



# A fair procedure in a marriage market

Aditya Kuvalekar<sup>1</sup> · Antonio Romero-Medina<sup>2</sup>

Received: 25 January 2023 / Accepted: 29 December 2023

© The Author(s) 2024

## Abstract

We propose a new algorithm in the two-sided marriage market wherein both sides of the market propose in each round. The algorithm always yields a stable matching. Moreover, the outcome is a Rawlsian stable matching if the number of men and women is equal. Lastly, the algorithm can be computed in polynomial time and, from a practical standpoint, can be used in markets where fairness considerations are important.

**Keywords** Two-sided matching · Fair procedure · Deferred acceptance algorithm

**Mathematics Subject Classification** C72 · C78 · D41

## 1 Introduction

In a classical two-sided marriage market, the celebrated Deferred Acceptance Algorithm (DAA henceforth) due to Gale and Shapley (1962) proved that there always exists a stable matching—a matching for which no man-woman pair would leave their current partner to match with each other. DAA is a simple iterative algorithm in which one side of the market proposes to the other. The DAA is strategy-proof for the propos-

---

Aditya Kuvalekar is indebted to Jay Sethuraman for countless discussions and guidance in developing the early version of this algorithm (Kuvalekar 2014) that is now subsumed in this paper. We sincerely thank Vikram Manjunath for his detailed comments and feedback on the paper. We also thank Christine Cheng, Kevin Dick, Elliot Lipnowski, Kaustubh Nimkar, Efe Ok, Debraj Ray, Parag Pathak, Ennio Stacchetti, Leeat Yariv, and two anonymous referees, for their useful comments. Romero-Medina thanks financial support from Ministerio Economía y Competitividad grants ECO2017-87769-P, MDM 2014-043, Comunidad de Madrid grants EPUC3M11 (V PRICIT) and H2019/HUM-5891, and Spanish Ministerio de Ciencia e Innovación through Grant AEI PID2020-118022GB-I0/AEI/10.13039/501100011033.

---

✉ Aditya Kuvalekar  
a.kuvalekar@essex.ac.uk

Antonio Romero-Medina  
aromero@eco.uc3m.es

<sup>1</sup> University of Essex, Colchester, UK

<sup>2</sup> Universidad Carlos III de Madrid, Getafe, Spain

ing side, and in environments like school choice, the student-proposing mechanism Pareto-dominates all other stable matching(s). However, the asymmetry in proposals, whereby one side proposes while the other can only passively accept or reject, translates into an asymmetry of outcomes—the resulting stable matching is the most favorable stable outcome for the proposing side. This property can be problematic in markets where fairness considerations are important. This can be said for allocation mechanisms that go from task allocation platforms to dating apps. As (Knuth 1997) suggests:

The different algorithms considered until now favor men, and if we interchange the roles of men and women, they would become favorable to women. Such injustice is too shocking for the present day. Can we, therefore, find a solution that treats both sexes fairly?

Notwithstanding the unfairness embedded in the DAA, one of its appeals is its simplicity. This raises a natural question: can we devise a simple iterative algorithm in which both sides of the market make proposals, and the resulting outcome is stable but, hopefully, non-extremal?

To this end, we propose an algorithm in which both sides propose in each round and generates a stable matching. There are three reasons why we view our algorithm as an attractive alternative. First, the algorithm is “procedurally fair”—it treats both sides of the market equally [see (Klaus and Klijn 2006) e.g.]. Second, in markets with an equal number of men and women where people prefer being matched to someone over remaining single, it produces a Rawlsian stable matching—a stable matching with minimum regret.<sup>1</sup> In fact, even if the number of men and women is unequal, the outcome of our matching is Rawlsian in “regular markets.” A market is regular if, for every unstable matching, we can find at least one blocking pair such that both the agents of this blocking pair are matched to someone in any stable matching. While this requirement may be harder to verify, it is only a sufficient condition for obtaining a Rawlsian matching.

Third, the algorithm is computationally efficient—i.e., it can be run in polynomial time. The main contribution of the paper is to provide a procedurally fair algorithm with the iterative nature of DAA to produce a stable matching while allowing both sides of the market to make proposals.

It is well-known that in markets with even a slight imbalance between the number of men and women, there is essentially a unique stable matching (Ashlagi et al. 2017) if the market is sufficiently large. Therefore, given any sufficiently large market, our algorithm produces a Rawlsian stable matching with probability close to one regardless of whether the market is regular or not.<sup>2</sup>

<sup>1</sup> Regret of a matching is defined as the highest regret of a matched agent in a given matching. Notice that, in defining the Rawlsian criterion, we focus only on the set of matched agents. We argue that this is a more reasonable definition of the Rawlsian criterion since the set of matched agents is the same across every stable matching (Roth and Sotomayor 1990). This concept coincides with the minimum regret matching proposed by Stant Selkow (Knuth 1976).

<sup>2</sup> Of course, any algorithm that produces a stable matching would trivially obtain a Rawlsian stable matching with probability close to 1 in large markets with an unequal number of men and women, due to uniqueness of a stable matching in such markets. However, the outcomes can be extremal matchings depending on

## 1.1 Related literature

We are, obviously, not the first ones to address the asymmetry in DAA, wherein it picks the extreme points of the set of stable matchings depending on who the proposing side is. There are, primarily, two approaches to this problem. First is the idea of procedural fairness proposed by Klaus and Klijn (2006). They discuss how well-known randomized procedures such as Employment by Lotto proposed by Aldershof et al. (1999), or Random Order Mechanism by Ma (1996), that is based on (Roth and Vate 1990), achieve ex-ante fairness. While our algorithm is also procedurally fair in the sense of Klaus and Klijn (2006), a key distinction between the earlier procedures and the one we propose is the nature of the algorithm itself; that is, due to both sides proposing in every round, the outcomes tend to be at the center of the stable matchings' lattice rather than at the extremes (Example 4). This is also bolstered by the fact that our algorithm picks one of the Rawlsian stable matchings in regular markets, unlike these algorithms.

The second approach regarding fairness is based on the outcomes. For example, Gusfield (1987) provides a polynomial time algorithm to compute a Rawlsian stable matching in markets with an equal number of men and women. We do not impose such a requirement. Also (Knuth 1976) presents two algorithms to compute Rawlsian (or minimum regret) matchings by choosing arbitrary pairs. In both cases, the algorithm resolves to choose an agent at random after a series of non-random instances. None of the algorithms allows men and women to propose simultaneously. Our algorithm combines these two approaches by proposing a procedurally fair algorithm that produces a Rawlsian stable matching in markets studied by Knuth (1976) and Gusfield (1987). We do this by suitably modifying the DAA's most appealing feature—simplicity due to its iterative nature—while allowing both sides to make proposals in each round to tackle the asymmetry. Also, Romero-Medina (2005) defines an *Equitable Set* to capture fairness.<sup>3</sup> His proposed procedure is closer to the Random Priority mechanism due to Ma (1996) as he takes as primitive a random ordering over agents while we require no such order. Also, Masarani and Gokturk (1989) show the impossibilities of obtaining a fair stable matching, based on a Rawlsian notion of fairness. This would seem to contradict our algorithm. However, in their paper, the Rawlsian criterion is defined over the set of all the matchings, not just stable matchings.

