s_j} b_j$ in I' by construction. Thus (a_i, s_j) blocks M', a contradiction.

Case (ii): $\boldsymbol{w_j \succ_{m_i} M(m_i)}$ and $\boldsymbol{w_j}$ is single. Let $M(m_i) = w_k$. Then, by construction of M, $M'(a_i)$ is either s_k or t_k. By construction of I', $w_j \succ_{m_i} w_k$ implies both $s_j \succ_{a_i} s_k$ and $s_j \succ_{a_i} t_k$, and in either case we have that $s_j \succ_{a_i} M'(a_i)$ in I'. Since w_j is single in M, by the same argument as Case (i), s_j is either single or matched with b_j in M'. Hence (a_i, s_j) blocks M'.

Case (iii): $\boldsymbol{m_i}$ is single and $\boldsymbol{m_i \succ_{w_j} M(w_j)}$. Since m_i is single in M, a_i is single in M' by the same argument as Case (i). Let $M(w_j) = m_k$. Then, by construction of M, either s_j or t_j is matched with a_k, and the other is matched with b_j since b_j can never be single as we have seen in an earlier stage of this proof. In particular, $M'(s_j)$ is either a_k or b_j. Note that $m_i \succ_{w_j} m_k$ in $P(w_j)$ implies

$a_i \succ_{s_j} a_k$ in $Q(w_j)$, so in either case $a_i \succ_{s_j} M'(s_j)$ in I' due to the construction of s_j's list. Therefore (a_i, s_j) blocks M'.

Case (iv): $\boldsymbol{w_j \succ_{m_i} M(m_i)}$ **and** $\boldsymbol{m_i \succ_{w_j} M(w_j)}$. By the same argument as Case (ii), we have that $s_j \succ_{a_i} M'(a_i)$ in I'. By the same argument as Case (iii), we have that $a_i \succ_{s_j} M'(s_j)$ in I'. Hence (a_i, s_j) blocks M'.

□

Lemma 29. *Algorithm 6 always outputs a 1.5-approximate solution.*

Proof. Let I be an input, M_{opt} be a maximum stable matching for I, and M be the output of Algorithm 6. We show that $\frac{|M_{opt}|}{|M|} \leq 1.5$. Let $G = (X \cup Y, E)$ be a bipartite (multi-)graph with vertex bipartition X and Y, where X corresponds to men and Y corresponds to women of I. The edge set E is a union of M and M_{opt}, that is, $(m_i, w_j) \in E$ if and only if (m_i, w_j) is a pair in M or M_{opt}. If (m_i, w_j) is a pair in both M and M_{opt}, then E contains two edges (m_i, w_j), which constitute a "cycle" of length two. An edge in E corresponding to M (M_{opt}, respectively) is called an M-edge (M_{opt}-edge, respectively). Since the degree of each vertex of G is at most 2, each connected component of G is an isolated vertex, a cycle, or a path.

It is easy to see that G does not contain a single M_{opt}-edge as a connected component, since if such an edge (m_i, w_j) exists, then (m_i, w_j) is a blocking pair for M, contradicting the stability of M. In the following, we show that G does not contain, as a connected component, a path of length three $m_i - w_j - m_k - w_\ell$ such that (m_i, w_j) and (m_k, w_ℓ) are M_{opt}-edges and (m_k, w_j) is an M-edge. If this is true, then for any connected component C of G, the number of M-edges in C is at least two-thirds of the number of M_{opt}-edges in C, implying $\frac{|M_{opt}|}{|M|} \leq 1.5$.

Suppose that such a path exists. Note that m_i and w_ℓ are single in M. If $m_i \succ_{w_j} m_k$, then (m_i, w_j) blocks M. Since women's preference lists do not contain ties, we have that $m_k \succ_{w_j} m_i$. If $w_\ell \succ_{m_k} w_j$, then (m_k, w_ℓ) blocks M. If $w_j \succ_{m_k} w_\ell$, then (m_k, w_j) blocks M_{opt}. Hence w_j and w_ℓ are tied in m_k's list. Then by construction of I', (i) $t_\ell \succ_{a_k} s_j$. (Hereafter, referring to Fig. 6.6 would be helpful. Here, the order of t_j and t_ℓ in a_k's list is uncertain, i.e., it may be the opposite, but this order is not important in the rest of the proof.) Since w_ℓ is single in M, either s_ℓ or t_ℓ is single in M'. If s_ℓ is single in M', then (b_ℓ, s_ℓ) blocks M', a contradiction. Hence (ii) t_ℓ is single in M'. Since $M(m_k) = w_j$, either $M'(a_k) = s_j$ or $M'(a_k) = t_j$ holds. In the former case, (i) and (ii) above imply that (a_k, t_ℓ) blocks M', so assume the latter,

