Algorithmica (2011) 60: 3–20
DOI 10.1007/s00453-009-9371-7

Better and Simpler Approximation Algorithms for the Stable Marriage Problem

Zoltán Király

Received: 1 December 2008 / Accepted: 4 November 2009 / Published online: 20 November 2009

Abstract We first consider the problem of finding a maximum size stable matching if incomplete lists and ties are both allowed, but ties are on one side only. For this problem we give a simple, linear time $3/2$-approximation algorithm, improving on the best known approximation factor $5/3$ of Irving and Manlove (J. Comb. Optim., doi:10.1007/s10878-007-9133-x, 2007). Next, we show how this extends to the Hospitals/Residents problem with the same ratio if the residents have strict orders. We also give a simple linear time algorithm for the general problem with approximation factor $5/3$, improving the best known $15/8$-approximation algorithm of Iwama, Miyazaki and Yamauchi (SODA '07: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 288–297, 2007). For the cases considered in this paper it is NP-hard to approximate within a factor of $21/19$ by the result of Halldórsson et al. (ACM Transactions on Algorithms 3(3):30, 2007).

Our algorithms not only give better approximation ratios than the cited ones, but are much simpler and run significantly faster. Also we may drop a restriction used in (J. Comb. Optim., doi:10.1007/s10878-007-9133-x, 2007) and the analysis is substantially more moderate.

Preliminary versions of this paper appeared in (Király, Egres Technical Report TR-2008-04, www.cs.elte.hu/egres/, 2008; Király in Proceedings of MATCH-UP 2008: Matching Under Preferences—Algorithms and Complexity, Satellite Workshop of ICALP, July 6, 2008, Reykjavík, Iceland, pp. 36–45, 2008; Király in ESA 2008, Lecture Notes in Computer Science, vol. 5193, pp. 623–634, 2008). For the related results obtained thenceforth see Sect. 5.

Research is supported by EGRES group (MTA-ELTE), OTKA grants NK 67867, K 60802, and by Hungarian National Office for Research and Technology programme NKFP072-TUDORKA7.

Z. Király (✉)
Department of Computer Science and Communication Networks Laboratory, Eötvös University, Pázmány Péter sétány 1/C, 1117 Budapest, Hungary
e-mail: kiraly@cs.elte.hu

Keywords Stable matching · Hospitals/Residents problem · Approximation algorithms

1 Introduction

An instance of the stable marriage problem consists of a set U of N men, a set V of N women, and a preference list for each person, that is a weak linear order (ties are allowed) on some members of the opposite gender. A pair $(m \in U,\ w \in V)$ is called acceptable if m is on the list of w and w is on the list of m. We model acceptable pairs with a bipartite graph $G = (U, V, E)$, (where E is the set of acceptable pairs; we may assume that if w is not on the list of m then m is also missing from the list of w). A *matching* in this graph consists of mutually disjoint acceptable pairs. A matching M is *stable* if there is no blocking pair, where an acceptable pair is *blocking* if they strictly prefer each other to their current partners (the exact definition is given below). It is well-known that a stable matching always exists and can be found in linear time. An interesting problem, motivated by applications, is to find a stable matching of maximum size. This problem is known to be NP-hard for even very restricted cases [7, 12]. Moreover, it is APX-hard [2] and cannot be approximated within a factor of strictly less than $21/19$, even if ties occur only in the preference lists on one side only, furthermore if every list is either totally ordered or consists of a single tied pair [3]. Moreover, refining the ideas of [3], Yanagisawa [14] proved that an approximation within a factor of $4/3 - \varepsilon$ implies $2 - \varepsilon$ approximation of vertex cover, and this applies for the case when each tie has length two. If, moreover, ties occur only in the preference lists on one side only, he proved that an approximation within a factor of $5/4 - \varepsilon$ implies $2 - \varepsilon$ approximation of vertex cover. We note that interestingly the minimization version (where we are looking for a stable matching of minimum size) is also APX-hard [2].

As the applications of this problem are important, researchers started to develop good approximation algorithms in the last decade. We say that an algorithm is approximating with factor r if it gives a stable matching M with size $|M| \geq (1/r) \cdot |M_{\text{opt}}|$ where M_{opt} is a stable matching of maximum size. It is easy to give a 2-approximating algorithm, as running Algorithm GS of Gale and Shapley (see later) after an arbitrary tie-breaking gives a stable matching, and it evidently has size at least a half of any matching. The first non-trivial approximation algorithm was given by Halldórsson et al. [3] and was recently improved by Iwama, Miyazaki and Yamauchi [8] to a $15/8$-approximation. This was later improved for the special case, where ties are allowed on one side only and moreover only at the ends of the lists, by Irving and Manlove [5]. (We must emphasize that the second restriction is not needed for our results.) They gave a $5/3$-approximating algorithm for this special case. Their algorithm also applies for the Hospitals/Residents problem (see later) if residents have strictly ordered lists. If, moreover, ties are of size 2, Halldórsson et al. [3] gave an $8/5$-approximation and in [4] they described a randomized algorithm for this special case with expected factor of $10/7$. The paper of Irving and Manlove [5] also gives a detailed list of known and possible applications that motivate investigating approximation algorithms.

We think that our results also have some didactic importance. People teaching approximation algorithms usually look for a nice example, such that

– it is a simple and fast approximation algorithm,
– it solves an interesting and APX-hard problem,
– it gives the best known approximation factor for that problem,
– its correctness is straightforward, and
– it has a simple proof for the approximation ratio.

We have some nice examples (like Christofides' algorithm for metric TSP), but not too many of them. Section 2 offers a new example of this type.

We store the weak order of lists as priorities. For an acceptable pair (m, w) let $\mathrm{pri}(m, w)$ be an integer from 1 up to $d(m)$ representing the priority of w for m, where $d(m)$ denotes the degree of m, i.e., the size of the m's list. We say that $m \in U$ strictly prefers $w \in V$ to $w' \in V$ if $\mathrm{pri}(m, w) > \mathrm{pri}(m, w')$. Ties are represented by the same number, e.g., if m equally prefers w_1, w_2 and w_3 then $\mathrm{pri}(m, w_1) = \mathrm{pri}(m, w_2) = \mathrm{pri}(m, w_3)$. Of course, $\mathrm{pri}(m, w)$ is not related to $\mathrm{pri}(w, m)$. We represent these priorities in the figures by writing $\mathrm{pri}(m, w)$ and $\mathrm{pri}(w, m)$ close to the corresponding endvertex of edge mw ($\mathrm{pri}(m, w)$ is written near m, while $\mathrm{pri}(w, m)$ is written near w).

Let M be a matching. If m is *matched* in M, or in other words m is not *single*, we denote m's partner by $M(m)$. Similarly we use $M(w)$ for the partner of woman w.