In a contemporaneous paper, Dworzak (2021) proposes an iterative algorithm, Deferred Acceptance with Compensation Chains (DACC), in which both men and women make proposals. While similar in motivation, the key difference from our algorithm is that his procedure makes the agents propose in a pre-determined order (as opposed to simultaneously in our algorithm). Moreover, by varying the initial order,

---

the algorithm when the number of men and women is equal. In contrast, our algorithm would produce a Rawlsian stable matching in such cases. Therefore, the probability that our algorithm does not pick a Rawlsian stable matching vanishes in large markets for any number of men and women, so long as they are sufficiently large.

<sup>3</sup> The equitable set also focuses on the Rawlsian criterion. The Rawlsian criterion in these papers is defined over the set of all the agents, unlike ours. This seemingly minor distinction is important in markets where merely one agent can render all the matchings equally “unfair” according to his criterion. Lastly, whether this algorithm can be executed in polynomial time is unclear. See (Romero-Medina 2001) as well for another related algorithm.

DACC can obtain all the stable matchings, while, in general, we obtain a strict subset of stable matchings.

Ma (1996) proposes a procedure wherein we start with a *Random Priority* over agents. Following the priority, we start with an empty match and add agents one by one to the matching, satisfying all the blocking pairs *within the match*. That is, blocking pairs outside the agents not added so far are ignored. While this procedure does not distinguish between sexes, it can result in situations where, even in markets with an equal number of men and women, it can select only the extremal stable matchings while not selecting the Rawlsian ones. In contrast, in such markets, our algorithm will pick a strict subset of stable matchings that do not involve any extremal matchings if the Rawlsian matching(s) are non-extremal (See Sect. 4.1).

Using another approach, Teo and Sethuraman (1998) and Sethuraman et al. (2006) establish the existence of *the median stable matching(s)*, which is appealing from a fairness perspective. However, the main issue with median stable matching is that, if it can be computed efficiently, then computing the number of stable matchings can also be done efficiently; and the latter is known to be a computationally hard problem (Cheng 2010).

Cheng et al. (2011) propose a concept of *Center Stable Matchings* that are stable matchings whose maximum distance to any stable matching is as small as possible. These matchings are close to median stable matchings and can also be computed efficiently. However, primarily median stable matchings or center stable matchings are fair due to the properties of the stable matchings themselves. Our approach to fairness through the proposed algorithm focuses on the procedural fairness of the mechanism itself. In that sense, our algorithm is more like the mechanism due to (Ma 1996).

For the problem of school choice, when schools carry out their matching independently, Manjunath and Turhan (2016) propose a new mechanism to avoid wasted seats: a *Matching and Rematching* mechanism that yields a solution different from the two extremes. The outcome of their mechanism requires the coordination of the different school districts. The outcome need not be a stable matching and depends on the number of iterations for which the procedure is executed. Moreover, it is also not gender-neutral. Finally, the outcomes of algorithms such as one due to (Ma 1996) or Manjunath and Turhan (2016), when applied to all possible orders or partitions of agents, do not coincide with all the possible matchings that our algorithm generates.

The paper is organized as follows. Section 2 introduces the model and notation. Section 3 presents our algorithm and main results. Section 4 presents examples of the algorithm. Section 5 presents results on computational complexity, and Sect. 6 concludes.

## 2 Model

Let  $M$  and  $W$  denote two finite sets of men and women. Let  $A := M \cup W$  stand for the set of all agents. A generic agent in  $M(W)$  will be denoted by  $m(w)$ . When the distinction is immaterial, we will denote an agent by  $i$ . Let  $O(i) = W \cup \{i\}$  if  $i \in M$  and  $O(i) = M \cup \{i\}$  if  $i \in W$ ; denote the *opposite side* for agent  $i$ . Let

$N := \max\{|M|, |W|\}$ . For the sake of presentation, we assume that  $|M| = |W|$  and  $j \succ_i i$  for all  $j \in O(i) \setminus \{i\}$  and for all  $i$ . That is, any agent prefers being matched to someone over remaining single.<sup>4</sup>

Each agent  $i \in A$  is endowed with a preference relation,  $\succeq_i$  over  $O(i)$ .  $\succeq_i$  is a binary relation that is complete, reflexive, transitive, and antisymmetric (strict).

$\succ$  denotes the strict part of  $\succeq$ . Finally,  $\succeq := (\succeq_i)_{i \in A}$  is called a preference profile. An instance of a marriage market is a tuple  $(M, W, (\succ_i)_{i \in A})$ .

A bijection  $\mu : A \rightarrow A$  is called a matching if  $\mu(i) \in O(i)$  for all  $i \in A$ , and  $\mu(i) = j \implies \mu(j) = i$ .  $\mathcal{M}$  denotes the set of all possible matchings. Also,  $\mu(i) = i$  means that  $i$  is single or unmatched. Say that a matching  $\mu \in \mathcal{M}$  is *individual rational* if  $\mu(i) \neq i \implies \neg(i \succ_i \mu(i))$  for all  $i$ . We define the following notions of stability.

**Definition 1** A matching  $\mu$  is stable if it is individually rational and there is no  $(m, w) \in M \times W$ , such that  $w \succ_m \mu(m)$  and  $m \succ_w \mu(w)$ . If such a  $(m, w)$  exists, then we say that  $(m, w)$  is a “blocking pair” for  $\mu$ .

As it is well known, Gale and Shapley (1962) shows that a stable matching exists using the DAA. The set of stable matchings is a complete lattice.<sup>5</sup> In particular, the men- (women)-proposing DAA produces the stable matching most preferred by men (women). Roughly, in a men-proposing DAA, men propose in each round, and women can choose only from the men who propose to them. This passive role played by women, where they cannot initiate proposals themselves, leads to an extreme outcome—men-proposing DAA selects the stable matching most preferred by men. However, given the appeal of the DAA—a simple iterative procedure that produces a stable matching—the natural question that motivates us is the following: Can we have an iterative algorithm, along the lines of the DAA, in which both sides propose in each round and we obtain something non-extremal?<sup>6</sup> With this motivation, we provide an algorithm in which both sides propose in each round and the result is a stable matching—a “Rawlsian” one.

