

Analysis of Stochastic Matching Markets

Péter Biró^{1,*} and Gethin Norman²

¹ Institute of Economics, Hungarian Academy of Sciences, H-1112, Budaörsi út 45,
Budapest, Hungary.

Email: `birop@econ.core.hu`.

² School of Computing Science, University of Glasgow, Glasgow G12 8QQ, UK.

Email: `gethin.norman@glasgow.ac.uk`.

Abstract. Suppose that the agents of a matching market contact each other randomly and form new pairs if it is in their interest. Does such a process always converge to a stable matching if one exists? If so, how quickly? Are some stable matchings more likely to be obtained by this process than others? In this paper we are going to provide answers to these and similar questions, posed by economists and computer scientists. In the first part of the paper we give an alternative proof for the theorems by Diamantoudi *et al.* and Inarra *et al.* which imply that the corresponding stochastic processes are absorbing Markov chains. The second part of the paper proposes new techniques to analyse the behaviour of matching markets. We introduce the Stable Marriage and Stable Roommates Automaton and show how the probabilistic model checking tool PRISM may be used to predict the outcomes of stochastic interactions between myopic agents. In particular, we demonstrate how one can calculate the probabilities of reaching different matchings in a decentralised market and determine the expected convergence time of the stochastic process concerned. We illustrate the usage of this technique by studying some well-known marriage and roommates instances and randomly generated instances.

1 Introduction

The Stable Roommates problem (SR) is a classical combinatorial problem that has been studied extensively in the literature, see e.g. [21]. An instance I of SR contains an undirected graph $G(V, E)$, where $V = \{v_1, \dots, v_n\}$ and $m = |E(G)|$. We refer to G as the *underlying graph* of I , and we interchangeably refer to the vertices of G as the *agents*. If (v_i, v_j) is an edge in $E(G)$, then we say that v_i and v_j find each other *acceptable*. A *matching* is a set of independent edges in the graph.³ Each agent v_i has a linear order $>_{v_i}$ over her acceptable partners, where $v_k >_{v_i} v_j$ means that v_i prefers v_k to v_j . Let $M(v_i)$ denote the *partner*

* Supported by EPSRC grants EP/E011993/1, by OTKA grant K69027 and by the Hungarian Academy of Sciences under its Momentum Programme (LD-004/2010).

³ Note that by this definition of matching we restrict our attention to the individually rational matchings.

of v_i in a given matching M . An edge (v_i, v_j) is said to be *blocking* with respect to M if (i) either v_i is unmatched in M or prefers v_j to $M(v_i)$, and (ii) either v_j is unmatched in M or prefers v_i to $M(v_j)$. A matching is called *stable* if it admits no blocking edge. If G is bipartite, then the problem of finding a stable matching is called the Stable Marriage problem (SM). In this case, if the graph is $G(U, W, E)$, then we refer to $U = \{m_1, \dots, m_{n_1}\}$ and $W = \{w_1, \dots, w_{n_2}\}$ as the sets of men and women, respectively.

Note that both the Stable Roommates and the Stable Marriage problems can be seen as NTU-games (hedonic coalition formation games, in particular), since for any SR or SM instance the set of stable matchings coincide with the core of the corresponding game. For further details, see for example the celebrated book by Roth and Sotomayor [42].

Gale and Shapley [20] give a linear time algorithm that finds a stable matching for any instance of SM, while also illustrating an instance of SR that does not admit a stable matching (i.e., which is not *solvable*). Irving [25] gives a linear time algorithm that, for any instance of SR, finds a stable matching or reports that none exists. Both algorithms assume that the preference lists are complete (i.e., the graph G is complete), although it is straightforward to extend the algorithms to incomplete lists [21].

Suppose that we are given a SR instance I with underlying graph G . For a matching M , if a pair (v_i, v_j) is blocking, then we may *satisfy this blocking pair* and get a new matching $M^{(v_i, v_j)}$, where $(v_i, v_j) \in M^{(v_i, v_j)}$ and for each $w \in \{v_i, v_j\}$, if w is matched in M , then $M(w)$ is unmatched in $M^{(v_i, v_j)}$. Roth and Vande Vate [43] prove that, given an instance of SM, starting from any unstable matching we can always obtain a stable matching by successively satisfying blocking pairs.⁴ Diamantoudi *et al.* [18] show that a similar result holds for the roommates problem, namely, for a given instance of SR that admits a stable matching and starting from any unstable matching, one can obtain a stable matching by successively satisfying blocking pairs. This essentially means that the corresponding stochastic processes (to be defined in Section 3) are *absorbing Markov chains* (for more details of these stochastic processes see, e.g. Chapter 3 of [27]). Since there are only finitely many matchings in any instance, the result of Roth and Vande Vate implies that, starting from an arbitrary matching, the process of allowing randomly chosen blocking pairs to match will converge to a stable matching with probability one.

