A 1.875–Approximation Algorithm for the Stable Marriage Problem*

Kazuo Iwama[†] Shuichi Miyazaki[‡] Naoya Yamauchi[§]

Abstract

We consider the problem of finding a stable matching of maximum size when both ties and unacceptable partners are allowed in preference lists. This problem is known to be APX-hard, and the current best known approximation algorithm achieves the approximation ratio $2 - c\frac{1}{\sqrt{N}}$, where c is some positive constant. In this paper, we give a 1.875–approximation algorithm, which is the first result on the approximation ratio better than two.

1 Introduction

An instance of the stable marriage problem consists of N men, N women, and each person's preference list. A preference list is a totally ordered list including all members of the opposite sex depending on his/her preference. For a matching M between men and women, a pair of a man m and a woman w is called a *blocking pair* if both prefer each other to their current partners. A matching with no blocking pair is called *stable*. Gale and Shapley showed that every instance admits at least one stable matching, and proposed a polynomial-time algorithm to find one, which is known as the Gale-Shapley algorithm [9]. There are several examples of using the stable marriage problem in assignment systems. Probably, one of the best known applications is to assign medical students to hospitals based on preference lists of both sides, which is known as NRMP in the U.S. [11, 28], CaRMS in Canada [6], SPA in Scotland [17, 18], and JRMP in Japan [24]. Another application is to assign students to schools in Singapore [32].

Considering such applications, it is unrealistic to require each participant to submit a preference list *strictly* ordering *all* members of the opposite side. One natural extension is to allow an *incomplete list*; in other words, one may drop persons from the list whom he/she does not want to be matched with. In this case, a stable matching may not be a perfect matching, but all stable matchings for a fixed instance are of the same size [10], and a slight modification of the Gale-Shapley algorithm can find one in polynomial time. Another natural extension is to allow *ties* in the list [11, 16]. Again, it is easy to find a stable matching by a modified Gale-Shapley algorithm that breaks ties arbitrarily.

However, if we allow *both* extensions, one instance can admit stable matchings of different sizes, and the problem of finding a largest stable matching, which we call *MAX SMTI* (MAXimum Stable Marriage with Ties and Incomplete lists), is NP-hard [21, 29]. It is known that any stable matching has size at least a half the largest size, and there is a polynomial-time algorithm that finds one if we do not care the size. Hence, a 2-approximation algorithm is trivial. However, it has been open if there is an approximation algorithm with the ratio strictly better than two. The best known bound was $2 - c\frac{1}{\sqrt{N}}$, where c is an arbitrary constant that satisfies $c \leq \frac{1}{4\sqrt{6}}$ [23].

Our Contribution. In this paper, we give a 1.875–approximation algorithm, the first result giving the approximation ratio strictly better than two. Our algorithm is based on a local search similar to the previous papers [22, 23]. It uses two subroutines, Increase and Stabilize. For a given stable matching M, Increase outputs another matching M' with a desirable property such that $|M'| > |M|$ but M' may not be stable. Stabilize changes this M' into M'' which is stable without decreasing the size. In [23], it was proved that Increase successfully works if $|M| < \frac{OPT}{2} + c\sqrt{|M|}$ where OPT denotes the size of a largest stable matching, which can only guarantee an approximation factor of $2 - o(1)$. In this paper, we improve Increase so that it will work if $|M| < \frac{OPT}{2} + \frac{OPT}{30}$, which then implies an approximation factor of $2 - \frac{1}{8}$.

The basic idea of the improvement is as follows: Suppose that we are now given a matching M and a set U of men who are single in M. Then we want to increase the size of M by finding a "good" partner for one of U. Here, "good" means that such a new matching does not create new blocking pairs of prohibited type. In the previous papers, such a partner was found by a sort of exhaustive method. In this paper, we consider possible prohibited blocking pairs created by the new matching

*Supported in part by Scientific Research Grant, Ministry of Education, Japan, 16300002, 16092101, 16092215, 15700010, 17700015.

[†]Graduate School of Informatics, Kyoto University.

[‡]Academic Center for Computing and Media Studies, Kyoto University.

[§]Graduate School of Informatics, Kyoto University.

in advance, and kills such possibilities by modifying the current M also *in advance*. We can thus guarantee that the new matching does not create new prohibited blocking pairs.

Related Results. As mentioned above, MAX SMTI was proved to be NP-hard [21] and then to be APX-hard [12]. Subsequently, it was shown that MAX SMTI cannot be approximated within 21/19 unless P=NP [14]. For restricted inputs, there are a couple of approximation algorithms with factors better than two [13, 14]. Other than SMTI, research on the problems of finding matchings based on preference lists has been quite intensive recently, which includes variants of stable matchings [7, 3, 8, 20, 27], rank-maximal matchings [19], Pareto optimal matchings [1], and popular matchings [2].

There are several optimization problems that resemble MAX SMTI, where designing a 2-approximation algorithm is trivial but obtaining a $(2-\epsilon)$-approximation algorithm for a positive constant ϵ appears to be extremely hard, such as Minimum Vertex Cover (MIN VC for short) and Minimum Maximal Matching (MIN MM for short). As is the case with MAX SMTI, there are a lot of approximability results for restricted versions of these problems. For example, MIN VC is approximable within 7/6 if the maximum degree of an input graph is bounded by 3 [5], or within $2/(1+\epsilon)$ if every vertex has degree at least $\epsilon|V|$ [26]. For MIN MM, there is a $(2-1/d)$-approximation algorithm for regular graphs with degree d [33], and a PTAS for planar graphs [31]. For general inputs, $(2-o(1))$-approximation algorithms are presented for MIN VC, namely, $2-\frac{\log\log|V|}{2\log|V|}$ and $2-(1-o(1))\frac{2\ln\ln|V|}{\ln|V|}$ [30, 4, 15]. Very recently, this has been improved to $2-\Theta(\frac{1}{\sqrt{\log|V|}})$ [25].