A few simple definitions will be useful in presenting the algorithm. First, let us define the regret of agent  $j \neq i$  for agent  $i$  to mean the position that  $j$  occupies in  $i$ 's preference ordering. Formally,  $\rho_i(j) := |\{k : k \succeq_i j\}|$ . The reason to call this regret is that  $\rho_i(j) > \rho_i(k) \implies k \succ_i j$ . We will sometimes refer to  $\rho_i(j)$  as the rank of agent  $j$  in  $i$ 's preference list.

**Definition 2** For any  $k \in \mathbb{N}$ ,  $k$ -truncated preferences of  $\succ$ , denoted by  $\succ^k$ , are preferences with the following two properties:

1. For each  $i \in A$  and  $j, l \neq i$  such that  $\max\{\rho_i(j), \rho_i(l)\} \leq k$ ,  $j \succeq_i l \Leftrightarrow j \succeq_i^k l$ .
2. For each  $i \in A$  and  $j \neq i$  such that  $\rho_i(j) > k$ ,  $i \succ_i^k j$ .

For a matching  $\mu$ , we say that  $(m, w)$  is a “ $\succ^k$  blocking pair” if  $w \succ_m^k \mu(m)$  and  $m \succ_w^k \mu(w)$ .

<sup>4</sup> Our algorithm produces a stable matching even if we relax these two. We need an additional condition specified after Theorem 1 to obtain a Rawlsian stable matching.

<sup>5</sup> To be precise, it is a complete lattice under two natural orders:  $\succ_M$  and  $\succ_W$ . We say that  $\mu \succ_M \mu'$  iff  $\mu \neq \mu'$  and  $\mu(m) \succeq_m \mu'(m)$  for all  $m \in M$ . We define  $\succ_W$  similarly. See (Roth and Sotomayor 1990) for more details.

<sup>6</sup> As mentioned in Sect. 1.1, several papers address a similar question.

**Table 1** Preferences for Example 1

$\succ_{m_1}$	$\succ_{m_2}$	$\succ_{m_3}$	$\succ_{m_4}$	$\succ_{w_1}$	$\succ_{w_2}$	$\succ_{w_3}$	$\succ_{w_4}$
$w_1$	$w_2$	$w_3$	$w_4$	$m_2$	$m_3$	$m_4$	$m_1$
$w_2$	$w_3$	$w_4$	$w_1$	$m_3$	$m_4$	$m_1$	$m_2$
$w_3$	$w_4$	$w_1$	$w_2$	$m_4$	$m_1$	$m_2$	$m_3$
$w_4$	$w_1$	$w_2$	$w_3$	$m_1$	$m_2$	$m_3$	$m_4$

Simply speaking,  $\succ^k$  is a truncation of  $\succ$  that looks only at the top  $k$  agents for each  $i \in A$ .

Now, we define a  $k$ -stable matching and a  $k$ -stable submatching.

**Definition 3** Say that an individually rational matching  $\mu \in \mathcal{M}$  is “ $k$ -stable” if it is a stable matching with respect to  $\succ^k$ . Also, say that a matching  $\mu \in \mathcal{M}$  is “ $k$ -stable with active agents in  $V \subset A$ ”, if, for every  $\succ^k$  blocking pair  $(m, w)$  of  $\mu$ ,  $\{m, w\} \not\subset V$

**Definition 4** Given an individually rational matching  $\mu$  and a number  $k \in \mathbb{N}$ , such that  $\mu(i) \neq i \implies \rho_i(\mu(i)) \leq k$ ,  $\mu'$  is called “a submatching” of  $\mu$  if,  $\mu'(i) \neq i \implies \mu'(i) = \mu(i)$ .  $\mu'$  is called  $k$ -stable submatching of  $\mu$  with active agents in  $V \subset A$ , if  $\mu'$  is a submatching of  $\mu$  that is  $k$ -stable with active agents in  $V$ .

### 3 The algorithm

A broad overview of our algorithm is as follows. In any Round  $k$ , we start with an existing match. Agents propose to either their top  $k$  agents if they are unmatched, or to those sets of agents weakly better than their current match. Based on these proposals, for each agent, we compute the set of mutually proposing agents—the set to whom (s)he proposes and from which (s)he also receives a proposal. Using these, we construct a graph with nodes as agents and edges given by each agent, pointing to his/her top mutually proposing agent. This graph contains cycles. We form a tentative match by randomly breaking these cycles in favor of men or women, and we remove those sets of agents to redraw the graph. Once the graph is empty, we check whether the tentative match is stable with respect to the truncated (top  $k$  agents) preferences. If yes, we move to the next round; and otherwise, we construct a stable (with respect to the truncated preferences) matching before proceeding to the next round. We stop the algorithm if, at any point, the tentative matching is stable.

#### 3.1 An example

**Example 1** Below is a simple example of a market in which one could expect to obtain an outcome different than the DAA outcome should the two sides be allowed to make proposals in each round (Table 1). ■

The DAA outcomes are:

$$\{(m_1, w_1), (m_2, w_2), (m_3, w_3), (m_4, w_4)\} \text{ and } \{(m_1, w_4), (m_2, w_1), (m_3, w_2), (m_4, w_3)\}.$$

Imagine a procedure whereby the agents from both sides of the market make proposals to a set of agents in every round. If there is no agent to whom they are proposing and who is also proposing to them, then they expand their *proposing to* set of agents in the following round. They start by proposing to the top agent according to their preferences and then expand the set of agents they propose to progressively.

In the first round, all the agents propose to their top choice, and no pair of agents is “mutually proposing.” For example,  $m_1$  proposes to  $w_1$ , while  $w_1$  proposes to  $m_2$ , and so on. In the following round, for example,  $m_1$  proposes to  $w_1$  and  $w_2$ . However,  $w_1$  proposes to  $m_2$  and  $m_3$ , while  $w_2$  proposes to  $m_3$  and  $m_4$ . It is easy to see that there are no two agents who propose to each other in this round, either.

Let us move to round 3. Consider agent  $m_1$ . He proposes to  $w_1, w_2$  and  $w_3$ . Observe that  $w_2$  proposes to  $m_3, m_4$  and  $m_1$ , while  $w_3$  proposes to  $m_4, m_1$  and  $m_2$ . That is,  $m_1$  is interested in being matched with two agents who are also interested in being matched with  $m_1$ . Those two agents are  $w_2$  and  $w_3$ .  $m_1$  would prefer being matched to  $w_2$  over  $w_3$ . But observe that  $w_2$  also has two mutually proposing agents ( $m_1$  and  $m_4$ ), and she prefers  $m_4$  over  $m_1$ . Continuing this way, if we look at the top agent for each person from the list of their mutually acceptable agents in this round, we obtain a cycle as below.

$$m_1 \rightarrow w_2 \rightarrow m_4 \rightarrow w_1 \rightarrow m_3 \rightarrow w_4 \rightarrow m_2 \rightarrow w_3 \rightarrow m_1$$

The outcome depends on breaking this cycle in favor of either the men or the women; we obtain the following two matches respectively:

$$\{(m_1, w_2), (m_2, w_3), (m_3, w_4), (m_4, w_1)\} \text{ or } \{(m_1, w_3), (m_2, w_4), (m_3, w_1), (m_4, w_2)\}$$

The algorithm we propose makes this simple idea from the above example formal and solves a complicated cycling problem that can often arise in markets with arbitrary preferences. However, at the core, the motivation and the essence are embodied in this simple example.