a_i: $\cdots$ s_j $\cdots$	s_j: $\cdots$ a_i $\cdots$ b_j
b_i: s_i t_i	t_j: b_j $\cdots$ a_k $\cdots$
a_k: $\cdots$ t_j $\cdots$ t_ℓ $\cdots$ s_j $\cdots$	s_ℓ: $\cdots$ b_ℓ
b_k: s_k t_k	t_ℓ: b_ℓ $\cdots$
a_ℓ: $\cdots$	
b_ℓ: s_ℓ t_ℓ	

Fig. 6.6 A part of the preference lists of I'

i.e., $M'(a_k) = t_j$. Recall from the proof of Lemma 28 that either s_j or t_j is matched with b_j in M', so $M'(s_j) = b_j$. Since (m_i, w_j) is an acceptable pair in I, we have that $a_i \succ_{s_j} b_j$ due to the construction of s_j's list. Since m_i is single in M, a_i is single in M'. Hence (a_i, s_j) blocks M', a contradiction. □

Lemma 30. *Algorithm 6 is a man-strategy-proof mechanism.*

Proof. The proof is similar to that of Lemma 24. Suppose that Algorithm 6 is not a man-strategy-proof mechanism. Then there are MAX SMTI-1TM instances I and J and a man m_i having the following properties: I and J differ in only m_i's preference list, and m_i prefers M_J to M_I, where M_I and M_J are the outputs of Algorithm 6 for I and J, respectively. Then either (i) $M_J(m_i) \succ_{m_i} M_I(m_i)$ in I, or (ii) m_i is single in M_I and $M_J(m_i)$ is acceptable to m_i in I.

Let I' and J' be the SMI-instances constructed by Algorithm 7. Since I and J differ in only m_i's preference list, I' and J' differ in only a_i's preference list. Let $M_{I'}$ and $M_{J'}$, respectively, be the outputs of MGS applied to I' and J'. In case of (i), we have that $M_{J'}(a_i) \succ_{a_i} M_{I'}(a_i)$ in I', due to line 4 of Algorithm 7 and line 3 of Algorithm 6. In case of (ii), a_i is single in $M_{I'}$ because m_i is single in M_I, and $M_{J'}(a_i)$ is acceptable to a_i in I' because $M_J(m_i)$ is acceptable to m_i in I. This implies that a_i has a successful strategy in I', contradicting man-strategy-proofness of MGS for SMI [GI89]. □

By Lemmas 28 to 30, we can conclude that Algorithm 6 is a man-strategy-proof

1.5-approximate-stable mechanism for MAX SMTI-1TM. □

Thus, we have the following theorem:

Theorem 12. *MAX SMTI-1TM admits a man-strategy-proof 1.5-approximate-stable mechanism, but no man-strategy-proof* $(1.5-\epsilon)$*-approximate-stable mechanism for any positive* ϵ*.*

6.4 Extensions

In the above discussion, man-strategy-proofness (woman-strategy-proofness) is defined in terms of a manipulation of a preference list by one man (woman). We can extend this notion to a *coalition* of men (or women) as follows; a coalition C of men has a successful strategy if there is a way of falsifying preference lists of members of C which improves the outcome of *every* member of C. It is known that MGS is strategy-proof against a coalition of men in this sense (Theorem 1.7.1 of [GI89]), and this strategy-proofness holds also in SMI (page 57 of [GI89]). Since all our strategy-proofness results (Lemmas 24 and 30) are attributed to strategy-proofness of MGS in SMI, we can easily modify the proofs so that Theorem 11, Corollary 5, and Theorem 12 hold for strategy-proofness against coalitions.