2 Preliminaries

2.1 Notations and Definitions

In this section, we formally define MAX SMTI and approximation ratios. An instance I of MAX SMTI comprises N men, N women and each person's preference list that may be incomplete and may include ties. Let Y and X denote the set of men and women, respectively. We usually use m, m_i, etc for men, and w, w_i, etc for women. If a man m includes a woman w in his list, we say that w is *acceptable* to m. If m strictly prefers w_i to w_j in I, we write $w_i \succ_m w_j$. If w_i and w_j are tied in m's list (including the case that $w_i = w_j$), we write $w_i =_m w_j$. The statement $w_i \succeq_m w_j$ is true if and only if $w_i \succ_m w_j$ or $w_i =_m w_j$. We use similar notations for women's preference lists.

A matching M is a set of pairs (m,w) such that m is acceptable to w and vice versa, and each person appears at most once in M. For a matching M, let M^Y and M^X denote the set of men and women in M, respectively. If $m \notin M^Y$ ($w \notin M^X$, respectively), we say that m (w, respectively) is *single* in M. We sometimes call a pair $(m,w) \in M$ an *edge* of M.

If a man m is matched with a woman w in M, we write $M(m) = w$ and $M(w) = m$. We say that m and w form a *blocking pair* for M (or simply, (m,w) *blocks* M) if the following three conditions are met: (i) $M(m) \neq w$ but m and w are acceptable to each other. (ii) $w \succ_m M(m)$ or m is single in M. (iii) $m \succ_w M(w)$ or w is single in M. A matching M is called *stable* if there is no blocking pair for M. MAX SMTI is the problem of finding a largest stable matching.

A goodness measure of an approximation algorithm T for a maximization problem is defined in the usual way: the *approximation ratio* of T is $\max\{opt(x)/T(x)\}$ over all instances x of size N, where $opt(x)$ and $T(x)$ are the size of the optimal and the algorithm's solution, respectively. Throughout this paper, we denote the maximum size by OPT.

2.2 The Gale-Shapley Algorithm

Since we use the Gale-Shapley algorithm (*GS algorithm* for short) as a subroutine, let us briefly explain it (see e.g. [11] for the full description). At the beginning of the algorithm, all men and women are free. At each iteration, an arbitrary free man (say, m) makes a proposal to the best woman (say, w) among those he has not yet made a proposal. If w is free, w instantly receives a proposal, and m and w become newly matched. If w is currently matched, then she compares m and her current partner, and takes the better one. The man who is refused by w becomes (or remains) free. When there is no free man who has a woman to make a proposal, then the GS algorithm terminates. We can prove that the resulting matching is stable.

Of course, we can consider the same algorithm with the role of men and women exchanged, so that women make and men receive proposals. This may produce a different stable matching, so when considering the GS algorithm, we have to specify which gender makes proposals. In this paper, we use the former variant, which we call *men-propose GS algorithm*. It should be noted that, in the GS algorithm, once a woman receives a proposal, she never becomes single again, so that she is matched at the end of the algorithm. This property is used in the analysis of the approximation ratio.

2.3 Algorithm LocalSearch(I)

In this section, we review the algorithm LocalSearch presented in

[22, 23]. It takes a MAX SMTI instance I and outputs a stable matching for I. The algorithm first uses the GS algorithm described above. Then it goes to the while loop consisting of two subroutines, Increase and Stabilize. Increase takes a stable matching M for I and outputs a (not necessarily stable) matching M' such that $|M'| > |M|$ and M' satisfies the following property Π:

> Π: For any blocking pair (m, w) for M', either m or w is single in M'.

Increase may fail to find such a matching. Stabilize takes a matching M' with property Π, and outputs a stable matching of size at least $|M'|$. Stabilize never fails. (Stabilize will be explained in Sec. 2.4.) The complete description of LocalSearch is as follows:

Algorithm LocalSearch(I)

```
M = arbitrary stable matching for I.
    /* This can be done in polynomial time by
       arbitrary tie-breaking and applying the
       GS algorithm. */
while (true)
    { M' = Increase (M).
        If Increase fails, exit while loop and
        output M.
      M = Stabilize (M'). }
```

It was shown in [23] that while $|M| < \frac{OPT}{2} + c\sqrt{|M|}$, Increase never fails, and hence we finally obtain a stable matching of size at least $\frac{OPT}{2} + c\sqrt{|M|}$. This gave an upper bound on the approximation ratio of LocalSearch.

2.4 Procedure Stabilize **(M)** Recall that Stabilize takes a matching M having the property Π and makes it stable without decreasing the size. Here we give the full description of Stabilize. For a matching M, define $BP_{s,m}(M)$ (s means single and m matched) to be the set of blocking pairs (m, w) for M such that m is single and w is matched in M. Similarly, define $BP_{-,s}(M)$ to be the set of blocking pairs (m, w) for M such that w is single in M (m may be single or matched).

Procedure Stabilize(M)

```
1: while ( BP_{s,m}(M) ≠ ∅ )
2:   {Select (m, w) ∈ BP_{s,m}(M).
3:     w* = woman such that (m, w*) ∈ BP_{s,m}(M)
         and there is no (m, w') ∈ BP_{s,m}(M) such
         that w' ≻_m w*.
4:     M = M − {(M(w*), w*)} ∪ {(m, w*)}. }
5: while ( BP_{−,s}(M) ≠ ∅ )
6:   {Select (m, w) ∈ BP_{−,s}(M).
7:     m* = man such that (m*, w) ∈ BP_{−,s}(M)
         and there is no (m', w) ∈ BP_{−,s}(M) such
         that m' ≻_w m*.
8:     if ( m* is matched in M )
9:         M = M − {(m*, M(m*))} ∪ {(m*, w)}.
10:    else
11:        M = M ∪ {(m*, w)}. }
```

We will give an idea of the correctness proof. During the first while loop, Stabilize resolves blocking pairs in $BP_{s,m}(M)$. Consider an execution of the 4th line. It is easy to see that this operation does not decrease the matching size. Observe the following:

(1) For women's side, only w^* changes the status. She gets a better partner (since (m, w^*) was a blocking pair).

(2) For men's side, $M(w^*)$ becomes single from matched, m becomes matched from single, and other men do not change the status.