### 3.2 Procedure $\mathbb{H}$

We now begin a formal description of our algorithm. To this end, we first describe a procedure  $\mathbb{H}$  that forms an important ingredient of the algorithm, and then we will describe the RO mechanism due to Ma (1996).

Given an arbitrary matching  $\mu$ , a subset of agents  $V \subset A$ , and a number  $k \in \mathbb{N}$ , define, for each  $i \in V$ ,

$$S_{i,V,k} := \{j \in V : \rho_i(j) \leq k, j \succeq_i \mu(i)\}$$

$$T_{i,V,k} := \{j \in V : j \in S_{i,V,k}, i \in S_{j,V,k}\}.$$

As we shall see in the description of the algorithm,  $S_{i,V,k}$  is the set of agents from set  $V$  that agent  $i$  proposes to in round  $k$ , and  $T_{i,V,k}$  is the set of mutually proposing agents for  $i$ . That is, these are agents from set  $V$  that  $i$  proposes to and also receives

a proposal from. For each  $i \in V$  such that  $T_{i,V,k} \neq \emptyset$ , define,

$$\alpha_{i,V,k} := \{j : j \succ_i j', \forall j, j' \in T_{i,V,k}, j' \neq j\}.$$

In words,  $\alpha_{i,V,k}$  is the most preferred agent of  $i$  who also proposes to  $i$ . Since the preferences are strict,  $\alpha_{i,V,k}$  is a singleton.

We will use some simple graphs in  $\mathbb{H}$ . To this end, suppose that we have a directed bipartite graph  $G = (V, E)$  with the following properties:

- (i)  $V \subset A$  is the set of nodes, and  $E$  is the set of directed edges, such that if  $(i, j) \in E$ , then exactly one of  $(i, j)$  is a man and the other is a woman.
- (ii)  $(i, j) \in E$  if and only if  $j = \alpha_{i,V,k}$ . That is,  $j$  is  $i$ 's most preferred agent amongst those who also propose to  $i$ . Notice that, if  $i$  has an incoming edge, it means that the set of agents that propose to  $i$ , and  $i$  proposes to them is nonempty. Since the preferences are strict, there is a unique most preferred agent from this set according to  $i$ . Therefore, the outdegree (number of outgoing edges) of  $i$  is exactly 1.

First, we claim that such a graph will contain disjoint cycles if  $E$  is nonempty. To see why there will be disjoint cycles if  $E$  is nonempty, start with some node, say  $m_1$ , with an outgoing edge to  $w_1$ . Since  $w_1$  must have exactly one outgoing edge, we have found a cycle if it is  $m_1$ . Otherwise, let it be  $m_2$ . Since the number of agents is finite, we must eventually form a cycle. That the cycles will be disjoint is immediate because any node has an outdegree of, at most, 1.

**Definition 5** Consider a cycle  $C = \{m_1 \rightarrow w_1 \rightarrow m_2 \rightarrow w_2 \dots w_k \rightarrow m_1\}$  and a matching  $\mu$ . We refer to the following operation as: “resolve a cycle in favor of men given a matching  $\mu$ .”

**Input:**  $\mu, C$ .

**Output:**  $\mu'$ .

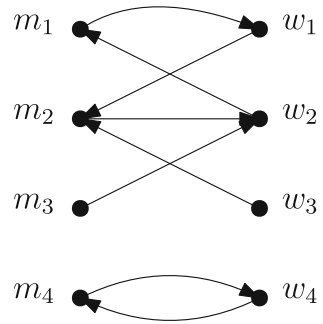
1. Set  $\mu' = \mu$ .
2. For each  $i \in C$ , set  $\mu'(i) = i$  and  $\mu'(\mu(i)) = \mu(i)$ . And, for each  $j$  such that  $\mu(j) \in C$ , set  $\mu'(j) = j$ .
3. For each  $(m, w) \in C$  such that  $m \rightarrow w$ , set  $\mu'(m) = w$ ,  $\mu'(w) = m$ . In words, if there is an edge from  $m$  to  $w$ , we match  $m$  and  $w$ .

Similarly, if, for every edge  $(w, m)$  in  $C$ , if we match  $m$  and  $w$ , we refer to that as “resolving a cycle in favor of women given a matching  $\mu$ .”

For example, given a cycle  $m_1 \rightarrow w_1 \rightarrow m_2 \rightarrow w_2 \rightarrow m_1$ , we can resolve it in two ways. We can either match  $\{(m_1, w_1), (m_2, w_2)\}$  or  $\{(m_1, w_2), (m_2, w_1)\}$ .



Fig. 1 Example of a graph of  $\mathbb{H}$



Now, we describe the procedure  $\mathbb{H}$  below.

**Algorithm 1:** Procedure  $\mathbb{H}$

```

input :  $\mu, V, k$ 
output:  $\mu', V'$ 
1 initialization: Let  $\mu' = \mu$ . Draw a graph  $G = (V, E)$  where  $(i, j) \in E$  iff
    $j = \alpha_{i,V,k}$ .
2 if  $E = \emptyset$ , then
3   |  $V' = \emptyset$ ;
4 else
5   | for Each cycle  $C$  in  $G$ , do
6     |   Resolve  $C$  in favor of either men or women with probability  $\frac{1}{2}$  each.;
7     |   Let the output matching be  $\mu'$ ;
8     |    $V' = V' \setminus C$ ;
9   | end
10 end

```

Notice that  $k$  is not used anywhere in  $\mathbb{H}$ . However, recall that the graph  $G$  is drawn by computing  $S_{i,V,k}$  and  $T_{i,V,k}$  for each  $i$ . Computing those objects requires a number  $k$ .

**Example 2** (Example of procedure  $\mathbb{H}$ ) As a simple example, suppose that, for some  $k$  and an empty matching  $\mu$  (all agents are single), we obtain the graph in Fig. 1 in running  $\mathbb{H}(\mu, A, k)$ . ■

There are two cycles here:  $(m_1 \rightarrow w_1 \rightarrow m_2 \rightarrow w_2 \rightarrow m_1)$  and  $(m_4 \rightarrow w_4 \rightarrow m_4)$ . We first unmatched everyone involved in any cycle. Thereafter, we resolve each cycle randomly in favor of either the men or the women. Regardless of how we resolve the cycle involving  $m_4$  and  $w_4$ , we obtain the match  $\mu'(m_4) = w_4$  and  $\mu'(w_4) = m_4$ . For the first cycle, say we resolve it in favor of the women. Then, we obtain,  $\{(m_1, w_2), (m_2, w_1)\}$  as the two new matches in  $\mu'$ . Finally, since  $m_3$  and  $w_3$  are not involved in any cycle, we set  $V' = \{m_3, w_3\}$ .