Clearly, the negative parts of Theorem 11, Corollary 5, and Theorem 12 hold for a many-to-one extension of MAX SMTI, denoted *MAX HRT*. Also, we can show that man-strategy-proofness in Theorems 11 and 12 carry over to resident-strategy-proofness in MAX HRT by cloning hospitals (see e.g., page 283 of [IM08] for cloning). By contrast, woman-strategy-proofness in Theorem 11 and Corollary 5 do not hold for hospital-strategy-proofness in MAX HRT; there is no hospital-strategy-proof stable mechanism even without ties (see Section 1.7.3 of [GI89]).

When only ties are present (SMT) or only incomplete lists are present (SMI), all the stable matchings of one instance have the same cardinality. The former is due to the fact that any stable matching is a perfect matching, and the latter is due to the Rural Hospitals theorem [GS85, Rot84, Rot86]. Hence approximability is not an important issue in these cases. As for strategy-proofness, since SMT and SMI are generalizations of SM, Roth's impossibility theorem holds and no strategy-proof stable mechanism exists. Existence of one-sided strategy-proofness for SMI is already known as we have mentioned above, and that for SMT follows directly from Theorem 11.

6.5 Concluding Remarks

In this chapter, we first gave a man-strategy-proof 2-approximate-stable mechanism and a woman-strategy-proof 2-approximate-stable mechanism for MAX SMTI. We also considered a restricted variant of MAX SMTI, which we call MAX SMTI-1TM, where only men's lists can contain ties (and women's lists must be strictly ordered). Then we gave a woman-strategy-proof 2-approximate-stable mechanism and a man-strategy-proof 1.5-approximate-stable mechanism for MAX SMTI-1TM. All these results are tight in terms of approximation ratios.

Considering strategy-proof algorithms for other stable matching problems is interesting. Since our technique of obtaining strategy-proofness by constructing an algorithm as a translation of an instance and applying existing strategy-proof algorithm is generic, it may be useful for constructing strategy-proof algorithms for other problems. Recently, it is used to construct strategy-proof algorithms for some stable matching problems [GMMY22, Yok21].

Chapter 7

Conclusion

In this thesis, we studied computational tractability of various extensions of stable matching problems. In Chapter 3, we improved inapproximability of MAX SIZE MIN BP SMI. In Chapter 4, we defined HRLQ and showed some positive and negative results on its optimization variants. In Chapter 5, we positively solved two open problems on noncrossing stable matchings in SMI, and extended them to SMTI. In Chapter 6, we considered strategy-proof approximation algorithms for MAX SMTI and showed polynomial-time approximation algorithms that have tight approximation ratios. These results contribute to the understanding of computational tractability of complex problems for further applications of stable matching problems in the real world.

Our results have also made several contributions to the overall study of stable matching problems. One contribution is to strengthen the common understanding that minimizing the number of BPs is difficult. Although the number of BPs is a natural and effective measure of the degree of instability [EH08], minimizing it in SR and SRT [ABM05] and in MAX SMI [BMM10] were shown to be very difficult. We strengthened the results of Biro et al. [BMM10] in Chapter 3, and showed in Chapter 4 that minimizing the number of BPs is also difficult for HRLQ. These results are evidence of the computational intractability of minimizing the number of BPs in stable matching problems.

Another contribution is that we have obtained tight results for the stable matching problems. Tight result is a steady step forward in the understanding of computational tractability. In Chapter 4, we showed a polynomial-time $(|H| + |R|)$ approximation algorithm for Min-BP HRLQ and a tight lower bound $(|H| + |R|)^{1-\epsilon}$ for any positive ϵ of approximation ratio. In Chapter 6, we showed tight upper and lower bounds of approximation ratios of man-strategy-proof and woman-strategy-proof algorithms for MAX SMTI and MAX SMTI-1TM. Furthermore, in Chapter 5, we showed that there is a polynomial-time algorithm for the problem of determining the existence of weak-SSNM in SMTI if the length of each man's preference list is less than or equal to one and that this condition is tight.