We can prove that the new matching does not break the condition Π. By (2), if there arises a new blocking pair, then it must include the man $M(w^*)$ because only he gets worse off. But $M(w^*)$ is single in the new matching, and hence does not break Π. Another possibility is that (m, w) was a blocking pair in M such that m was single and w was matched. In the new matching, m gets matched and (m, w) remains a blocking pair, resulting that (m, w) is a blocking pair such that both m and w are matched. However, this does not happen since m selects w^*, one of the best women among all women w such that (m, w) is a blocking pair and w is matched.

By (1), we can see that Stabilize exits the while loop within N^2 iterations, and by the condition of exiting the loop, all blocking pairs in $BP_{s,m}(M)$ are resolved.

The second while loop resolves blocking pairs in $BP_{-,s}(M)$. Similarly, we can show that Stabilize exits this loop within N^2 iterations, and this loop does not create prohibited blocking pairs. (This time, "prohibited blocking pairs" include ones in $BP_{s,m}(M)$, which we have already resolved.) Then, clearly the output matching is stable since all blocking pairs are resolved.

3 Improved Algorithm

Now we give an improved algorithm Increase. Figs. 1, 2 and 3 show an example of the execution of Increase. (See below for details.) In the next section, we prove that Increase never fails if $|M| < \frac{8}{15}OPT$. (Recall

INCREASE takes a stable matching M as its input.) Hence, we can obtain a stable matching of size at least $\frac{8}{15}OPT$ by LOCALSEARCH.

Procedure INCREASE(M)

/* See Sec. 2.1 for the definition of M^Y, M^X, etc. */

1: Construct a bipartite graph $G_1 = (U_1, V_1, E_1)$, where $U_1 = M^X$, $V_1 = Y - M^Y$, and E_1 includes an edge between $w \in U_1$ and $m \in V_1$ if and only if $M(w) =_w m$.

2: Find a maximum matching in G_1, and let it be P.

3: Let M' be the subset of M such that $w \in M'^X$ iff $w \in P^X$.

4: Let $M_1 = M - M' \cup P$.

5: Construct a bipartite graph $G_2 = (U_2, V_2, E_2)$, where $U_2 = M'^Y$, $V_2 = M_1^X$, and E_2 is defined as follows:
  $F = X - M_1^X$.
  For each $m \in U_2$
    { Let $M_m = M_1 \cup \{(m, w^*)\}$, where w^* is the woman whom m prefers best within F. (Select an arbitrarily woman if there are ties. M_m is undefined if there is no such w^*.)
    Add an edge (m, w) to E_2 iff M_m is defined and (m, w) blocks M_m. }

6: Apply men-propose GS algorithm to G_2, and let Q be the resulting matching. (Before applying GS algorithm, for each man m and woman w, delete m from w's list, and w from m's list if $(m, w) \notin E_2$. Also, for each person's preference list, break ties arbitrarily.)

7: Let M'' be the subset of M_1 such that $w \in M''^X$ iff $w \in Q^X$.

8: Let $M_2 = M_1 - M'' \cup Q$.

9: For each m in $M'^Y - M_2^Y$, do the following:
  $\{M^* = M_2 \cup \{(m, w^*)\}$, where w^* is the woman whom m prefers best in F. (Select an arbitrary woman if there are ties. M^* is undefined if there is no such w^*.)
  If M^* satisfies Π, then output M^* and stop. }

10 ∼ 18: Do the same operation as lines 1 through 9 by exchanging the role of men and women.

19: Output failure, and stop.

Here, let us briefly describe an execution of the first half (lines 1 through 9) of INCREASE using Figs. 1 to 3. Vertices corresponding to men are denoted by black ones, while those corresponding to women white ones. Fig. 1 (1) shows an input matching M. Fig. 2 shows how to construct G_1, where "$w_2 : m_3\ (m_2\ m'_3\ m'_4)\ m'_5$" shows the preference list of w_2, meaning that w_2 likes m_3 best, then m_2, m'_3 and m'_4 with equal preference, and then m'_5. Dotted lines are then added to E_1. At line 2, INCREASE finds a matching P, and P defines the matching M' (Fig. 1 (2)). At line 4, matching M_1 is constructed from M by removing M' and adding P (Fig. 1 (3)).

Using M_1, INCREASE constructs the bipartite graph G_2 at line 5. Fig. 3 (1) shows U_2, V_2 and F. When determining the edge set E_2 of G_2, each member of U_2 is provisionally matched to a best woman in F. In Fig. 3 (1), m is provisionally matched to w^*, and as a result, m creates two blocking pairs between women in V_2, denoted by dotted edges. These two edges are added to E_2 (Fig. 3 (2)). G_2 is the result of doing the same operation to all men in U_2 (Fig. 3 (3)). At line 6, INCREASE finds a matching Q by applying GS algorithm to G_2 (Fig. 3 (4)).

INCREASE then adds Q to M_1 (Fig. 1 (4)), and removes old edges if a woman is matched with two men. The resulting matching is M_2 (Fig. 1 (5)). Note that M_2 is of the same size as the input matching M. At line 9, INCREASE creates a matching by adding one edge (m, w^*) to M_2 for each single man $m \in U_2$ (Fig. 1 (6)), and outputs it if it satisfies Π. If this happens, then we are successful since the output matching size is greater than the input matching size by one.

4 Analysis of the Approximation Ratio

It is not hard to see that INCREASE runs in time polynomial in N. As mentioned above, if INCREASE outputs M^* at lines 9 or 18, then INCREASE succeeds. In the following, we show that this is the case if $|M| < \frac{8}{15}OPT$, namely, either there is at least one man m^* in $M'^Y - M_2^Y$ at line 9 such that $M_2 \cup \{(m^*, w^*)\}$ satisfies Π (where w^* is the woman selected for m^* at line 9), or there is at least one similarly desirable woman at line 18. In the rest of this paper, we always assume that $|M| < \frac{8}{15}OPT$ even if it is not explicitly stated.