**3.3 The algorithm**

We call steps 3–16 above a “Round  $k$ .”

**Algorithm 2: Main**


---

**input** : An instance of a marriage market:  $M, W, \succ$ .  
**output**: A matching.

- 1 initialization: Set  $\mu(i) = i$  for all  $i$ , and  $k = 1$ ;
- 2 **while**  $k \leq N$  **do**
- 3     Set  $V = A$ ;
- 4     **repeat**
- 5          $(\mu', V') = \mathbb{H}(\mu, V, k)$ ;
- 6          $\mu = \mu', V = V'$ ;
- 7     **until**  $V = \emptyset$ ;
- 8     **if**  $\mu$  is  $k$ -stable, **then**
- 9          $k = k + 1$ ;
- 10    **else**
- 11         Let  $\mu'$  be some  $k$ -stable submatching of  $\mu$  within  $V = \{i : \mu'(i) \neq i\}$ ;
- 12          $\mu = RO(\mu', M, W, \succ^k, k)$ . Set  $V = A, k = k + 1$ ;
- 13    **end**
- 14    **if**  $\mu$  is stable, **then**
- 15         Terminate with output  $\mu$
- 16    **end**
- 17 **end**

---

**Algorithm 3: The RO mechanism**


---

**input** :  $\mu', M, W, \succ^k, k$   
**output**:  $\mu$

- 1 Set  $V = \{i : \mu'(i) \neq i\}, R = A \setminus V$ ;
- 2 **repeat**
- 3     Pick an agent  $i \in R$  randomly and set  $V = V \cup \{i\}, R = A \setminus V$ ;
- 4     **repeat**
- 5          $V' = V$ ;
- 6          $(\mu'', V') = \mathbb{H}(\mu', V', k)$ ;
- 7          $\mu' = \mu''$ ;
- 8     **until**  $\mu'$  is  $k$ -stable with active agents in  $V$ ;
- 9     **until**  $R = \emptyset$ ;
- 10  $\mu = \mu'$

---

As we show in Lemma 2 in the appendix, step 12 above is the “Random Order Mechanism” (RO mechanism henceforth) due to Ma (1996). We use it to generate a  $k$ -stable matching in case the procedure  $\mathbb{H}$  fails to produce a  $k$ -stable matching. We now describe the RO mechanism in Algorithm 3. The original version of the RO mechanism starts with an empty matching, and agents are “made active” in random order. From among the active agents, we form a stable matching. After doing so, we make the next agent active. In the above algorithm, rather than starting with an empty matching, we start with a  $k$ -stable submatching. Thereafter, we make agents active in random order and apply  $\mathbb{H}$  repeatedly to obtain a  $k$ -stable matching with an updated set of active agents.<sup>7</sup>

<sup>7</sup> The original RO mechanism is not stated in terms of  $\mathbb{H}$ . However, the procedure described here converges to produce a  $k$ -stable matching, and it is essentially the same as the original RO mechanism illustrated formally in Lemma 2 in the Appendix.

### 3.3.1 Properties of the algorithm

The Algorithm 2 is well-defined and produces a stable matching. In particular, it produces a Rawlsian stable matching, which we define formally below. For any stable matching  $\mu$ , define the regret of a matching  $\mu$  as

$$f(\mu) := \max\{\rho_i(\mu(i)) : i \in A \text{ such that } \mu(i) \neq i\}$$

$f(\mu)$  is the regret of the agent with the maximum regret. Notice that we focus only on agents who are matched in defining the Rawlsian score. Since the set of single agents is identical across all stable matchings [Theorem 2.2, (Roth and Sotomayor 1990)], to consider the well-being of the single agents in comparing two stable matchings is unappealing. A Rawlsian stable matching  $\mu$  is a stable matching with minimum regret, i.e.,  $\mu$  is a Rawlsian stable matching if  $f(\mu) \leq f(\mu')$  for all stable matchings  $\mu'$ .

**Theorem 1** *For every instance of the stable marriage problem, Algorithm 2 terminates to produce a stable matching. Moreover, the output is a Rawlsian stable matching.*

**Proof** To prove that the algorithm terminates amounts to proving that steps 4 – 7 in Algorithm 2 do not cycle for any  $k$ . This is proved in Lemma 1 below.

**Lemma 1** *For every instance of a marriage market, Algorithm 2 reaches Step 8 for every  $k$ . That is, steps 4 – 7 in Algorithm 2 do not cycle for any  $k \in \mathbb{N}$ .*

**Proof** Notice that we start with  $V = A$ . If  $E$  is empty, then we set  $V = V' = \emptyset$  and proceed. If not, then any iteration of  $\mathbb{H}$  resolves all the cycles. Moreover, the output  $V'$  is a *strict subset* of  $V$ . Therefore, eventually, we must reach a situation where  $E$  is empty, and therefore,  $V$  is empty. Hence, we go to Step 8.  $\square$

If the algorithm terminates with  $k = N$ , then we have a stable matching because  $N$ –stable matchings and stable matchings are equivalent by definition. If we terminate for  $k < N$ , then the stopping condition establishes that  $\mu$  is stable.

Therefore, we need to only prove that  $\mu$  is a Rawlsian stable. To this end, suppose that  $\mu$  is not a Rawlsian stable matching. Thus,  $\exists$  a stable matching  $\mu'$  such that  $r' := f(\mu') < f(\mu)$ . Let us consider what happens in Round  $r'$ . Since the algorithm did not terminate in Round  $r'$ , the output in this round, say  $\mu''$  is  $r'$ –stable but is not stable. Since  $\mu'$  is stable, it is also  $r'$ –stable. The following claim will help us complete the proof.

**Claim 1** *There is at least one blocking pair in  $\mu''$ ,  $(m, w)$ , such that both  $m$  and  $w$  are matched to someone in  $\mu''$ .*

**Proof** By Theorem 2.22 of Roth and Sotomayor (1990), the set of agents who are single is the same across all the stable matchings. Therefore, if we view the truncated market with preferences given by  $\succ^{r'}$ , then we know that the set of matched agents in  $\mu'$  and  $\mu''$  is the same as both are  $r'$ –stable. At the same time, since  $\mu'$  is a stable matching, we also have, from the same theorem, that the set of matched agents in  $\mu'$  and any other stable matching is the same. Therefore, the set of matched agents in  $\mu''$ —an

unstable matching—and any other stable matching is the same. Since the number of men and women is equal, and every agent prefers being matched to remaining single, all the agents are matched in  $\mu'$  and  $\mu''$ . Therefore, there is a blocking pair,  $(m, w)$ , in  $\mu''$  such that both the agents are matched to someone in  $\mu''$ .  $\square$

Consider a blocking pair,  $(m, w)$ , in  $\mu''$ . The fact that  $m, w$  are matched and form a blocking pair means that  $\rho_w(m) \leq r'$  and  $\rho_m(w) \leq r'$ . Therefore,  $\mu''$  is not  $r'$ -stable, a contradiction.  $\square$

Notice that we used the fact that the number of men and women is equal only in the last step of the proof. We can, therefore, relax that requirement to define the notion of a regular market.