Definition 1 A pair (m, w) is *blocking* if $mw \in E \setminus M$ (they are an acceptable pair and they are not matched) and

– m is either single or $\mathrm{pri}(m, w) > \mathrm{pri}(m, M(m))$, and
– w is either single or $\mathrm{pri}(w, m) > \mathrm{pri}(w, M(w))$.

Definition 2 A *matching* is called *stable* if there is no blocking edge.

The famous algorithm of Gale and Shapley [1] for finding a stable matching is the following. Initially every man is active and, by breaking ties arbitrarily, makes any strict order of acceptable women according to the priorities (higher priority comes before lower).

Each active man m proposes to the next woman w on his strict list if w exists, otherwise (if he has processed the whole list) m inactivates himself. If the proposal was (temporarily) accepted then m inactivates himself, otherwise, if m was rejected, m keeps on proposing to the next woman from his list.

Each woman w who got some proposals keeps the best man as a partner and rejects all other men. More precisely, the first man m who proposed to w will be her first partner ($M(w) := m$). Later if w gets a new proposal from another man m', she rejects m' if $\mathrm{pri}(w, m') \leq \mathrm{pri}(w, M(w))$; otherwise w rejects $M(w)$, then $M(w)$ is re-activated, and finally w keeps $M(w) := m'$ as a new partner. The algorithm finishes if every man is inactive (either has a partner or has searched over his strict list). This algorithm runs in $O(|E|)$ time if G is given by edge-lists and sorting is done by bucket sort.

Theorem 1 (Gale-Shapley) *Algorithm GS defined above always ends in a stable matching M.*

Proof Let $mw \in E \setminus M$. If m never made a proposal to w then in the end he has a partner w' who precedes w on m's strict list, consequently $\text{pri}(m, w') \geq \text{pri}(m, w)$. Otherwise, w rejected m at some point, when w had a partner m' not worse than m. Observe that after w received a proposal, she will always have a partner. Moreover, when w changes partner, she always chooses a (strictly) better one. Thus in the end $\text{pri}(w, M(w)) \geq \text{pri}(w, m') \geq \text{pri}(w, m)$, so mw is not blocking. □

Observe that after a run of GS no single man and single woman can form an acceptable pair. Consequently Algorithm GS gives a 2-approximation, and in the case, when the bipartite graph is complete (every woman-man pair is acceptable), it gives a stable matching of size N, i.e., the optimum.

In what follows, we will use not only the statement of this theorem (as most of the previous results do), but the Algorithm GS itself with some modifications/extensions.

In the *Hospitals/Residents* problem the roles of women are played by hospitals and the roles of men are played by residents. Moreover, each hospital w has a positive integer capacity $c(w)$, the number of free positions. Instead of matchings we consider *assignments*, that is a subgraph F of G, such that all residents have degree at most one in F, and each hospital w has degree at most $c(w)$ in F, i.e., $d_F(w) \leq c(w)$. For a resident m, who is assigned, $F(m)$ denotes the corresponding hospital. For a hospital w, $F(w)$ denotes the set of residents assigned to it. We say that hospital w is *full* if $|F(w)| = c(w)$ and otherwise *under-subscribed*. Here a pair (m, w) is *blocking* if $mw \in E \setminus F$ (they are an acceptable pair and they are not assigned to each other) and

– m is either single or $\text{pri}(m, w) > \text{pri}(m, F(m))$, and
– w is either under-subscribed or $\text{pri}(w, m) > \text{pri}(w, m')$ for at least one resident $m' \in F(w)$.

An assignment is *stable* if there is no blocking pair. It is easy to modify Algorithm GS to give a stable assignment for the Hospitals/Residents problem (for the details see Sect. 3).

In the next section we consider the special case of the maximum stable marriage problem, where each man's list is strictly ordered. We allow arbitrary number of arbitrarily long ties for each woman. We give a simple algorithm running in time $O(|E|)$. First we run Algorithm GS, then we give extra scores to single men, that raise their priorities. These men are re-activated and start making proposals from the beginning of their lists. A simple proof shows that this slightly modified algorithm gives a 3/2-approximation to the maximum stable marriage problem.

In Sect. 3 we show that this algorithm applies to the Hospitals/Residents problem as well in the (practically plausible) case when residents have strictly ordered lists, also giving 3/2-approximation for the maximum assignment in time $O(|E|)$.

Section 4 contains a slightly more complicated algorithm for the general case. First we run the algorithm of Sect. 2, then change the roles of men and women. In the second phase women get extra scores and make proposals to men. This algorithm still runs in linear-time, and gives a 5/3-approximation. Finally we propose some open problems and review some results obtained since the first version.

Several people asked the author, why did he presume to think that such a simple algorithmic idea might work? A partial answer can be found in the Appendix, where an unpublished preliminary work of the present author can be found about finding maximum matching in bipartite graphs. This was worked out in 2007 for educational purposes.

2 Men Have Strictly Ordered Lists

In this section we suppose that the lists of men are strictly ordered. We are going to define extra scores, $\pi(m)$ for every man with the following properties. Initially $\pi(m) = 0$ and at any time $0 \le \pi(m) < 1$ for each man. We also define adjusted priorities: $\mathrm{pri}'(m, w) := \mathrm{pri}(m, w)$ and $\mathrm{pri}'(w, m) := \mathrm{pri}(w, m) + \pi(m)$ for each acceptable pair (m, w). It is straightforward to see that if M is stable with respect to pri' then it is also stable with respect to pri.

We define a modification of Algorithm GS, that is called rmGS (reduced men-proposal GS), as follows. This algorithm starts with a stable matching, given extra scores and a set of active men. Run the original GS algorithm (active men make proposals; at the beginning of the algorithm they start from the beginning of their lists), where women use pri' to decide rejections. Stop when every man is inactive.

If some men with zero extra score remained single, we increase the score of those men to ε and re-activate them. In the next round they start making proposals from the beginning of their strict list. At any time let SM denote the set of single men, and $\Pi_0 := \{m \in U : \pi(m) = 0\}$. We fix $\varepsilon = 1/2$.

Our approximation algorithm is as follows:

ALGORITHM GSA1

run GS
FOR $m \in U \quad \pi(m) := 0$
WHILE $SM \cap \Pi_0 \neq \emptyset$
    FOR $m \in SM \cap \Pi_0$
        $\pi(m) := \varepsilon$
        re-activate m
    run rmGS

This simple algorithm runs in $O(|E|)$ time, as there are at most $2|E|$ proposals altogether. It is easy to see that Algorithm GSA1 gives a stable matching M with respect to the adjusted priority, hence M is stable for the original problem as well.