Now we start analysis. First, let us fix an arbitrary optimal solution M_{opt} (which is of course unknown to the algorithm), a largest stable matching for the given instance I, and let M be an input for INCREASE, a stable matching for I. Let us define the following bipartite graph $G_{M_{opt},M}$: Each vertex of $G_{M_{opt},M}$ corresponds to a person in I. There is an edge between vertices m and w if and only if $M_{opt}(m) = w$ or $M(m) = w$. If both $M_{opt}(m) = w$ and $M(m) = w$ hold, we include two edges between m and w; hence $G_{M_{opt},M}$ is a multigraph. An edge (m, w) associated with $M_{opt}(m) = w$ is called an *M_{opt}-edge*. Similarly, an edge associated with $M(m) = w$ is called an *M-edge*. See Fig. 4 for an example, where M_{opt}-edges are bold and M-edges are light. Observe that the degree of each vertex is at most two, and hence each connected component of $G_{M_{opt},M}$

is a simple path, a cycle, or an isolated vertex.

We now divide the M-edges of $G_{M_{opt},M}$ into two categories, good edges and bad edges. If an edge of M is in a component that is a path of length three starting from and ending with M_{opt}-edges, then it is called M_{opt}-*good* or simply *good* if M_{opt} is clear. (See Z-shaped components of Fig. 4.) Otherwise, it is M_{opt}-*bad* or simply *bad*. Intuitively speaking, if $|M| \approx \frac{OPT}{2}$, then there must be a lot of good edges; our algorithm attacks each good edge to replace it by two matching edges. The following lemma, proved in [22], gives an important property of good edges:

Lemma 4.1. *[22] If (m, w) is a good edge of M, then (i) $w \succeq_m M_{opt}(m)$, (ii) $m \succeq_w M_{opt}(w)$, and (iii) either $w =_m M_{opt}(m)$ or $m =_w M_{opt}(w)$.*

Proof. (i) If $M_{opt}(m) \succ_m w$, then $(m, M_{opt}(m))$ blocks M, which contradicts the stability of M. So, $w \succeq_m M_{opt}(m)$. (ii) For the same reason, $m \succeq_w M_{opt}(w)$. (iii) If both $w \succ_m M_{opt}(m)$ and $m \succ_w M_{opt}(w)$ hold, then (m, w) blocks M_{opt}, which contradicts the stability of M_{opt}.

Let (m, w) be a good edge. Then, by Lemma 4.1 (iii), either $w =_m M_{opt}(m)$ or $m =_w M_{opt}(w)$. Without loss of generality, we assume that at least half of good edges of M satisfy $m =_w M_{opt}(w)$, and prove that Increase outputs a desirable matching at line 9. (Otherwise, it outputs a desirable matching at line 18, the proof of which is similar and is hence omitted in this paper.) We first show, in the next lemma, that there must be at least one man $\bar{m}$ in $M'^Y - M_2^Y$ which has a specific property and later in Lemma 4.3, we show that this man is exactly what we want as m^* mentioned above.

Lemma 4.2. *Assume that $|M| < \frac{8}{15}OPT$. Then at line 9, $M'^Y - M_2^Y$ is not empty and contains at least one $\bar{m}$ such that $(\bar{m}, M(\bar{m}))$ is a good edge of M.*

Proof. This is a key lemma and the proof goes like this: Since $(\bar{m}, M(\bar{m}))$ must be good, we first consider the set $U_2'(\subseteq U_2)$ which contains all m such that $(m, M(m))$ is good. Claim 4.1 bounds the size of U_2' from below. We then evaluate the size of $N(U_2') \subseteq V_2$, the set of vertices (women) adjacent to at least one vertex in U_2', in G_2. Observe that in Q, men in U_2' can be matched with only women in $N(U_2')$. So, if we show $|U_2'| > |N(U_2')|$, then there is at least one man in U_2' who remains single in Q, and hence single in M_2. This is the man that we want to show existence. So, we are done if we can prove $|U_2'| > |N(U_2')|$. To do so, in Claims 4.2 and 4.3, we verify the conditions that $w \in V_2$ is not adjacent to U_2', which allows us to bound $|N(U_2')|$ from above and to make the above mentioned goal. Let b be the number of bad edges of M.

Claim 4.1. $|U_2'| \geq \frac{|M|-3b}{2}$.

Proof. First, we show that $|P| \geq \frac{|M|-b}{2}$. Since the number of bad edges of M is b, the number of good edges of M is $|M| - b$. Recall that we have assumed that at least half of these good edges (m, w) satisfy $m =_w M_{opt}(w)$. Hence, this number of such w have an edge to $M_{opt}(w)$ in the bipartite graph G_1 constructed at line 1 of Increase. Observe that $M_{opt}(w) \neq M_{opt}(w')$ if $w \neq w'$. So a maximum matching in G_1 is of size at least $\frac{|M|-b}{2}$, which implies $|P| \geq \frac{|M|-b}{2}$.

Observe that $|U_2'|$ is equal to the number of good edges of M included in M' by the definition of U_2'. Also, observe that $|M'| = |P|$, implying that $|M'| \geq \frac{|M|-b}{2}$. Since there are b bad edges, M' includes at least $|M'| - b \geq \frac{|M|-3b}{2}$ good edges of M, which completes the proof.

Claim 4.2. *If $(M(w), w)$ is a good edge of M, then $w \notin N(U_2')$.*

Proof. Suppose for contradiction that $w \in N(U_2')$. Then, by the definition of $N(U_2')$, there are a man $m \in U_2'$ and an edge $(m, w) \in E_2$. Consider the matching M_m created in constructing G_2, which is the result of adding a pair (m, w^*) to M_1. Since $m \in U_2'$, $(m, M(m))$ is a good edge of M by definition. Then, $M_{opt}(m)$ is single in M by the definition of good edges, and is also single in M_1 by the construction of M_1, which means that $M_{opt}(m)$ is in F. So, $w^* \succeq_m M_{opt}(m)$ since w^* is one of the best women in F for m. But since $(m, w) \in E_2$, $w \succ_m w^*$. Hence, $w \succ_m M_{opt}(m)$.

Since $(m, w) \in E_2$, $m \succ_w M_1(w)$, and by the construction of M_1, $M_1(w) =_w M(w)$. Now, since $(M(w), w)$ is a good edge of M, $M(w) \succeq_w M_{opt}(w)$ by Lemma 4.1 (ii), so that $m \succ_w M_{opt}(w)$. Then, (m, w) blocks M_{opt}, contradicting the stability of M_{opt}.