**Definition 6** For an instance of a marriage market, let  $S$  be the set of agents matched in any stable matching. We say that an instance of a matching market is regular if, for every unstable matching, there is at least one blocking pair, such that both the agents of the blocking pair belong to  $S$ .

It is easy to see that, if the market is regular, then the outcome is a Rawlsian stable matching, as the reasoning in Claim 1 shows.

While the condition of regular market in Definition 6 may be hard to verify at times, it is only a sufficient condition for obtaining a Rawlsian matching. Moreover, as mentioned in the Introduction, large markets with unequal numbers of men and women essentially have a unique stable matching as shown by Ashlagi et al. (2017).

**Discussion:** Intuitively, ours is a greedy algorithm: it produces a  $k$ -stable submatching for each  $k$ , and  $k$  is incremented if the outcome is unstable with respect to the original preferences. This is also the reason why it produces a Rawlsian stable matching. A reader may wonder, then, if the sole goal is to produce a Rawlsian stable matching, why not simply run a DAA for each  $k$  instead of our algorithm, which is ostensibly more complicated.<sup>8</sup> There may, indeed, be several ways to produce a  $k$ -stable submatching if the goal were to obtain a Rawlsian stable matching this way. However, the motivation of our exercise was to construct a “DAA-like” procedure wherein both sides of the market propose in each round. This was, in part, motivated by the remark due to Knuth (1997) mentioned in the introduction. Apriori, it is unclear how one can devise such a procedure and ensure that it does not cycle. We show that this is, indeed, feasible. As a *byproduct*, we obtain that the outcome is a Rawlsian stable matching.

## 4 Some examples

**Example 3** Here is a slightly more complicated example than Example 1 (Table 2).  $\blacksquare$

Notice that  $m_4$  and  $w_1$  appear as each other's top preference. Therefore, they will be matched in the first round, and that match will never be broken. We will, therefore, exclude them in writing the  $S_{i,v,k}$ 's and the  $T_{i,v,k}$ 's for each round.

Round 1 In the first round,

<sup>8</sup> We thank an anonymous referee for this suggestion.

**Table 2** Preferences

$\succ_{m_1}$	$\succ_{m_2}$	$\succ_{m_3}$	$\succ_{m_4}$	$\succ_{w_1}$	$\succ_{w_2}$	$\succ_{w_3}$	$\succ_{w_4}$
$w_2$	$w_3$	$w_1$	$w_1$	$m_4$	$m_3$	$m_4$	$m_3$
$w_3$	$w_4$	$w_3$	$w_4$	$m_1$	$m_4$	$m_1$	$m_2$
$w_1$	$w_1$	$w_4$	$w_2$	$m_3$	$m_1$	$m_2$	$m_4$
$w_4$	$w_2$	$w_2$	$w_3$	$m_2$	$m_2$	$m_3$	$m_1$

**Table 3** Table for  $S_i, T_i$  and  $\alpha_{i,V,k}$  for Round 2

Agent	$S_i$	$T_i$	$\alpha_{i,V,k}$
$m_1$	$w_2, w_3$	$w_3$	$w_3$
$m_2$	$w_3, w_4$	$w_4$	$w_4$
$m_3$	$w_1, w_3$		
$w_2$	$m_3, m_4$		
$w_3$	$m_4, m_1$	$m_1$	$m_1$
$w_4$	$m_3, m_2$	$m_2$	$m_2$

**Table 4** Table for  $S_i, T_i$  and  $\alpha_{i,V,k}$  for Round 3

Agent	$S_i$	$T_i$	$\alpha_{i,V,k}$
$m_1$	$w_2, w_3$	$w_2, w_3$	$w_2$
$m_2$	$w_3, w_4$	$w_4$	
$m_3$	$w_1, w_3, w_4$	$w_4$	$w_4$
$w_2$	$m_3, m_4, m_1$	$m_1$	$m_1$
$w_3$	$m_4, m_1$	$m_1$	
$w_4$	$m_3, m_2$	$m_3, m_2$	$m_3$

$$T_i = \emptyset, \forall i \in \{m_1, m_2, m_3, w_2, w_3, w_4\}, T_{m_4} = \{w_1\}, T_{w_1} = \{m_4\}$$

and  $\mu = \{(m_4, w_1)\}$ .

At this point, we set  $k = 2$ .

Round 2 Let us present a table for  $S_i, T_i$  and  $\mu_i$  for each agent in this round (Table 3).

Therefore, at the end of this round, agents  $m_3$  and  $w_2$  are unmatched. The allocation

$$\mu = \{(m_1, w_3), (m_2, w_4), (m_4, w_1), (m_3, m_3), (w_2, w_2)\}$$

is 2-stable but it is not stable; it can be blocked by  $(m_1, w_2)$ . Hence, we proceed to  $k = 3$ .

Round 3 Let us look at a similar table as above in this round (Table 4).

**Table 5** Preferences

$\succ_{m_1}$	$\succ_{m_2}$	$\succ_{m_3}$	$\succ_{m_4}$	$\succ_{w_1}$	$\succ_{w_2}$	$\succ_{w_3}$	$\succ_{w_4}$
$w_1$	$w_2$	$w_3$	$w_4$	$m_4$	$m_3$	$m_2$	$m_1$
$w_2$	$w_1$	$w_4$	$w_3$	$m_3$	$m_4$	$m_1$	$m_2$
$w_3$	$w_4$	$w_1$	$w_2$	$m_2$	$m_1$	$m_4$	$m_3$
$w_4$	$w_3$	$w_2$	$w_1$	$m_1$	$m_2$	$m_3$	$m_4$

Therefore, at the end of this round, agents  $m_2$  and  $w_3$  are unmatched. The allocation

$$\mu = \{(m_1, w_2), (m_3, w_4), (m_4, w_1), (m_2, m_2), (w_3, w_3)\}$$

is not 3-stable, as it can be blocked by  $(m_2, w_3)$ . At this point, we apply the RO mechanism due to Ma (1996). We start with a 3-stable submatching. Notice that  $\mu$  is a 3-stable submatching (of itself, in fact; see Definition 4), as there are no blocking pairs involving any agent who is matched. Hence, we are now ready to start the RO mechanism.