Our results also provide an avenue for subsequent studies. The constraint on the lower quotas of HR defined in Chapter 4 reflects an important real-world requirement of balancing the number of residents assigned to hospitals. Together with models introduced in [BFIM10, Hua10], it has triggered a number of subsequent studies on HRLQ such as direct subsequent works [ÁBM16, FK16, Yok17, BH20, MS20], extensions to other variants [MT13, Kam13, CF17, CFP21], and works with relaxation of stability as shown below. In addition, the inapproximability results presented in Chapters 3 and 4 showed that using the number of blocking pairs or the number of blocking residents as a measure of instability is unrealistic in terms of computational complexity. It led to consider alternative notion of stability. Envy-freeness seems to be a good candidate of alternatives since it is a relaxation of stability. In envy-free matchings, we allow for the existence of a blocking pair between a vacant position in a hospital and a resident. In fact, there has been several studies on the problem of finding an envy-free matching or almost envy-free matching in HRLQ [FIT$^+$16, Yok20, HG21]. In addition to envy-freeness, problems of finding a matching with relaxation of stability called relaxed stability [KLNN20] or other notion called popularity [NN17, MNNR18] in HRLQ have also been studied. In Chapter 6, we gave a generic technique of proving strategy-proofness that rewrites the algorithm that we want to show strategy-proofness to a translation of an instance and applying existing strategy-proof algorithm. It seems useful for other proofs of strategy-proofness; since proving strategy-proofness tends to be complicated. This technique is used to construct strategy-proof algorithms for some stable matching problems [GMMY22, Yok21].

Bibliography

[ABE+09] Esther M. Arkin, Sang Won Bae, Alon Efrat, Kazuya Okamoto, Joseph S. B. Mitchell, and Valentin Polishchuk. Geometric stable roommates. *Information Processing Letters*, 109(4):219–224, 2009.

[ABM05] David J. Abraham, Péter Biró, and David F. Manlove. "Almost stable" matchings in the roommates problem. In *Proceedings of the Third International Workshop on Approximation and Online Algorithms, WAOA 2005*, volume 3879 of *Lecture Notes in Computer Science*, pages 1–14. Springer, 2005.

[ÁBM16] Kolos Csaba Ágoston, Péter Biró, and Iain McBride. Integer programming methods for special college admissions problems. *Journal of Combinatorial Optimization*, 32(4):1371–1399, 2016.

[ACG+18] 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.

[AIM07] David J. Abraham, Robert W. Irving, and David F. Manlove. Two algorithms for the student-project allocation problem. *Journal of Discrete Algorithms*, 5(1):73–90, 2007.

[AITT00] Yuichi Asahiro, Kazuo Iwama, Hisao Tamaki, and Takeshi Tokuyama. Greedily finding a dense subgraph. *Journal of Algorithms*, 34(2):203–221, 2000.

[APR05] Atila Abdulkadiroğlu, Parag A. Pathak, and Alvin E. Roth. The New York city high school match. *American Economic Review*, 95(2):364–367, May 2005.

[APR09] Atila Abdulkadiroğlu, Parag A Pathak, and Alvin E Roth. Strategy-proofness versus efficiency in matching with indifferences: Redesigning the NYC high school match. *American Economic Review*, 99(5):1954–78, 2009.

[APRS05] Atila Abdulkadiroğlu, Parag A. Pathak, Alvin E. Roth, and Tayfun Sönmez. The Boston public school match. *American Economic Review*, 95(2):368–371, May 2005.

[Ata85] Mikhail J. Atallah. A matching problem in the plane. *Journal of Computer and System Sciences*, 31(1):63–70, 1985.

[BCC+10] Aditya Bhaskara, Moses Charikar, Eden Chlamtac, Uriel Feige, and Aravindan Vijayaraghavan. Detecting high log-densities: an $O(n^{1/4})$ approximation for densest k-subgraph. In *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010*, pages 201–210. ACM, 2010.

[BFIM10] 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, Jul 2010.

[BH20] Niclas Boehmer and Klaus Heeger. A fine-grained view on stable many-to-one matching problems with lower and upper quotas. In *Web and Internet Economics - 16th International Conference, WINE 2020*, volume 12495 of *Lecture Notes in Computer Science*, pages 31–44. Springer, 2020.

[BKS03] Piotr Berman, Marek Karpinski, and Alex D. Scott. Approximation hardness of short symmetric instances of MAX-3SAT. *Electronic Colloquium on Computational Complexity*, (049), 2003.