Claim 4.3. *Consider the bipartite graph $G_{M_{opt},M}$ (see the top of this section for the definition). Consider its connected component which is a path of length at least five, and starting from and ending with M_{opt}-edges. Let its vertices be $m_1, w_1, m_2, w_2, \cdots, m_k, w_k$ in this order along the path. Then $w_1 \notin N(U_2')$.*

Proof. Suppose that $w_1 \in N(U_2')$. Then, by the definition of $N(U_2')$, there are a man $m \in U_2'$ and an edge $(m, w_1) \in E_2$. By the same argument as the proof of Claim 4.2, we can conclude that $w_1 \succ_m M_{opt}(m)$.

Since m_1 and w_1 are matched in M_{opt}, m_1 includes w_1 in his list. If $m_1 \succ_{w_1} m_2$, then (m_1, w_1)

blocks M, contradicting the stability of M. So, $m_2(= M(w_1)) \succeq_{w_1} m_1(= M_{opt}(w_1))$. By the construction of M_1, $M_1(w_1) =_{w_1} M(w_1)$. Hence, $M_1(w_1) \succeq_{w_1} M_{opt}(w_1)$. Finally, since $(m, w_1) \in E_2$, $m \succ_{w_1} M_1(w_1)$, resulting that $m \succ_{w_1} M_{opt}(w_1)$. Then, (m, w_1) blocks M_{opt}, a contradiction.

Now, we continue to prove Lemma 4.2. Recall that b is the number of bad edges of M. We classify bad edges of M according to the type of the connected component they belong to in $G_{M_{opt},M}$. Let x be the number of edges of M that belong to, in $G_{M_{opt},M}$, a cycle, a path of even length, or a path starting from and ending with M-edges. For a positive integer $i(\geq 2)$, let y_i be the number of edges in M that belong to a path of length $2i+1$ starting from and ending with M_{opt}-edges (so, this path contains i M-edges). Then, $b = x + y_2 + y_3 + \cdots$. Let g be the number of good edges of M. Then, $|M| = g + b = g + x + y_2 + y_3 + \cdots$. The following inequality is easy to verify from the definition: $OPT \leq 2g + x + \frac{3}{2}y_2 + \frac{4}{3}y_3 + \cdots$. Using this inequality,

$$g \geq \frac{1}{2}(OPT - x - \frac{3}{2}y_2 - \frac{4}{3}y_3 - \cdots). \tag{4.1}$$

By our assumption, $|M| < \frac{8}{15}OPT$. Using this and the above (4.1), we have

$$\begin{aligned} g &> \frac{1}{2}(\frac{15}{8}|M| - x - \frac{3}{2}y_2 - \frac{4}{3}y_3 - \cdots) \\ &= \frac{1}{2}(\frac{15}{8}(g + x + y_2 + y_3 + \cdots) \\ &\quad -x - \frac{3}{2}y_2 - \frac{4}{3}y_3 - \cdots). \end{aligned}$$

From this inequality, we have that

$$g > 7x + \sum_{k=2} \frac{7k-8}{k} y_k. \tag{4.2}$$

Using Claim 4.1 and the above (4.2),

$$\begin{aligned} |U_2'| &\geq \frac{|M| - 3b}{2} = \frac{g}{2} - x - \sum_{k=2} y_k \\ &> \frac{5}{2}x + \sum_{k=2} \frac{5k-8}{2k} y_k. \end{aligned} \tag{4.3}$$

We then estimate the size of $N(U_2')$. Consider a woman w in $N(U_2')$. Since w is matched in M_1, she is matched in M (note that $M^X = M_1^X$), so there is a pair $(M(w), w) \in M$. By Claim 4.2, this is not a good edge of M. Now, suppose that the connected component of $G_{M_{opt},M}$ that $(M(w), w)$ belongs to is a path $(m_1, w_1, m_2, w_2, \cdots, m_{k+1}, w_{k+1})$ of length $2k+1$ $(k \geq 2)$ starting from and ending with M_{opt}-edges. Then, by Claim 4.3, $w \neq w_1$, namely, $(M(w), w)$ is not $(m_2, w_1) \in M$. So, among k edges of M in this path, at most $k-1$ edges can be such $(M(w), w)$. Hence,

$$|N(U_2')| \leq x + \sum_{k=2} \frac{k-1}{k} y_k. \tag{4.4}$$

Since $\frac{5}{2}x + \sum_{k=2} \frac{5k-8}{2k} y_k = x + \sum_{k=2} \frac{k-1}{k} y_k + \frac{3}{2}x + \sum_{k=2} \frac{3k-6}{2k} y_k$, it is not hard to see that for any positive integers x, y_k and $k \geq 2$, $\frac{5}{2}x + \sum_{k=2} \frac{5k-8}{2k} y_k \geq x + \sum_{k=2} \frac{k-1}{k} y_k$. Thus, from (4.3) and (4.4), we have that $|U_2'| > |N(U_2')|$. Recall that this is what we wanted to show.

Let m^* be the man whose existence is guaranteed by Lemma 4.2. We show that the matching $M_2 \cup \{(m^*, w^*)\}$ constructed at line 9 satisfies Π, namely, Increase performs a successful computation. This will be proved in Lemma 4.3, but to help it, we first prove Claim 4.4.

Claim 4.4. *The matching M_2 constructed at line 8 of* Increase *satisfies* Π *(Π is originally for the matching M', but here we use it for M_2).*

Proof. Partition edges of M_2 into three types: (1) $(m, w) \in M \cap M_2$, (2) $(m, w) \in P \cap M_2$, and (3) $(m, w) \in Q \cap M_2$. For $\ell = 1, 2$ and 3, let Y_ℓ and X_ℓ be the sets of men and women, respectively, matched in M_2 by an edge of type (ℓ).

Assume that M_2 does not satisfy Π. Then, there are edges (m, w) and (m', w') in M_2 such that (m, w') blocks M_2. We consider four cases according to which sets m and w' belong to.