We have,  $V = \{m_1, m_3, m_4, w_1, w_2, w_3\}$  in the RO mechanism given in Algorithm 3, and  $R = \{m_2, w_3\}$ . We pick one agent randomly—say,  $m_2$ —and set  $V' = V \cup \{m_2\}$ . Notice that  $\mu$  is 3-stable with active agents in  $V'$ . Therefore,  $\mathbb{H}(\mu, V', 3)$  produces  $\mu' = \mu$ . We go back to Step 3 in the RO mechanism (Algorithm 3). The only remaining agent is  $w_3$ . Therefore, we now have  $V = A$ . It is easy to check that  $\mathbb{H}(\mu, V, 3) = \mu' := \mu \cup \{(m_2, w_3)\}$ . This is a 3-stable and a stable matching. Therefore, the algorithm terminates with the output.

$$\mu = \{(m_1, w_2), (m_2, w_3), (m_3, w_4), (m_4, w_1)\}$$

It can be checked that this is also a Rawlsian matching.

#### 4.1 Difference between the output of Algorithm 2 and the median matching and the RO Mechanism

**Example 4** The following example is due to Knuth, reproduced in Roth and Sotomayor (1990) (Table 5).

We enumerate all the stable matchings in Table 6.

The median stable matching of Teo and Sethuraman (1998) is matching either 4 or 7. The RO Mechanism due to Ma (1996) generates any matching *except* for 4, 5, 6, and 7. Our algorithm can produce any matching out of 4, 5, 6, and 7.

## 5 Computational complexity

The *Median Stable Matching* of Teo and Sethuraman (1998) is a very appealing solution concept when fairness concerns matter. However, computing the median stable

**Table 6** All the stable matchings

	$w_1$	$w_2$	$w_3$	$w_4$
1	$m_1$	$m_2$	$m_3$	$m_4$
2	$m_2$	$m_1$	$m_3$	$m_4$
3	$m_1$	$m_2$	$m_4$	$m_3$
4	$m_2$	$m_1$	$m_4$	$m_3$
5	$m_3$	$m_1$	$m_4$	$m_2$
6	$m_2$	$m_4$	$m_1$	$m_3$
7	$m_3$	$m_4$	$m_1$	$m_2$
8	$m_4$	$m_3$	$m_1$	$m_2$
9	$m_3$	$m_4$	$m_2$	$m_1$
10	$m_4$	$m_3$	$m_2$	$m_1$

matching(s) is NP-hard (Cheng 2008). On the other hand, Theorem 2 below shows that Algorithm 2 can be executed in polynomial time. Therefore, our algorithm is a practically feasible alternative in markets where fairness considerations are important.

**Theorem 2** *Algorithm 2 can be executed in polynomial time.*

**Proof** We need, at most,  $N$  rounds to end the algorithm. Each round has an execution of procedure  $\mathbb{H}$ , potentially  $N$  number of times.  $\mathbb{H}$  involves computing  $T_{i,V,k}$  to construct the graph  $G$ .  $T_{i,V,k}$  can be computed in  $\mathcal{O}(N^3)$ . Following the construction of the graph, we need to enumerate the cycles, which is the same as enumerating all the strongly connected components. This can be done using Tarjan’s algorithm in  $\mathcal{O}(|V| + |E|) = \mathcal{O}(N)$ . The elimination of cycles can be repeated within a particular execution of  $\mathbb{H}$ . Every iteration removes at least two agents by forming a match between them. Therefore, steps 4 – 7 have a complexity of  $\mathcal{O}(n^4)$ . Checking for stability or  $k$ -stability has a  $\mathcal{O}(N^2)$  complexity. Moreover, if the output is not  $k$ -stable, we run the RO mechanism that uses a  $k$ -stable submatching as an input. Such submatching can be obtained by removing any agent involved in a blocking pair with a  $\mathcal{O}(N^2)$  complexity. After that, the RO mechanism can be run in polynomial time as mentioned in Cheng (2016).  $\square$

## 6 Conclusion

Motivated by the asymmetry in DAA, we have proposed an alternative algorithm in the classical two-sided marriage market. The algorithm has the flavor of a DAA but attempts to correct the asymmetry in the outcomes by allowing both sides of the market to make proposals. We show that we can still retain the simplicity of the DAA in constructing such a procedure and obtain a stable matching. More importantly, allowing both sides to make proposals results in a Rawlsian stable matching—a well-established criterion of fairness in numerous settings—with certainty in regular markets, and with probability close to one in large markets. Finally, practical con-

siderations are paramount when one proposes a new algorithm in these contexts. On this matter, the fact that the algorithm can be executed in polynomial time makes it a compelling alternative in markets where the designer considers the fairness of the procedure and the outcomes as important.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## Appendix

### A Proof of the RO Mechanism

**Claim 2** *If  $\mu$  is  $k$ -stable with active agents in  $V$ , then the graph of  $\mathbb{H}(\mu, V, k)$  has the following properties.*

1. Any agent with  $\mu(i) \neq i$  is involved in a cycle of length two,  $i \rightarrow \mu(i) \rightarrow i$ . And,
2. Any agent with  $\mu(i) = i$  has an outdegree of 0.

**Proof** Consider any agent  $i$  such that  $\mu(i) \neq i$ . Then,  $\mu(i) \in T_{i,V,k}$ . If  $\alpha_{i,V,k} \neq \mu(i)$ , then  $\exists j$  such that  $j \succ_i^k \mu(i)$  and  $i \succ_j^k \mu(j)$ . Therefore,  $(i, j)$  is a  $\succ^k$  blocking pair. Therefore,  $\mu$  is not  $k$ -stable, a contradiction. Therefore,  $i \rightarrow \mu(i) \rightarrow i$  in the graph of  $\mathbb{H}(\mu, V, k)$ .

For any agent  $i$  such that  $\mu(i) = i$ , if  $T_{i,V,k} \neq \emptyset$ , then  $\exists j$  such that  $j \succ_i^k i$  and  $i \succ_j^k \mu(j)$ . But then,  $(i, j)$  constitute a  $\succ^k$  blocking pair for  $\mu$ , i.e.,  $\mu$  is not  $k$ -stable; a contradiction. □

**Lemma 2** *Algorithm 3 produces a  $k$ -stable matching.*

**Proof** Consider a matching  $\mu'$  that is  $k$ -stable with to active agents in  $V$ . Consider some man, wlog,  $m \in R := A \setminus V$ . Then, in step 7 of Algorithm 3, we draw a graph  $G$  with nodes  $V' := V \cup \{m\}$ , and edges as described in map  $\mathbb{H}$ . We have the following three cases in the structure of this graph, denoted by  $G = (V', E)$ .