[BMM10] Péter Biró, David Manlove, and Shubham Mittal. Size versus stability in the marriage problem. *Theoretical Computer Science*, 411(16-18):1828–1841, 2010.

[BMM12] Péter Biró, David F. Manlove, and Eric McDermid. "Almost stable" matchings in the roommates problem with bounded preference lists. *Theoretical Computer Science*, 432:10–20, 2012.

[CaR] CaRMS. Canadian resident matching service. `http://www.carms.ca/`.

[CF17] Katarína Cechlárová and Tamás Fleiner. Pareto optimal matchings with lower quotas. *Mathematical Social Sciences*, 88:3–10, 2017.

[CFP21] Ágnes Cseh, Tobias Friedrich, and Jannik Peters. Pareto optimal and popular house allocation with lower and upper quotas. *CoRR*, abs/2107.03801, 2021.

[CHSY18] Jiehua Chen, Danny Hermelin, Manuel Sorge, and Harel Yedidsion. How hard is it to satisfy (almost) all roommates? In *45th International Collo-*

quium on Automata, Languages, and Programming, ICALP 2018, volume 107 of *LIPIcs*, pages 35:1–35:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.

[CIM19] Ágnes Cseh, Robert W. Irving, and David F. Manlove. The stable roommates problem with short lists. *Theory of Computing Systems*, 63(1):128–149, 2019.

[CLW15] Danny Ziyi Chen, Xiaomin Liu, and Haitao Wang. Computing maximum non-crossing matching in convex bipartite graphs. *Discrete Applied Mathematics*, 187:50–60, 2015.

[Coo71] Stephen A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing*, pages 151–158. ACM, May 1971.

[CR21] Jiehua Chen and Sanjukta Roy. Euclidean 3D stable roommates is NP-hard. *CoRR*, abs/2108.03868, 2021.

[DF81] Lester E. Dubins and David A. Freedman. Machiavelli and the Gale-Shapley algorithm. *The American Mathematical Monthly*, 88(7):485–494, 1981.

[DG10] Shaddin Dughmi and Arpita Ghosh. Truthful assignment without money. In *Proceedings of the 11th ACM Conference on Electronic Commerce, EC 2010*, pages 325–334. ACM, 2010.

[EE08] Aytek Erdil and Haluk Ergin. What's the matter with tie-breaking? improving efficiency in school choice. *American Economic Review*, 98(3):669–89, 2008.

[EH08] Kimmo Eriksson and Olle Häggström. Instability of matchings in decentralized markets with various preference structures. *International Journal of Game Theory*, 36(3-4):409–420, 2008.

[Fei02] Uriel Feige. Relations between average case complexity and approximation complexity. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing, STOC 2002*, pages 534–543. ACM, 2002.

[FIT+16] Daniel Fragiadakis, Atsushi Iwasaki, Peter Troyan, Suguru Ueda, and Makoto Yokoo. Strategyproof matching with minimum quotas. *ACM Transactions on Economics and Computation*, 4(1):1–40, 2016.

[FK12] Tamás Fleiner and Naoyuki Kamiyama. A matroid approach to stable matchings with lower quotas. In *Proceedings of the Twenty-Third Annual*

ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, pages 135–142. SIAM, 2012.

[FK16] Tamás Fleiner and Naoyuki Kamiyama. A matroid approach to stable matchings with lower quotas. *Mathematics of Operations Research*, 41(2):734–744, 2016.

[FKP01] Uriel Feige, Guy Kortsarz, and David Peleg. The dense k-subgraph problem. *Algorithmica*, 29(3):410–421, 2001.

[Gab83] Harold N. Gabow. An efficient reduction technique for degree-constrained subgraph and bidirected network flow problems. In *Proceedings of the 15th Annual ACM Symposium on Theory of Computing*, pages 448–456. ACM, 1983.

[GI89] Dan Gusfield and Robert W. Irving. *The Stable Marriage Problem: Structure and Algorithms.* Foundations of computing series. MIT Press, 1989.

[GIK+16] Masahiro Goto, Atsushi Iwasaki, Yujiro Kawasaki, Ryoji Kurata, Yosuke Yasuda, and Makoto Yokoo. Strategyproof matching with regional minimum and maximum quotas. *Artificial Intelligence*, 235:40–57, Jun 2016.

[GJ79] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, 1979.