Case (i): $m \in Y_1 \cup Y_2$ and $w' \in X_1 \cup X_2$. If $m \in Y_1$, m is matched with the same partner in M and M_2. If $m \in Y_2$, m was single in M but is matched in M_2. If $w' \in X_1$, w' is matched with the same partner in M and M_2. If $w' \in X_2$, $M_2(w') =_{w'} M(w')$ by the construction of P. For any combination of m and w', if (m, w') blocks M_2, then it also blocks M, contradicting the stability of M.

Case (ii): $m \in Y_1 \cup Y_2$ and $w' \in X_3$. $w' \in X_3$ means that $(m', w') \in E_2$ in the bipartite graph G_2. This means that $m' \succ_{w'} M_1(w')$ by the construction of G_2. By the construction of M_1, $M_1(w') =_{w'} M(w')$, so that $m' \succ_{w'} M(w')$. By the same observation on m as Case (i), if (m, w') blocks M_2, then it also blocks M, a contradiction.

Case (iii): $m \in Y_3$ and $w' \in X_1 \cup X_2$. We further consider two cases: (iii)-(a) $(m, w') \in E_2$ and (iii)-(b) otherwise.

Case (iii)-(a): Since (m, w') blocks M_2, $w' \succ_m w$. Since $(m, w') \in E_2$, w' was in m's list when GS algorithm was applied to G_2. Since m is matched to w as a result of GS algorithm, m proposed to w' but was rejected. This means that w' is also matched in Q. Then w' must be in X_3, contradicting the assumption $w' \in X_1 \cup X_2$ of this case.

Case (iii)-(b): Since (m, w') blocks M_2, $m \succ_{w'} m'$, and since $w' \in X_1 \cup X_2$, $m' = M_2(w') = M_1(w')$. Because $m \in Y_3$, (m, w) is an edge in E_2. Consider the matching M_m created when constructing G_2. In M_m, (m, w'') is added to M_1, where w'' is a best woman for m in F. Since $(m, w) \in E_2$, (m, w) is a blocking pair for M_m, which implies $w \succ_m w''$. Since (m, w') blocks M_2, $w' \succ_m w$, so $w' \succ_m w''$. Then, (m, w') blocks M_m, and the edge (m, w') must be in E_2, a contradiction.

Case (iv): $m \in Y_3$ and $w' \in X_3$. Again, we consider two cases: (iv)-(a) $(m, w') \in E_2$ and (iv)-(b) otherwise.

Case (iv)-(a): For the same reason as Case (iii)-(a), we can show that during the execution of GS algorithm, m proposed to w' but was rejected. At this moment, w' was matched with a man better than m, and after that, she never becomes worse off, implying that $m' \succeq_{w'} m$ since w' is matched with m' in Q. Hence (m, w') cannot block M_2.

Case (iv)-(b): Since $(m', w') \in E_2$, $m' \succ_{w'} M_1(w')$. Since (m, w') blocks M_2, $m \succ_{w'} m'$, resulting that $m \succ_{w'} M_1(w')$. Then, for the same reason as Case (iii)-(b), we can show that (m, w') blocks M_m, and hence (m, w') is in E_2, a contradiction.

LEMMA 4.3. *Let m^* be the man described in Lemma 4.2. Then, $M_2 \cup \{(m^*, w^*)\}$, where w^* is the woman selected for m^* at line 9, satisfies* Π.

Proof. For simplicity, denote $H = M_2 \cup \{(m^*, w^*)\}$. By Claim 4.4, M_2 satisfies Π. Hence, if H does not satisfies Π, then there is an edge (m, w) in M_2 such that (m, w^*) or (m^*, w) blocks H.

First, suppose that (m^*, w) blocks H. Observe that $m \succeq_w M_1(w)$ (because if w is matched with different partners in M_1 and M_2, then she prefers the partner in M_2), and since (m^*, w) is a blocking pair, $m^* \succ_w m$, resulting that $m^* \succ_w M_1(w)$. Also, since (m^*, w) is a blocking pair, $w \succ_{m^*} w^*$. Then, since w^* is one of the best women for m^* in F, this means that (m^*, w) blocks M_{m^*}, i.e., $(m^*, w) \in E_2$. Then, since m^* is single in Q, he proposed to w during the GS algorithm, but was rejected. So, w must be in X_3 (in the classification defined in the proof of Claim 4.4), and is matched with a man better than m^* in Q, i.e., $m \succeq_w m^*$. This contradicts the assumption that (m^*, w) blocks H.

Next, suppose that (m, w^*) blocks H. Consider the classification Y_1, Y_2 and Y_3 used in the proof of Claim 4.4. If $m \in Y_1$, $H(m) = M(m)$, and if $m \in Y_2$, m was single in M but is matched in H. Note that w^* was single in M but is matched in H. Then, if (m, w^*) blocks H, it blocks M, a contradiction. Now, suppose that $m \in Y_3$. Since (m, w^*) blocks H, $w^* \succ_m w$. As $m \in Y_3$, $(m, w) \in E_2$. Consider the matching M_m, and suppose that $(m, w^+) \in M_m$. Then, $w^+ \succeq_m w^*$ since both w^* and w^+ were in F, and w^+ was selected rather than w^* as a partner of m in M_m. Recall that $(m, w) \in E_2$. This means that $w \succ_m w^+$ by the construction of E_2. So, $w \succ_m w^+ \succeq_m w^* \succ_m w$, a contradiction.

Since M^* at line 9 can be this matching $M_2 \cup \{(m^*, w^*)\}$, the following theorem is now immediate.

THEOREM 4.1. *If $|M| < \frac{8}{15}OPT$,* INCREASE *does not fail.*

COROLLARY 4.1. *The approximation ratio of* LOCALSEARCH *is at most 1.875.*

5 Concluding Remarks

If we consider more technical details in the analysis, which will make it much more complicated, it seems possible to obtain somewhat better approximation factor. Currently, we have an example for which our LOCALSEARCH fails for M such that $|M| = \frac{OPT}{1.75}$, which means that there is a room for improving analysis to obtain an approximation ratio of 1.75. One of the challenging future work is to investigate whether we can improve our analysis to prove this upper bound.

References

[1] D. J. Abraham, K. Cechlárová, D. F. Manlove and K. Mehlhorn, "Pareto optimality in house allocation problems," *Proc. ISAAC 2004*, LNCS 3341, pp. 3–15, 2004.