Let M_{opt} denote any maximum size stable matching (stable for the original priorities).

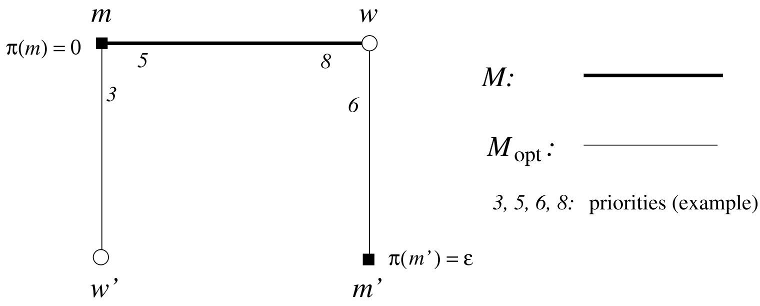


Fig. 1 A short augmenting path

Theorem 2 *If men have strictly ordered preference lists, M is the output of Algorithm GSA*1 *and M_{opt} is a maximum size stable matching then*

$$|M_{\mathrm{opt}}| \leq \frac{3}{2} \cdot |M|.$$

Proof We use an idea of Iwama, Miyazaki and Yamauchi [8]. Take the union of M and M_{opt}. We consider common edges as a two-cycle. Each component of $M \cup M_{\mathrm{opt}}$ is either an alternating cycle (of even length) or an alternating path. An alternating path component is called augmenting path if both end-edges are in M_{opt}. An augmenting path is called short, if it consists of 3 edges (see Fig. 1). It is enough to prove that in each component there are at most 3/2 times as many M_{opt}-edges as M-edges. This is clearly true for each component except for a short augmenting path.

We claim that a short augmenting path cannot exist. Suppose that $M(m) = w$, $M_{\mathrm{opt}}(m) = w' \neq w$, $M_{\mathrm{opt}}(w) = m' \neq m$ and that m' and w' are single in M. Observe first that w' never got a proposal during Algorithm GSA1. Consequently $\pi(m) = 0$ at the end, as otherwise he would have proposed to each acceptable woman. We may also conclude that $\mathrm{pri}(m, w) > \mathrm{pri}(m, w')$ because there are no ties in the men's lists. When the algorithm finishes, $\pi(m') = \varepsilon$, and m' proposed to every acceptable woman with this extra score, but w rejected him. This means that $\mathrm{pri}(w, m) = \mathrm{pri}'(w, m) \geq \mathrm{pri}'(w, m') = \mathrm{pri}(w, m') + \varepsilon$ consequently $\mathrm{pri}(w, m) > \mathrm{pri}(w, m')$. However, in this case edge mw blocks M_{opt}, a contradiction. □

We have an example (see Fig. 2) showing that for our algorithm this bound is tight (a possible order of proposals and extra score increases is the following: $mw, m'w, m'w'', m''w''$, $\pi(m'') = \varepsilon$, $m''w''$).

3 Hospitals/Residents with Strictly Ordered Residents' Lists

First we show that Algorithm GS with a slight modification always gives a stable assignment in linear time. Each hospital w manages to keep a set of buckets indexed by integers up to $d(w)$, containing each assigned resident m in the bucket indexed by

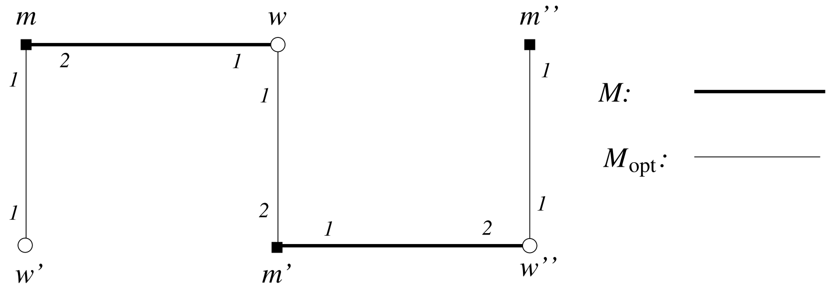


Fig. 2 An example where GSA1 gives $|M| = (2/3) \cdot |M_{\text{opt}}|$

pri(w, m); and w also stores the number of assigned residents, and if w is full then it also stores wpri(w), the priority of the worst assigned resident. If hospital w gets a new proposal from resident m then it accepts him either if w is under-subscribed or if pri$(w, m) >$ wpri(w). When hospital w is full and accepts, it rejects an arbitrary assigned resident m' with pri$(w, m') =$ wpri(w). Apart from these, the algorithm is the same. It clearly gives a stable assignment, and it is easy to see that also runs in $O(|E|)$ time (decision can be made in constant time, and updating the data at w needs constant time per operation plus total time $d(w)$ for finding the next nonempty bucket). We call this modified GS algorithm HRGS. As before, we are interested in giving a maximum size assignment, i.e., a stable assignment F with maximum number of edges (that is a maximum number of assigned residents).

We consider the Hospitals/Residents problem with the restriction that residents have strict orders on acceptable hospitals. Note, that for real-life applications of this scheme, this assumption is realistic. Here, as appropriate, residents get extra scores. The adjusted priorities are defined as in Sect. 2.

For a reader familiar with this topic it is straightforward that after "cloning" of hospitals the previous algorithm runs with the same approximation ratio. However, we describe an algorithm for this problem in some detail for not only to newcomers, but for three more reasons: (i) the cloning is not well defined in the literature, (ii) we give a linear time algorithm, and (iii) for showing an example and a theorem at the end of this section.

We modify GSA1 by replacing GS by HRGS and define rmHRGS as a modification of HRGS analogously to the derivation of rmGS from GS. Here *SM* denotes the set of unassigned residents and again $\Pi_0 := \{m \in U : \pi(m) = 0\}$.

ALGORITHM HRGSA1

run HRGS
FOR $m \in U$ $\pi(m) := 0$
WHILE $SM \cap \Pi_0 \neq \emptyset$
    FOR $m \in SM \cap \Pi_0$
        $\pi(m) := \varepsilon$
        re-activate m
    run rmHRGS

Algorithm HRGSA1 also runs in time $O(|E|)$ (hospital w need to have $2d(w)$ buckets), and gives a stable assignment F.