[GMMY22] Hiromichi Goko, Kazuhisa Makino, Shuichi Miyazaki, and Yu Yokoi. Maximally satisfying lower quotas in the hospitals/residents problem with ties. In *STACS 2022, 39th International Symposium on Theoretical Aspects of Computer Science, Proceedings*, 2022. To appear.

[GS62] David Gale and Lloyd S. Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.

[GS85] David Gale and Marilda Sotomayor. Some remarks on the stable matching problem. *Discrete Applied Mathematics*, 11(3):223–232, Jul 1985.

[HG21] Changyong Hu and Vijay K. Garg. Minimal envy matchings in the hospitals/residents problem with lower quotas. *CoRR*, abs/2110.15559, 2021.

[HIMY07] Magnús M. Halldórsson, Kazuo Iwama, Shuichi Miyazaki, and Hiroki Yanagisawa. Improved approximation results for the stable marriage problem. *ACM Transactions on Algorithms*, 3(3):30, 2007.

[Hua10] Chien-Chung Huang. Classified stable matching. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2010*, pages 1235–1253. SIAM, 2010.

[IM08] Robert W. Irving and David F. Manlove. Approximation algorithms for hard variants of the stable marriage and hospitals/residents problems. *Journal of Combinatorial Optimization*, 16(3):279–292, 2008.

[IMMM99] Kazuo Iwama, David F. Manlove, Shuichi Miyazaki, and Yasufumi Morita. Stable marriage with incomplete lists and ties. In *Proceedings of the 26th International Colloquium on Automata, Languages and Programming, ICALP 1999*, volume 1644 of *Lecture Notes in Computer Science*, pages 443–452. Springer, 1999.

[IMO09] Robert W. Irving, David F. Manlove, and Gregg O'Malley. Stable marriage with ties and bounded length preference lists. *Journal of Discrete Algorithms*, 7(2):213–219, 2009.

[IMS00] Robert W. Irving, David F. Manlove, and Sandy Scott. The hospitals/residents problem with ties. In *Proceedings of the 7th Scandinavian Workshop on Algorithm Theory, SWAT 2000*, volume 1851 of *Lecture Notes in Computer Science*, pages 259–271. Springer, 2000.

[IMS03] Robert W. Irving, David F. Manlove, and Sandy Scott. Strong stability in the hospitals/residents problem. In *Proceedings of the 20th Annual Symposium on Theoretical Aspects of Computer Science, STACS 2003*, volume 2607 of *Lecture Notes in Computer Science*, pages 439–450. Springer, 2003.

[IMS08] Robert W. Irving, David F. Manlove, and Sandy Scott. The stable marriage problem with master preference lists. *Discrete Applied Mathematics*, 156(15):2959–2977, 2008.

[Irv94] Robert W. Irving. Stable marriage and indifference. *Discrete Applied Mathematics*, 48(3):261–272, 1994.

[Kam13] Naoyuki Kamiyama. A note on the serial dictatorship with project closures. *Operations Research Letters*, 41(5):559–561, 2013.

[Kho06] Subhash Khot. Ruling out PTAS for graph min-bisection, dense k-subgraph, and bipartite clique. *SIAM Journal on Computing*, 36(4):1025–1071, 2006.

[Kir13] Zoltán Király. Linear time local approximation algorithm for maximum stable marriage. *Algorithms*, 6(3):471–484, 2013.

[KK10] Yuichiro Kamada and Fuhito Kojima. Improving efficiency in matching markets with regional caps: The case of the Japan Residency Matching Program. *Discussion Papers, Stanford Institute for Economic Policy Research*, 1, 2010.

[KLNN20] Prem Krishnaa, Girija Limaye, Meghana Nasre, and Prajakta Nimbhorkar. Envy-freeness and relaxed stability: Hardness and approximation algorithms. In *Algorithmic Game Theory - 13th International Symposium, SAGT 2020*, volume 12283 of *Lecture Notes in Computer Science*, pages 193–208. Springer, 2020.

[KMMP07] Telikepalli Kavitha, Kurt Mehlhorn, Dimitrios Michail, and Katarzyna E. Paluch. Strongly stable matchings in time $O(nm)$ and extension to the hospitals-residents problem. *ACM Transactions on Algorithms*, 3(2):15, 2007.