[2] D. J. Abraham, R. W. Irving, T. Kavitha and K. Mehlhorn, "Popular matchings," *Proc. SODA 2005*, pp. 424–432, 2005.

[3] V. Bansal, A. Agrawal and V. Malhotra, "Stable marriages with multiple partners: efficient search for an optimal solution," *Proc. ICALP 2003*, LNCS 2719, pp. 527–542, 2003.

[4] R. Bar-Yehuda and S. Even, "A local-ratio theorem for approximating the weighted vertex cover problem," In Analysis and Design of Algorithms for Combinatorial Problems, volume 25 of Annals of Disc. Math., Elsevier Science Publishing Company, Amsterdam, pp. 27–46, 1985.

[5] P. Berman and T. Fujito, "On the approximation properties of independent set problem in degree 3 graphs," *Proc. WADS 95*, LNCS 955, pp. 449–460, 1995.

[6] Canadian Resident Matching Service (CaRMS), http://www.carms.ca/

[7] K. Cechlárová, "On the complexity of exchange-stable roommates," *Discrete Applied Mathematics*, Vol. 116, pp. 279–287, 2002.

[8] T. Fleiner, "A fixed-point approach to stable matchings and some applications," *Mathematics of Operations Research*, Vol. 28, Issue 1, pp. 103–126, 2003.

[9] D. Gale and L. S. Shapley, "College admissions and the stability of marriage," *Amer. Math. Monthly*, Vol. 69, pp. 9–15, 1962.

[10] D. Gale and M. Sotomayor, "Some remarks on the stable matching problem," *Discrete Applied Mathematics*, Vol. 11, pp. 223–232, 1985.

[11] D. Gusfield and R. Irving, "The Stable Marriage Problem: Structure and Algorithms," MIT Press, Boston, MA, 1989.

[12] M. M. Halldórsson, R. Irving, K. Iwama, D. Manlove, S. Miyazaki, Y. Morita and S. Scott, "Approximability results for stable marriage problems with ties," *Theoretical Computer Science*, Vol. 306, pp. 431–447, 2003.

[13] M. M. Halldórsson, K. Iwama, S. Miyazaki and H. Yanagisawa, "Randomized approximation of the stable marriage problem," *Theoretical Computer Science*, Vol. 325, No. 3, pp. 439–465, 2004.

[14] M. M. Halldórsson, K. Iwama, S. Miyazaki and H. Yanagisawa, "Improved approximation of the stable marriage problem," *Proc. ESA 2003*, LNCS 2832, pp. 266–277, 2003.

[15] E. Halperin, "Improved approximation algorithms for the vertex cover problem in graphs and hypergraphs," *Proc. SODA 2000*, pp. 329–337, 2000.

[16] R. Irving, "Stable marriage and indifference," *Discrete Applied Mathematics*, Vol. 48, pp. 261–272, 1994.

[17] KavithaR. Irving, "Matching medical students to pairs of hospitals: a new variation on an old theme," *Proc. ESA 98*, LNCS 1461, pp. 381–392, 1998.

[18] R. W. Irving, D. F. Manlove and S. Scott, "The hospital/residents problem with ties," *Proc. SWAT 2000*, LNCS 1851, pp. 259–271, 2000.

[19] R. Irving, T. , K. Mehlhorn, D. Michail, and K. Paluch, "Rank-maximal matchings," *Proc. SODA 2004*, pp. 68–75, 2004.

[20] R.W. Irving, D.F. Manlove, S. Scott, "Strong stability in the hospitals/residents problem," *Proc. STACS 2003*, LNCS 2607, pp. 439–450, 2003.

[21] K. Iwama, D. F. Manlove, S. Miyazaki, and Y. Morita, "Stable marriage with incomplete lists and ties," *Proc. ICALP 99*, LNCS 1644, pp. 443–452, 1999.

[22] K. Iwama, S. Miyazaki and K. Okamoto, "A $(2 - c\log N/N)$-approximation algorithm for the stable marriage problem, *Proc. SWAT 2004*, LNCS 3111, pp. 349–361, 2004.

[23] K. Iwama, S. Miyazaki, N. Yamauchi, "A $(2 - c\frac{1}{\sqrt{N}})$-Approximation Algorithm for the Stable Marriage Problem," Proc, ISAAC 2005, LNCS 3827, pp. 902-914, 2005.

[24] Japanese Resident Matching Program (JRMP), http://www.jrmp.jp/

[25] G. Karakostas, "A better approximation ratio for the Vertex Cover problem," ECCC Report, TR04-084, 2004.

[26] M. Karpinski, "Polynomial time approximation schemes for some dense instances of NP-hard optimization problems," *Proc. RANDOM 97*, LNCS 1269, pp. 1–14, 1997.

[27] T. Kavitha, K. Mehlhorn, D. Michail, and K. Paluch, "Strongly stable matchings in time $O(nm)$ and extension to the hospitals-residents problem," *Proc. STACS 2004*, pp. 222–233, 2004.

[28] T. Le, V. Bhushan, and C. Amin, "First aid for the match, second edition," McGraw-Hill, Medical Publishing Division, 2001.

[29] D. F. Manlove, R. W. Irving, K. Iwama, S. Miyazaki, Y. Morita, "Hard variants of stable marriage," *Theoretical Computer Science*, Vol. 276, Issue 1-2, pp. 261–279, 2002.

[30] B. Monien and E. Speckenmeyer, "Ramsey numbers and an approximation algorithm for the vertex cover problem," *Acta Inf.*, Vol. 22, pp. 115–123, 1985.

[31] H. Nagamochi, Y. Nishida and T. Ibaraki, "Approximability of the minimum maximal matching problem in planar graphs," Institute of Electronics, Information and Communication Engineering, Transactions on Fundamentals, vol.E86-A, pp. 3251-3258, 2003.

[32] C.P. Teo, J.V. Sethuraman and W.P. Tan, "Gale-Shapley stable marriage problem revisited: Strategic issues and applications," *Proc. IPCO 99*, pp. 429–438, 1999.

[33] M. Zito,"Randomized techniques in combinatorial algorithmics," PhD thesis, Dept. of Computer Science, University of Warwick, 1999.