Theorem 3 *If residents have strictly ordered preference lists, F is the output of Algorithm HRGSA1 and F_{opt} is any maximum size stable assignment then*

$$|F_{\mathrm{opt}}| \le \frac{3}{2} \cdot |F|.$$

Proof We suppose that positions at hospital w are numbered by $1, \ldots, c(w)$. For the proof we make an auxiliary bipartite graph $G' = (U, V', E')$ and new preference lists as follows. The set U of residents remains unchanged. The set V' consists of the positions, i.e., $V' = \{w^i : w \in V,\ 1 \le i \le c(w)\}$. An edge connects resident m and position w^i if (m, w) was an acceptable pair (if hospital w was acceptable to m then all positions at w are acceptable to m). Each position w^i inherits the preference list of hospital w. For resident m we have to make a new (and also strict) preference list. Take the original list, and replace each w by $w^1 > w^2 > \cdots > w^{c(w)}$ (thus if w_1 was preferred by m to w_2 then all positions of w_1 will be preferred to all positions of w_2; and w_1's first position is preferred to the second, etc.). If F is an assignment in G then it defines a matching M in G' by distributing edges of F incident to a hospital w to distinct positions $w^1, w^2, \ldots, w^{d_F(w)}$, paying attention to the following. If m and m' are assigned to w by F and $\mathrm{pri}(w, m) > \mathrm{pri}(w, m')$ then m is matched to a smaller position number than m'.

And, conversely, any matching M of G' defines an assignment in G. The crucial observation is that if assignment F is stable in G then the associated matching M is stable in G', and if matching M is stable in G' then the associated assignment F is stable in G (this can be considered as a “folklore” theorem, widely used in the literature). Moreover, if we imagine running Algorithm GSA1 on G', the resulting matching M corresponds to the assignment F given by Algorithm HRGSA1. Using these observations Theorem 2 implies this one. □

We show that the example on Fig. 2 can be easily modified to show that this algorithm cannot achieve better approximation ratio than $3/2$, not even if all hospitals have large capacities and if each hospital has an absolutely unordered list (i.e., $\mathrm{pri}(w, m) = 1$ for every acceptable resident m).

We make c copies of the example shown in Fig. 3, one for each $i = 1, \ldots, c$. Then glue together the c copies of w_1^i, the c copies of w^i and the c copies of w_2^i. Assign capacity c to each hospital (w_1, w and w_2). The following is a possible run of Algorithm HRGSA1 yielding an assignment F with $|F| = 2c$, while $|F_{\mathrm{opt}}| = 3c$. First every resident m_i'' proposes to hospital w_2. Next, every resident m_i proposes to hospital w; now hospitals w and w_2 are full. Then every resident m_i' proposes first to w_2 and then to w, but they are always rejected. So every resident m_i' gets an extra score. They propose again to hospital w_2 and they succeed. Now every resident m_i'' gets an extra score, and proposes again to w_2 but they are rejected.

However, with a different type of restriction we are able to prove a stronger theorem. For a hospital w, let $\tau(w)$ denote the length of the longest tie for w, and let $\lambda := \max_{w \in V} \tau(w)/c(w)$.

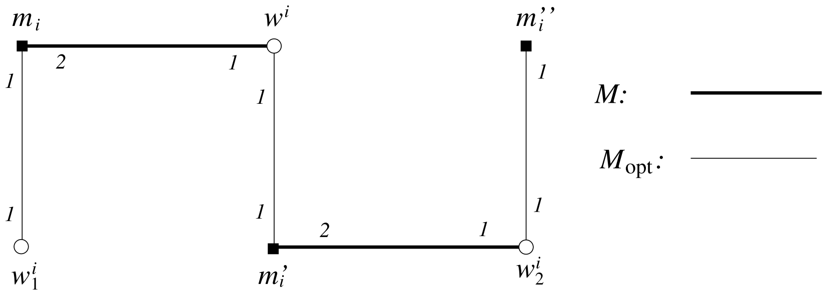


Fig. 3 A building block of the example where HRGSA1 gives $|F| = (2/3) \cdot |F_{\text{opt}}|$

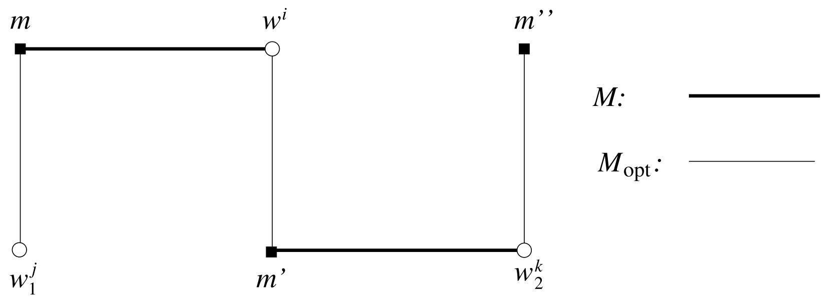


Fig. 4 A 5-path

Theorem 4 *Algorithm HRGSA*1 *gives approximation ratio not worse than*

$$\frac{4}{3} + \frac{\lambda}{6}.$$

Proof Again, we examine the components of the union of M and M_{opt} in G'. Call an augmenting path component a k-path, if it has k edges. By the proofs of Theorem 2 and Theorem 3, a 3-path cannot exist. First we need a technical lemma.

Lemma 1 *If* w^i *is the central vertex of a* 5*-path*: $w_1^j m w^i m' w_2^k m''$, *then* w, w_1, w_2 *are three distinct hospitals*, *hospital* w *is full and* $\text{pri}(w, m) = \text{pri}(w, m') = \text{wpri}(w)$ (*see Fig*. 4).

Proof As noted before, M is the same what we get, if we run algorithm GSA1 on G'.

(i) As w_1^j remained single, it never got a proposal. We have, as m never proposed to w_1^j, that $\pi(m) = 0$ and m prefers w^i to w_1^j in G'. We also conclude that $m'' w_1^j$ is not an edge of G', so, by the construction of G', $w_1 \neq w_2$. Observe that hospital w_1 is under-subscribed.

(ii) As m'' remained single, he proposed to w_2^k with his extra score $\pi(m'') = \varepsilon$, but was refused.

(iii) If m' never proposed to w^i then $\pi(m') = 0$ and m' prefers w_2^k to w^i. Edge $m'w_2^k$ is not a blocking edge for M_{opt}, so $\mathrm{pri}(w_2, m'') \geq \mathrm{pri}(w_2, m')$, and this contradicts to (ii).

(iv) Therefore m' proposed to w^i, but was rejected, consequently $\mathrm{pri}(w, m) \geq \mathrm{pri}(w, m')$. If $\mathrm{pri}(w, m) > \mathrm{pri}(w, m')$ then, by (i), mw^i is a blocking edge for M_{opt}. So we have $\mathrm{pri}(w, m) = \mathrm{pri}(w, m')$, and that hospital w is full. By (i), w_1 is under-subscribed, therefore $w \neq w_1$.