[KMRZ19] Piotr Krysta, David F. Manlove, Baharak Rastegari, and Jinshan Zhang. Size versus truthfulness in the house allocation problem. *Algorithmica*, 81(9):3422–3463, 2019.

[KMV94] Samir Khuller, Stephen G. Mitchell, and Vijay V. Vazirani. On-line algorithms for weighted bipartite matching and stable marriages. *Theoretical Computer Science*, 127(2):255–267, 1994.

[Knu76] Donald E. Knuth. *Mariages stables.* Les Presses de L'Université de Montréal, 1976.

[KT86] Yoji Kajitani and Toshihiko Takahashi. The noncross matching and applications to the 3-side switch box routing in VLSI layout design. In *Proceedings of International Symposium on Circuits and Systems*, pages 776–779, 1986.

[Lic15] Jared D. Lichtman. On the multidimensional stable marriage problem. *CoRR*, abs/1509.02972, 2015.

[Lim21] Girija Limaye. Envy-freeness and relaxed stability for lower-quotas: A parameterized perspective. *CoRR*, abs/2106.09917, 2021.

[LP19] Chi-Kit Lam and C. Gregory Plaxton. A (1 + 1/e)-approximation algorithm for maximum stable matching with one-sided ties and incomplete lists. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019*, pages 2823–2840. SIAM, 2019.

[Man13] David F. Manlove. *Algorithmics of Matching Under Preferences*, volume 2 of *Series on Theoretical Computer Science.* WorldScientific, 2013.

[McD09] Eric McDermid. A 3/2-approximation algorithm for general stable marriage. In *Proceedings of the 36th International Colloquium on Automata, Languages and Programming, ICALP 2009*, volume 5555 of *Lecture Notes in Computer Science*, pages 689–700. Springer, 2009.

[MII+02] David F. Manlove, Robert W. Irving, Kazuo Iwama, Shuichi Miyazaki, and Yasufumi Morita. Hard variants of stable marriage. *Theoretical Computer Science*, 276(1-2):261–279, 2002.

[MNNR18] Krishnapriya A M, Meghana Nasre, Prajakta Nimbhorkar, and Amit Rawat. How good are popular matchings? In *17th International Symposium on Experimental Algorithms, SEA 2018*, volume 103 of *LIPIcs*, pages 9:1–9:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.

[MO19] Shuichi Miyazaki and Kazuya Okamoto. Jointly stable matchings. *Journal of Combinatorial Optimization*, 38(2):646–665, 2019.

[MOP93] Federico Malucelli, Thomas Ottmann, and Daniele Pretolani. Efficient labelling algorithms for the maximum noncrossing matching problem. *Discrete Applied Mathematics*, 47(2):175–179, 1993.

[MS20] Matthias Mnich and Ildikó Schlotter. Stable matchings with covering constraints: A complete computational trichotomy. *Algorithmica*, 82(5):1136–1188, 2020.

[MT13] Daniel Monte and Norovsambuu Tumennasan. Matching with quorums. *Economics Letters*, 120(1):14–17, 2013.

[NN17] 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*, volume 93 of *LIPIcs*, pages 44:1–44:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

[NR04] Muriel Niederle and Alvin E Roth. Market culture: How norms governing exploding offers affect market performance, 2004.

[O'M07] Gregg O'Malley. *Algorithmic aspects of stable matching problems.* PhD thesis, University of Glasgow, 2007.

[Pal14] Katarzyna E. Paluch. Faster and simpler approximation of stable matchings. *Algorithms*, 7(2):189–202, 2014.

[RI19] Suthee Ruangwises and Toshiya Itoh. Stable noncrossing matchings. In *Proceedings of the 30th International Workshop on Combinatorial Algorithms, IWOCA 2019*, volume 11638 of *Lecture Notes in Computer*

Science, pages 405–416. Springer, 2019.

[Rot82] Alvin E. Roth. The economics of matching: Stability and incentives. *Mathematics of Operations Research*, 7(4):617–628, 1982.

[Rot84] Alvin E. Roth. The evolution of the labor market for medical interns and residents: A case study in game theory. *Journal of Political Economy*, 92(6):991–1016, Dec 1984.