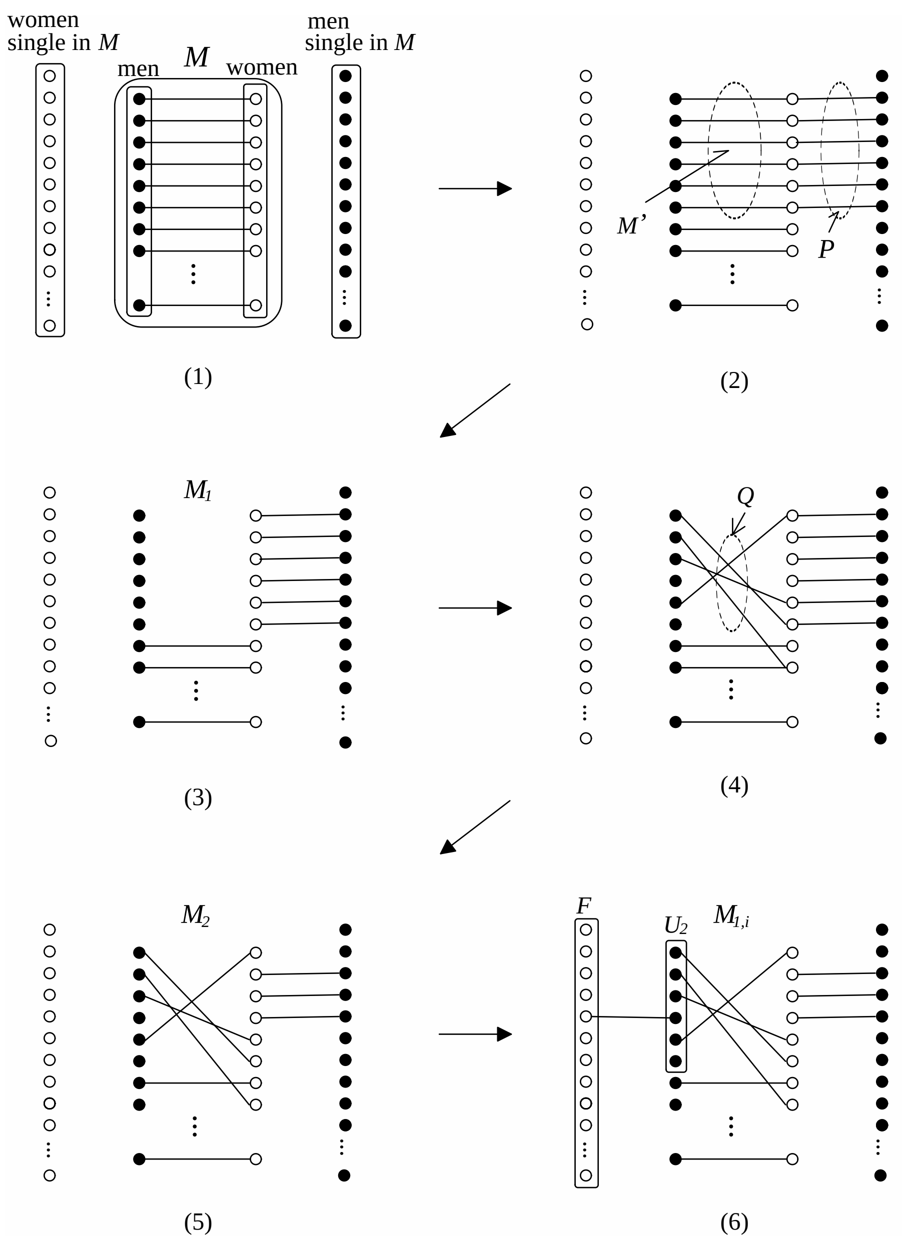

Figure 1: An example of computation by Increase

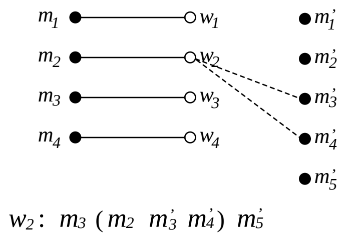

Figure 2: The construction of G_1

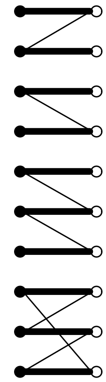

Figure 4: $G_{M_{opt},M}$

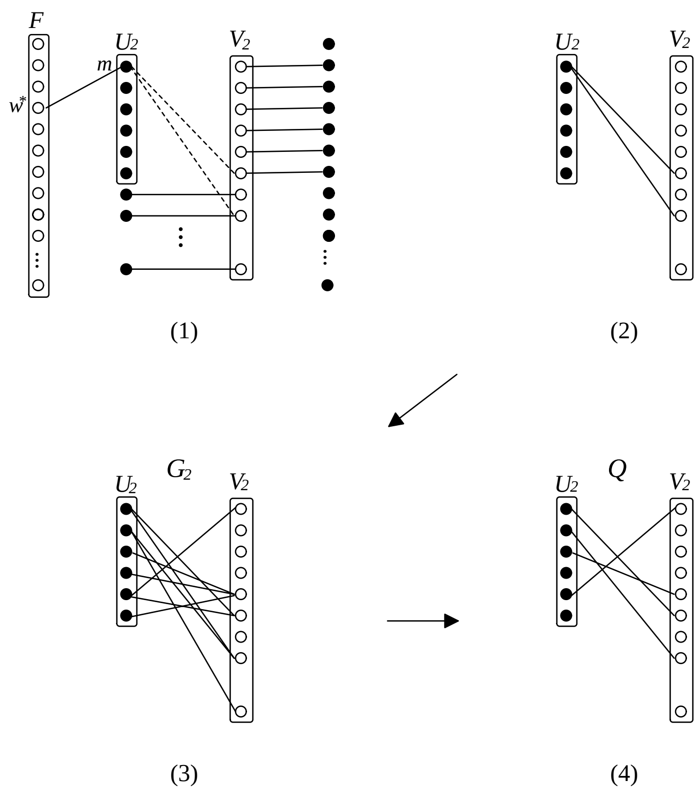

Figure 3: The bipartite graph G_2