(v) Suppose that $w = w_2$. By the construction of G', $m''w^i$ is also an edge, so m'' proposed to w^i by his extra score, and he was refused. As $\pi(m) = 0$ by (i), we conclude that $\mathrm{pri}(w_2, m) > \mathrm{pri}(w_2, m'')$. By (i), $w_1 \neq w_2 = w$ and $\mathrm{pri}(m, w) > \mathrm{pri}(m, w_1)$, therefore edge mw_2^k of G' is blocking for M_{opt}. So $w \neq w_2$.

(vi) As $\mathrm{pri}(w, m) = \mathrm{pri}(w, m')$ by (iv), and m is assigned to w, but m' unsuccessfully proposed to w (he is assigned to $w_2 \neq w$), clearly $\mathrm{pri}(w, m) = \mathrm{wpri}(w)$.

□

Let K_r denote the number of r-paths, and L denote the number of M-edges in other components (i.e., in alternating cycles and in non-augmenting paths). Moreover, let W denote the set of hospitals w, such that there exists a position w^i which is a middle position of a 5-path. For a hospital $w \in W$, let $\phi(w)$ denote the number of 5-paths where w^i is the middle hospital position for some i. On one hand, $K_5 = \sum_{w \in W} \phi(w)$, on the other hand $2K_5 + 3K_7 + 4K_9 + \cdots + L \geq \sum_{w \in W} c(w)$, because all hospitals in W are full (using Lemma 1), so their every position takes part in a component as an end of an M-edge. By Lemma 1, for each $w \in W$ we have $\phi(w) \leq \tau(w)/2$ (a 5-path with center w^i contains two men with priority $\mathrm{wpri}(w)$), so $K_5 \leq \sum_{w \in W} \tau(w)/2 \leq \frac{\lambda}{2} \cdot \sum_{w \in W} c(w) \leq \frac{\lambda}{2} \cdot (2K_5 + 3K_7 + 4K_9 + \cdots + L)$.

$$\begin{aligned}\frac{|M_{\mathrm{opt}}|}{|M|} &\leq \frac{3K_5 + 4K_7 + 5K_9 + \cdots + L}{2K_5 + 3K_7 + 4K_9 + \cdots + L} \\ &\leq \frac{4}{3} + \frac{(1/3) \cdot K_5}{2K_5 + 3K_7 + 4K_9 + \cdots + L} \leq \frac{4}{3} + \frac{\lambda}{6}.\end{aligned}$$

□

Corollary 1 *If for each hospital the length of any tie is not more than half of the hospital's capacity then the approximation ratio of our algorithm is at most* $\frac{17}{12}$. *If every tie has length at most three*, *and every hospital has capacity at least* 100, *then the approximation ratio is better than* 1.339.

4 General Stable Marriage

Now we consider the general maximum stable marriage problem. First we will run the algorithm of Sect. 2, then we will change the roles of men and women. In the second phase women will get extra scores and they will propose to men.

Accordingly, we also use extra scores $\pi(w)$ for women: initially $\pi(w) = 0$ and at any time $0 \leq \pi(w) < 1$ for each woman w. We also re-define adjusted priorities: $\mathrm{pri}'(m, w) := \mathrm{pri}(m, w) + \pi(w)$ and $\mathrm{pri}'(w, m) := \mathrm{pri}(w, m) + \pi(m)$ for each acceptable pair (m, w). It is straightforward to see that if M is stable with respect to pri' then it is also stable with respect to pri.

In the first phase we run Algorithm GSA1, using the convention made in the description of algorithm GS, i.e., first each man breaks the ties on his list arbitrarily making a strict order for using proposals. Women do not get extra scores in this phase, but at the end men forget these strict orders and use pri$'$ to decide later. In the second phase, where we change the roles of men and women, we increase extra scores of women only. At the beginning of the second phase each woman makes any strict order of acceptable men according to the adjusted priorities (higher priority comes before lower), we call these lists as "strict lists" of women.

We define Algorithm rwGS (reduced woman-proposal GS) similarly to Algorithm rmGS. The algorithm starts with a stable matching, given extra scores and a set of active women. Run the original GS algorithm with interchanged roles: active women make proposals, and men use pri$'$ to decide rejections. But here we have a major difference. If a woman w with $\pi(w) = 0$ is rejected by her actual partner at any time during the process then she gets $\pi(w) := \varepsilon/2$ extra scores, activates herself, and starts making proposals *from the beginning of her strict list*. Stop when every woman is inactive.

If some women with less than ε extra score remained single, we increase the score of those women to ε and re-activate them. In the next round they start making proposals from the beginning of their strict list. At any time let SW denote the set of single women and $\Pi := \{w \in V : \pi(w) \le \varepsilon/2\}$. We use again $\varepsilon = 1/2$.

Our approximation algorithm is as follows.

ALGORITHM GSA2

Phase 1
run GSA1
Phase 2
FOR $w \in V \quad \pi(w) := 0$
WHILE $SW \cap \Pi \neq \emptyset$
  FOR $w \in SW \cap \Pi$
    $\pi(w) := \varepsilon$
    re-activate w
  run rwGS

First we claim that the algorithm runs in time $O(|E|)$. To see this we must consider two things. In Phase 2, every woman processes her strict list at most twice, so there are at most $2|E|$ proposals in the second phase. The strict lists of women can be calculated in $O(|E|)$ time altogether using bucket sort.

Lemma 2 *The matching M given by Algorithm GSA2 is stable with respect to* pri$'$, *consequently it is stable with respect to* pri.

Proof We use the facts that in Phase 1 the positions of women do not decline, while during Phase 2 the positions of men do not decline. Let mw be any edge in $E \setminus M$. First suppose that at the end $\pi(w) > 0$. After woman w got her final extra score, she started to propose to men: either w did not propose to m, in this case