[Rot86] Alvin E. Roth. On the allocation of residents to rural hospitals: A general property of two-sided matching markets. *Econometrica*, 54(2):425–427, Mar 1986.

[Rot02] Alvin E Roth. The economist as engineer: Game theory, experimentation, and computation as tools for design economics. *Econometrica*, 70(4):1341–1378, 2002.

[RP99] Alvin E Roth and Elliott Peranson. The redesign of the matching market for american physicians: Some engineering aspects of economic design. *American Economic Review*, 89(4):748–780, 1999.

[RS90] Alvin E. Roth and Marilda A. Oliveira Sotomayor. *Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis*. Econometric Society Monographs. Cambridge University Press, 1990.

[RSÜ04] Alvin E Roth, Tayfun Sönmez, and M Utku Ünver. Kidney exchange. *The Quarterly Journal of Economics*, 119(2):457–488, 2004.

[RX97] Alvin E Roth and Xiaolin Xing. Turnaround time and bottlenecks in market clearing: Decentralized matching in the market for clinical psychologists. *Journal of Political Economy*, 105(2):284–329, 1997.

[Sot99] Marilda Sotomayor. Three remarks on the many-to-many stable matching problem. *Mathematical social sciences*, 38(1):55–70, 1999.

[TST01] Chung-Piaw Teo, Jay Sethuraman, and Wee-Peng Tan. Gale-Shapley stable marriage problem revisited: Strategic issues and applications. *Management Science*, 47(9):1252–1267, 2001.

[Vin07] Staal A. Vinterbo. A stab at approximating minimum subadditive join. In *Proceedings of the 10th International Workshop on Algorithms and Data Structures, WADS 2007*, volume 4619 of *Lecture Notes in Computer Science*, pages 214–225. Springer, 2007.

[WW85] Peter Widmayer and Chak-Kuen Wong. An optimal algorithm for the maximum alignment of terminals. *Information Processing Letters*, 20(2):75–82, 1985.

[Yan07] Hiroki Yanagisawa. *Approximation algorithms for stable marriage problems.* PhD thesis, Kyoto University, 2007.

[Yok17] Yu Yokoi. A generalized polymatroid approach to stable matchings with lower quotas. *Mathematics of Operations Research*, 42(1):238–255, 2017.

[Yok20] Yu Yokoi. Envy-free matchings with lower quotas. *Algorithmica*, 82(2):188–211, 2020.

[Yok21] Yu Yokoi. An approximation algorithm for maximum stable matching with ties and constraints. *CoRR*, abs/2107.03076, 2021.

Publication List

1. Koki Hamada, Kazuo Iwama, and Shuichi Miyazaki. An improved approximation lower bound for finding almost stable maximum matchings. *Information Processing Letters*, 109(18):1036–1040, 2009, Elsevier.
   `http://dx.doi.org/10.1016/j.ipl.2009.06.008`
2. Koki Hamada, Kazuo Iwama, and Shuichi Miyazaki. The hospitals/residents problem with lower quotas. *Algorithmica*, 74(1):440–465, 2016, Springer Nature.
   `http://dx.doi.org/10.1007/s00453-014-9951-z`
3. Koki Hamada, Shuichi Miyazaki, and Hiroki Yanagisawa. Strategy-proof approximation algorithms for the stable marriage problem with ties and incomplete lists. In *Proceedings of the 30th International Symposium on Algorithms and Computation, ISAAC 2019,* volume 149 of *Leibniz International Proceedings in Informatics*, pages 9:1–9:14, 2019, Schloss Dagstuhl - Leibniz Center for Informatics.
   `http://dx.doi.org/10.4230/LIPIcs.ISAAC.2019.9`
4. Koki Hamada, Shuichi Miyazaki, and Kazuya Okamoto. Strongly stable and maximum weakly stable noncrossing matchings. *Algorithmica*, 83(9):2678–2696, 2021, Springer Nature.
   `http://dx.doi.org/10.1007/s00453-021-00832-9`

- Chapter 3 is based on paper 1, which Elsevier allows authors to include in a thesis.
- Chapter 4 is based on paper 2. Reproduced with permission from Springer Nature.
- Chapter 5 is based on paper 4. Reproduced with permission from Springer Nature.
- Chapter 6 is based on paper 3, for which we retain the copyright.