**Case 1.**  $T_{m,V',k} = \emptyset$ .

**Case 2.**  $T_{m,V',k} \neq \emptyset$ , and  $(m, w), (w, m) \in E$  such that  $\mu'(w) = w$ .

**Case 3.**  $T_{m,V',k} \neq \emptyset$  and  $(m, w), (w, m) \in E$  with  $\mu'(w) = m' \in M$ .

The reason why we can have at most these three cases follows from Claim 2 which says that the graph of  $\mathbb{H}(\mu', V, k)$  contains only two cycles of the matched agents in  $\mu'$ . If  $T_{m,V',k} \neq \emptyset$ , then  $m$  has an outdegree of 1. Moreover, if  $(m, w) \in E$ , then  $m \succ_w \mu'(w)$ . Since  $\mu'$  was  $k$ -stable with active agents in  $V$ ,  $\nexists i \in V$  such that  $i \succ_w \mu'(w)$ . Therefore,  $(w, m) \in E$ .



In Case 1, we obtain  $\mu'' = \mu'$  that is  $k$ -stable, and hence we reach step 8 of Algorithm 3.

In Case 2, we obtain  $\mu'' = \mu' \cup \{(m, w)\}$ . Moreover,  $\mu''$  is  $k$ -stable. If not, a blocking pair must involve  $m$ . Let  $(m, w')$  be such a blocking pair. But then,  $w' \succ_m w$ . Therefore,  $(m, w) \notin E$ , a contradiction.

In Case 3,  $m$  displaces some  $m'$  that was matched in  $\mu'$ . The difference between  $\mu'$  and  $\mu''$  is the following:

1.  $m'$  is single in  $\mu''$  while he is matched in  $\mu'$ .
2.  $(m, \mu'(m'))$  are matched in  $\mu''$ .

If  $\mu''$  is  $k$ -stable, we reach step 8.

**Claim 3**  $\mu''$  is  $k$ -stable with active agents in  $V \setminus \{m'\}$ .

**Proof** Suppose there is some blocking pair  $(m'', w'')$ . Suppose that  $m'' \neq m$ . First,  $w'' \neq \mu'(m')$ , as that would mean that  $m'' \succ_{\mu'(m')} m$ . Therefore,  $m$  could not have matched  $w''$  with  $\mathbb{H}$ . If both  $(m'', w'')$  were single in  $\mu'$ , then  $\mu'$  could not have been  $k$ -stable. If one or both of them were matched to someone in  $\mu'$ , that match remains unchanged in  $\mu''$  since  $w'' \neq \mu'(m')$  and  $m'' \neq m$ . Therefore,  $\mu'$  could not have been  $k$ -stable with active agents in  $V$ . Therefore, any blocking pair in  $\mu''$  with active agents in  $V \setminus \{m'\}$  must involve  $m$ . Therefore,  $w'' \succ_m \mu'(m')$  and  $m \succ_{w''} \mu'(w'')$ . But then,  $(m, \mu'(m')) \notin E$  as  $m$  would point to his favorite mutually proposing agent, a contradiction.  $\square$

Therefore, if  $\mu''$  is not  $k$ -stable with active agents in  $V'$ , the only possible blocking pairs it can involve contain  $m'$ . At this point, the algorithm returns to step 5. We draw a graph of  $\mathbb{H}(\mu'', V', k)$ . Similar to before, the graph has three possible cases with  $m$  replaced by  $m'$ . We can continue this way to obtain a sequence  $\mu' =: \mu_0, \dots, \mu_n$ , each involving one and only man being displaced in the previous match. Every such operation strictly improves one woman, and, at no point in the sequence does any woman become worse off. Therefore, this sequence must terminate in finite time. As a result,  $\mu_n$  is  $k$ -stable with active agents in  $V'$ , i.e., we reach Step 8. At this point, we include another agent (if available) from among  $R = A \setminus V'$ . Therefore, when  $R = \emptyset$ , we obtain a  $k$ -stable matching with active agents in  $A$ .  $\square$

## References

- Aldershof B, Carducci OM, Lorenc DC (1999) Refined inequalities for stable marriage. *Constraints* 4(3):281–292
- Ashlagi I, Kanoria Y, Leshno JD (2017) Unbalanced random matching markets: the stark effect of competition. *J Polit Econ* 125(1):69–98
- Cheng C, McDermid E, Suzuki I (2011) Center stable matchings and centers of cover graphs of distributive lattices. *Automata, Languages and Programming*, pp 678–689
- Cheng CT (2008) The generalized median stable matchings: finding them is not that easy. In *LATIN 2008: theoretical informatics: 8th Latin American Symposium, Búzios, Brazil, April 7–11, 2008. Proceedings* 8, pp 568–579. Springer
- Cheng CT (2010) Understanding the generalized median stable matchings. *Algorithmica* 58(1):34–51
- Cheng CT (2016) On the stable matchings that can be reached when the agents go marching in one by one. *SIAM J Discret Math* 30(4):2047–2063

- Dworczak P (2021) Deferred acceptance with compensation chains. *Oper Res* 62(2):456–468
- Gale D, Shapley LS (1962) College admissions and the stability of marriage. *Am Math Mon* 69(1):9–15
- Gusfield D (1987) Three fast algorithms for four problems in stable marriage. *SIAM J Comput* 16(1):111–128
- Klaus B, Klijn F (2006) Procedurally fair and stable matching. *Econ Theor* 27(2):431–447
- Knuth D (1976) *Marriages stable*. universit  de montr al press, translated as “stable marriage and its relation to other combinatorial problems,”. In: CRM Proceedings and Lecture Notes, American Mathematical Society
- Knuth DE (1997) *Stable marriage and its relation to other combinatorial problems: An introduction to the mathematical analysis of algorithms*, vol 10. American Mathematical Soc
- Kuvalekar A (2014) A fair procedure in a marriage market. Available at SSRN 2688630
- Ma J (1996) On randomized matching mechanisms. *Econ Theor* 8(2):377–381
- Manjunath V, Turhan B (2016) Two school systems, one district: What to do when a unified admissions process is impossible. *Games Econom Behav* 95:25–40
- Masarani F, Gokturk SS (1989) On the existence of fair matching algorithms. *Theor Decis* 26(3):305–322
- Romero-Medina A (2001) Sex-equal stable matchings. *Theor Decis* 50(3):197–212
- Romero-Medina A (2005) Equitable selection in bilateral matching markets. *Theor Decis* 58(3):305–324
- Roth AE, Sotomayor M (1990) *Two-sided matching: a study in game-theoretic modeling and analysis*. Econometric Society Monographs. Cambridge University Press, Cambridge
- Roth AE, Vate JHV (1990) Random paths to stability in two-sided matching. *Econom J Econom Soc* 58:1475–1480
- Sethuraman J, Teo C-P, Qian L (2006) Many-to-one stable matching: geometry and fairness. *Math Oper Res* 31(3):581–596
- Teo C-P, Sethuraman J (1998) The geometry of fractional stable matchings and its applications. *Math Oper Res* 23(4):874–891

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.