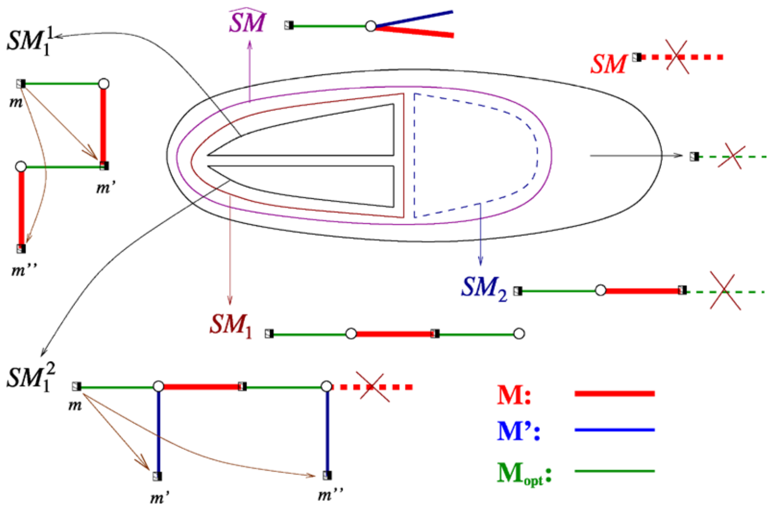


Fig. 5 Partitioning of single men

$\mathrm{pri}'(w, m) \le \mathrm{pri}'(w, M(w))$; or else w proposed to m but m rejected her, in this case $\mathrm{pri}'(m, w) \le \mathrm{pri}'(m, M(m))$. In both cases we get that the edge mw is not blocking. Now suppose that at the end $\pi(w) = 0$. In this case w is matched in M, and also matched in M', where M' denotes the matching at the end of Phase 1. Moreover $M(w) = M'(w) = m' \ne m$. In Phase 1, after man m got his final score, either m did not propose to w, in this case $\mathrm{pri}'(m, M(m)) \ge \mathrm{pri}(m, M(m)) \ge \mathrm{pri}(m, M'(m)) \ge \mathrm{pri}(m, w) = \mathrm{pri}'(m, w)$; or else m proposed to w but w rejected him, in this case $\mathrm{pri}'(w, M(w)) = \mathrm{pri}'(w, M'(w)) \ge \mathrm{pri}'(w, m)$. In both cases we get again that the edge mw is not blocking. □

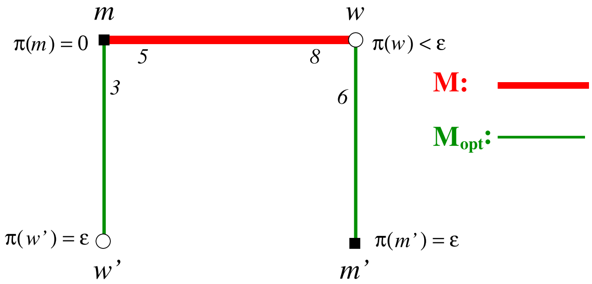
Theorem 5 *If M is the output of Algorithm GSA2 and M_{opt} is any maximum size stable matching then*

$$|M_{\mathrm{opt}}| \le \frac{5}{3} \cdot |M|.$$

Proof Consider components of $M \cup M_{\mathrm{opt}}$ as before. Here short augmenting path may exist. Let M' denote the matching given at the end of Phase 1. First, the technical Lemma 3 claims, that a single woman in a short augmenting path was matched in M'. After proving the lemma we will partition the men remained single at the end. (Actually we must consider the components of $M \cup M' \cup M_{\mathrm{opt}}$, but if we do this directly, it would lead to untreatable case analysis.) The most important class will be SM_1, consisting of single men, who are endvertex of a path starting with M_{opt}-M-M_{opt} edges (see Fig. 5). Our central Lemma 4 will state that $|SM_1| \le (2/3)|M|$. Using this lemma it will be easy to finish the proof of the theorem.

We will prove Lemma 4 by the following argument. By Lemma 3 such a path either continues with an M-edge or with an M'-edge. If more than the half continues by an M-edge, then we assign the two M-matched men on the path to the single man,

Fig. 6 A path of length three in $M \cup M_{\mathrm{opt}}$



who is the starting vertex (we assign m' and m'' to m on upper left part of Fig. 5). Otherwise we assign the two M'-neighbors of women on the path (we assign m' and m'' to m on lower left part of Fig. 5).

Lemma 3 *Suppose $M \cup M_{\mathrm{opt}}$ has a component that is an alternating path of length three, with the M-edge mw in the middle. Then $w' = M_{\mathrm{opt}}(m)$ is matched in M'.*

Proof Let $m' = M_{\mathrm{opt}}(w)$ (see Fig. 6) and suppose w' was single at the end of Phase 1 (i.e., w' is single in M'). As this is a component of $M \cup M_{\mathrm{opt}}$, clearly both m' and w' are single in M, and moreover, as matched men never become single in Phase 2, m' is also single in M'.

First we observe that as w' is single in M', m did not propose to her during Phase 1, so $\pi(m) = 0$. However m' remained single, so $\pi(m') = \varepsilon$ at the end of Phase 1.

In Phase 2, w did not propose to m' (m' remained single, thus he did not receive any proposals), so $\pi(w) \le \varepsilon/2$. We will use the fact, that $M(w) = m$. We consider two cases. If $M'(w) = m$ then in Phase 1, when w rejected m' the last time, she had $\mathrm{pri}'(w, m) \ge \mathrm{pri}'(w, m') = \mathrm{pri}(w, m') + \varepsilon$, so that in this case $\mathrm{pri}(w, m) > \mathrm{pri}(w, m')$. Otherwise, if $M'(w) \ne m$ then in Phase 2, woman w started to make proposals from the beginning of her strict list (that was made with respect to pri′ after Phase 1), but she did not propose to m', so $\mathrm{pri}'(w, m) \ge \mathrm{pri}'(w, m')$ also implying $\mathrm{pri}(w, m) > \mathrm{pri}(w, m')$.

At the beginning of Phase 2, $\pi(w')$ was set to ε, and w' remained single. This means that w' proposed to m and m rejected her. Consequently $\mathrm{pri}'(m, w) \ge \mathrm{pri}'(m, w')$, thus $\mathrm{pri}(m, w) > \mathrm{pri}(m, w')$. These arguments show that mw is blocking for M_{opt}, a contradiction. □

We continue the proof of the theorem. Let SM denote the set of single men at the end of the algorithm. First note, that men in SM were also single after Phase 1, since in Phase 2 men's positions do not decline. Let $\widehat{SM} \subseteq SM$ denote the set of those single men who are matched in M_{opt}. Observe that for each man $m \in \widehat{SM}$, woman $M_{\mathrm{opt}}(m)$ exists and is matched in both M' and M (at the end of any Phase at least one person in any acceptable pair is matched). We further partition $\widehat{SM}$: let SM_1 consist of each man $m \in \widehat{SM}$, for whom man $M(M_{\mathrm{opt}}(m))$ is matched in M_{opt}; and $SM_2 := \widehat{SM} \setminus SM_1$. Finally we partition SM_1: let $SM_1^1 := \{m \in SM_1 : M_{\mathrm{opt}}(M(M_{\mathrm{opt}}(m)))$ is matched in $M\}$ and $SM_1^2 := SM_1 \setminus SM_1^1$ (see Fig. 5). By Lemma 3, for every man m in SM_1^2, woman $M_{\mathrm{opt}}(M(M_{\mathrm{opt}}(m)))$ is matched in M' (i.e., at the end of Phase 1). The next lemma plays a crucial role in the proof of the theorem.

Lemma 4

$$|SM_1| \leq \frac{2}{3} \cdot |M|.$$

Proof

Case 1. $|SM_1^1| \geq |SM_1|/2$.

We form clubs, every club is led by a man in SM_1 and has one or two other men who are matched in M. For every man $m \in SM_1$ the second member of his club is $M(M_{\mathrm{opt}}(m))$. For each man $m \in SM_1^1$, his club contains a third member: $M(M_{\mathrm{opt}}(M(M_{\mathrm{opt}}(m))))$. We claim that these clubs are pairwise disjoint.

We formed one club for each man in SM_1 so it is enough to prove that any man m' who is matched in M belongs to at most one club. If $M(m')$ is single in M_{opt} then m' is not a member of any club. If $m = M_{\mathrm{opt}}(M(m')) \in SM$, then either $m \in SM_1$ and m' belongs to m's club or otherwise m' has no club at all. In the other case ($m \notin SM$), m' belongs to the club of $m^* = M_{\mathrm{opt}}(M(M_{\mathrm{opt}}(M(m'))))$ as a third member if m^* exists and $m^* \in SM_1^1$; and m' has no club otherwise.

Let MM denote the set of men who are matched in M. We have

$$|M| = |MM| \geq |SM_1| + |SM_1^1| \geq \frac{3}{2} \cdot |SM_1|.$$

Case 2. $|SM_1^2| > |SM_1|/2$.

In this case we form different clubs, here the non-leader members will be men matched in M'. For every man $m \in SM_1$ the second member of his club is $M'(M_{\mathrm{opt}}(m))$. For each man $m \in SM_1^2$, his club contains a third member: $M'(M_{\mathrm{opt}}(M(M_{\mathrm{opt}}(m))))$. We claim that these clubs are also pairwise disjoint.

If $M'(m')$ is single in M_{opt} then m' is not a member of any club. If $m = M_{\mathrm{opt}}(M'(m')) \in SM$, then either $m \in SM_1$ and m' belongs to m's club or otherwise m' has no club at all. Otherwise, m' belongs to the club of $m^* = M_{\mathrm{opt}}(M(M_{\mathrm{opt}}(M'(m'))))$ as a third member if m^* exists and $m^* \in SM_1^2$; and m' has no club otherwise.

Let MM' denote the set of men who are matched in M'. As men matched after Phase 1 remain matched till the end, we have

$$|M| = |MM| \geq |MM'| \geq |SM_1| + |SM_1^2| > \frac{3}{2} \cdot |SM_1|. \qquad \square$$

We are ready to finish the proof of the theorem. Let MM_{opt} denote the set of men who are matched in M_{opt}. We claim that $|MM \cap MM_{\mathrm{opt}}| \leq |MM| - |SM_2|$. This is true because $|SM_2|$ is the number of components of $M \cup M_{\mathrm{opt}}$ isomorphic to a path with two edges and with a woman in the middle; and for each such path the M-matched man is single in M_{opt}.

$$\begin{aligned} |M_{\mathrm{opt}}| &= |MM_{\mathrm{opt}}| = |MM \cap MM_{\mathrm{opt}}| + |SM \cap MM_{\mathrm{opt}}| \\ &\leq (|MM| - |SM_2|) + (|SM_1| + |SM_2|) \leq |M| + \frac{2}{3} \cdot |M| = \frac{5}{3} \cdot |M|. \qquad \square \end{aligned}$$

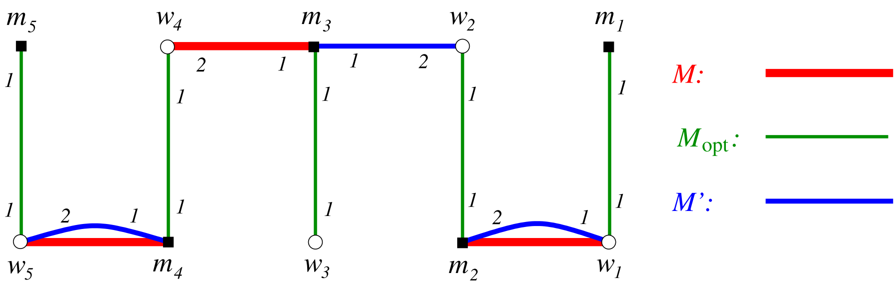


Fig. 7 Example of sharpness by Yanagisawa

5 Conjectures, Open Problems and Related Results Obtained Thenceforth

In the previous versions [9–11] and also in the talks given at the MATCH-UP workshop in Reykjavík and at ESA in Karlsruhe we posed several questions, conjectures and open problems. Two of them was solved meanwhile, we start now with these conjectures.

Conjecture 1 The performance ratio given for GSA2 is sharp.

This conjecture was proved to be true by Hiroki Yanagisawa [15], who gave a simple example where GSA2 really gives a matching of size $\frac{3}{5} \cdot M_{\text{opt}}$, see Fig. 7. In the first phase, m_1 proposes to w_1, m_3 to w_2 and m_4 to w_5. Then m_2 unsuccessfully proposes to w_1 and w_2, and m_5 unsuccessfully proposes to w_5. Then $\pi(m_2)$ and $\pi(m_5)$ are set to ε, and m_2 successfully proposes to w_1, but m_5 unsuccessfully to w_5. Now $\pi(m_1)$ is also set to ε, and he unsuccessfully proposes to w_1. After Phase 1 the matching $M' = \{m_2w_1, m_3w_2, m_4w_5\}$ arises. In Phase 2, first $\pi(w_3) = \pi(w_4) = \varepsilon$ are set, and w_4 successfully proposes to m_3, after that w_3 unsuccessfully proposes to m_3. Finally $\pi(w_2)$ is set to $\varepsilon/2$ and then to ε and she unsuccessfully proposes to m_3 and m_2, resulting $M = \{m_2w_1, m_3w_4, m_4w_5\}$.

Conjecture 2 Repeating GSA2 n times gives $3/2$ approximation (use $\varepsilon < 1/(n+1)$ instead of $1/2$ and in the repetitions raise extra scores of singles by ε).

Answered (partially) by Eric McDermid [13], who gave a $3/2$ approximation algorithm for the general case. He uses our algorithm, but not with simple repetitions, and uses novel and rather complicated techniques (and gives an $O(N\sqrt{N}|E|)$ algorithm rather than a linear one). Consequently, the original conjecture about the repetitions remained open.

Irwing and Manlove [6] implemented a basic version of our algorithm for the one-sided-ties Hospitals/Residents problem and gave a detailed comparison with their best heuristic. They tested carefully the algorithms with real-life and artificial data. We can summarize their result: for the most cases their best heuristic executed the best, but, on the average, our algorithm also gave a stable assignment of size at least 99.41% of their best one. We do not know too many other examples, where an algorithm with a guaranteed approximation ratio is so close to the best heuristic.

Conjecture 3 For the One-sided-ties case if someone gives $3/2 - \varepsilon$ approximation, then it implies something "surprising" (for example $3 - \varepsilon$ approximation for vertex cover in 3-uniform hypergraphs).

Open Problem 1 *Is it possible to improve the performance of GSA*2 *if we use the method of Halldórsson et al*. [3], *or the method of Iwama*, *Miyazaki and Yamauchi* [8] after *GSA*2*? And if this guarantee is not possible theoretically*, *can it be useful practically?*

Acknowledgement I am grateful to Tamás Fleiner for his invaluable advice. I am also indebted to the referees, whose valuable remarks helped us to improve the quality of the present paper.

Appendix

In this appendix we show, how a similar Gale-Shapley based algorithm can be used for the maximum matching problem. Here the input is a bipartite graph $G = (U, V, E)$, where U is a set of N men, and V is a set of N women, there are no priorities (or, equivalently, all priorities are set to zero), and we are looking for a matching of maximum size.

At any time let SM denote the set of single men, and $\Pi := \{m \in U : \pi(m) \le N\}$, and $\Pi_{N+} := \{m \in U : \pi(m) = N + 1\}$. We will call a natural number $0 \le i < N$ a whole, if no man has extra score exactly i, but some men have extra score $> i$, but $\le N$. If there exists a whole then i^* denotes the least one.

Our algorithm is as follows:

ALGORITHM GSMAX

run GS
FOR $m \in U$ $\pi(m) := 0$
WHILE $SM \cap \Pi \neq \emptyset$
    IF $\exists$ whole THEN
        i^* := the least whole
        FOR $m \in \Pi$
            IF $\pi(m) > i^*$ THEN $\pi(m) := N + 1$
    FOR $m \in SM \cap \Pi$
        $\pi(m) := \pi(m) + 1$ (1)
        re-activate m
    run rmGS

Theorem 6 *When Algorithm GSMAX finishes with a stable matching* M, *then* M *is a maximum size matching in* G.

Proof We begin with stating a lemma.

Lemma 5 *For a man m, let $\Gamma(m) = \{w \in V \mid mw \in E\}$. At any end of the WHILE loop, if $m \in \Pi_{N+}$ then every woman in $\Gamma(m)$ is matched to a man in Π_{N+}.*

Proof It is enough to prove the statement for the loop when $\pi(m)$ was set to $N+1$, because the positions of women do not decline. Let m be a man and consider the time when $\pi(m)$ was set to $N+1$, and let mw be any edge.

First we claim that $\pi(m) > i^*$ held that time, i.e., it is impossible that $\pi(m) = N$ and was incremented by one in line (1). This is because there are N men and $N+1$ numbers from zero up to N, so there must be a whole.

Once before m had extra score $\pi(m) = i^*$, and with this extra score he proposed to w, and was refused. This means that at the time of refusal, w had a partner with extra score at least i^*, and this still holds now, when no man has score i^*, and every man m' with higher score set to $\pi(m') = N+1$. □

Let $\Gamma(\Pi_{N+}) = \{w \in V \mid \exists m \in \Pi_{N+}, mw \in E\}$. We claim that at the end of the algorithm all men in Π and all women in $\Gamma(\Pi_{N+})$ are matched by M. The first statement follows from the halting criterion, while the second one follows from the lemma. Consequently, using the lemma again, there are $|\Pi_{N+}| - |\Gamma(\Pi_{N+})|$ single men, so M is maximum. □

This algorithm runs in time $O(N|E|)$.

References

1. Gale, D., Shapley, L.S.: College admissions and the stability of marriage. Am. Math. Mon. **69**, 9–15 (1962)
2. Halldórsson, M.M., Irving, R.W., Iwama, K., Manlove, D.F., Miyazaki, S., Morita, Y., Scott, S.: Approximability results for stable marriage problems with ties. Theor. Comput. Sci. **306**, 431–447 (2003)
3. Halldórsson, M.M., Iwama, K., Miyazaki, S., Yanagisawa, H.: Improved approximation results for the stable marriage problem. ACM Trans. Algorithms **3**(3), 30 (2007)
4. Halldórsson, M.M., Iwama, K., Miyazaki, S., Yanagisawa, H.: Randomized approximation of the stable marriage problem. Theor. Comput. Sci. **325**, 439–465 (2004)
5. Irving, R.W., Manlove, D.F.: Approximation algorithms for hard variants of the stable marriage and hospitals/residents problems. J. Comb. Optim. (2007). doi:10.1007/s10878-007-9133-x
6. Irving, R.W., Manlove, D.F.: Finding large stable matchings. J. Exp. Algorithmics **14** (2009). doi:10.1145/1498698.1537595
7. Iwama, K., Manlove, D.F., Miyazaki, S., Morita, Y.: Stable marriage with incomplete lists and ties. In: Proceedings of the 26th International Colloquium on Automata, Languages and Programming. Lecture Notes in Computer Science, vol. 1664, pp. 443–452. Springer, Berlin (1999)
8. Iwama, K., Miyazaki, S., Yamauchi, N.: A 1.875-approximation algorithm for the stable marriage problem. In: SODA '07: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 288–297. Society for Industrial and Applied Mathematics, Philadelphia (2007)
9. Király, Z.: Better and simpler approximation algorithms for the stable marriage problem. Egres Technical Report TR-2008-04, www.cs.elte.hu/egres/
10. Király, Z.: Better and simpler approximation algorithms for the stable marriage problem. In: Proceedings of MATCH-UP 2008: Matching Under Preferences—Algorithms and Complexity, Satellite Workshop of ICALP, July 6, 2008, Reykjavík, Iceland, pp. 36–45 (2008)
11. Király, Z.: Better and simpler approximation algorithms for the stable marriage problem. In: ESA 2008. Lecture Notes in Computer Science, vol. 5193, pp. 623–634. Springer, Berlin (2008)

12. Manlove, D.F., Irving, R.W., Iwama, K., Miyazaki, S., Morita, Y.: Hard variants of stable marriage. Theor. Comput. Sci. **276**, 261–279 (2002)
13. McDermid, E.J.: A $\frac{3}{2}$-approximation algorithm for general stable marriage. In: Automata, Languages and Programming, 36th International Colloquium, ICALP 2009, Rhodes, Greece. Lecture Notes in Computer Science, vol. 555, pp. 689–700. Springer, Berlin (2008)
14. Yanagisawa, H.: Approximation algorithms for stable marriage problems. PhD Thesis, www.lab2.kuis.kyoto-u.ac.jp/~yanagis/thesis_yanagis.pdf (2007)
15. Yanagisawa, H.: Personal communication