The proof of Roth and Vande Vate is based on the following idea. Suppose that we have a stable matching for an instance of SM and we add a new agent

⁴ Note that this question was originally proposed by Knuth [31] (Problem 8 from his twelve famous research problems) in a slightly different setting. In his case, the set of possible matchings was restricted to the complete matchings (as all the preference lists were supposed to be complete), and whenever a blocking pair was satisfied the left-alone agents formed a new pair immediately. The above described transition from a complete matching to another one was called an *interchange*. Knuth asked whether, given an instance of SM and a starting matching M , there always exist a sequence of interchanges from M to some stable matching? Tamura [45], and independently Tan and Su [49], answered this question negatively by giving counterexamples.

to the market, then there is a natural proposal-rejection sequence (described in Section 3) that leads to a stable matching for the extended instance. If, we start with the empty matching and run this incremental algorithm, then the resulting stable matching will depend on the order in which the agents arrive. This is called the *random order mechanism*. By assuming that each order is equally likely, we may calculate the probability of each stable matching being obtained. Ma [37] carried out this calculation for an instance, originally suggested by Knuth [31], and observed that not all stable matchings can be reached by this mechanism and there is a higher probability of reaching some stable matchings over others (although his calculation was not entirely correct as [29] pointed out).⁵ In this paper we will also study this instance (Example 2 in Section 3) with respect to a different stochastic process.

We may suppose that all agents are present in the market and, starting with the empty matching, the blocking pairs to be satisfied are chosen randomly (with equal probability in each step). In this case, every stable matching can be reached with positive probability (since we may satisfy all pairs involved in this matching at the beginning of the process), but still, as we will illustrate in Section 3, some stable matchings can be more likely to occur than others.

There is also a growing literature concerning stable roommates problems that may not admit stable solutions. Tan [47] shows that a *stable half-matching* always exists for any given instance of SR. A *half-matching* is a weight function $h : E(G) \rightarrow \{0, \frac{1}{2}, 1\}$ such that $\sum_{v_i \in e} h(e) \leq 1$ for each vertex v_i . A half-matching is said to be *stable* if, for each edge $(v_i, v_j) \in E(G)$, one of its vertices, say v_i satisfies $\sum_{(v_i, v_k) v_k \geq v_i v_j} h((v_i, v_k)) = 1$, (otherwise the edge (v_i, v_j) is said to be *blocking*).⁶ Note that if $h : E(G) \rightarrow \{0, 1\}$ is stable, then h corresponds to a stable matching.⁷ Tan [47] showed that a stable half-matching must consist of half-weighted cycles and one-weighted independent edges. Moreover, the set of half-weighted cycles of odd length (that we sometime just refer to as half-weighted

⁵ An explanation for the first observation is the result of Blum and Rothblum [12] which demonstrates that, when using the Roth-Vande Vate algorithm, the last agent to arrive always gets their best stable partner. (This observation directly follows from the results of Blum *et al.* [11]. An alternative proof and an extension of that result for roommates problems was given by Biró *et al.* [6].) Hence, a stable matching in which nobody gets their best partner cannot be obtained by this mechanism.

⁶ The weight of an edge in a half-matching can be seen as the intensity of a cooperation between the two corresponding agents. The notion of half-matching means that each agent can be involved in at most one cooperation with full intensity or in at most two cooperations with half intensities. Stability means that for each pair whose members are not cooperating with full intensity, either of the agents is not interested in increasing the intensity of this cooperation, as she must have filled her capacity with better partners.

⁷ The existence of a half-matching may be proved by the Lemma of Scarf [44], as Aharoni and Fleiner [2] demonstrate. The notion of *stable fractional matching* (or *fractional core*) is an extension of stable half-matching that may be defined for more general matching problems (or NTU-games) as well, see more on this theory in a recent paper by Biró and Fleiner [7].

odd cycles) and the set of unmatched agents are the same in every stable half-matching for a given instance of SR. Tan and Hsueh [48] give a polynomial time algorithm to find a stable half-matching. This algorithm is, in fact, a generalised version of the algorithm of Roth and Vande Vate (a detailed description of this is given in Section 3). Further descriptions on stable half-matchings and the Tan-Hsueh algorithm can be found in [6].

Inarra *et al.* [23] define an *h-stable matching* M relative to a stable half-matching h as follows⁸. Let M contain every edge that has weight 1 in h , every second edge from each even half-weighted cycle of h (if there were any half-weighted cycle of even length), and k (disjoint) edges from each half-weighted odd cycle of length $2k + 1$ in h .⁹ They show that, starting from an arbitrary matching, one can get an *h-stable matching* by successively satisfying blocking pairs for a given instance of SR. Note that for every solvable instance of SR the set of *h-stable matchings* is equivalent to the set of stable matchings, thus the above result generalises the theorem of Diamantoudi *et al.* [18]. In Section 2, we give an alternative short proof for the theorem of Inarra *et al.* [23] by using the Tan-Hsueh algorithm.

In another paper, Inarra *et al.* [24] define the *absorbing sets* for an instance of SR as follows. Each absorbing set consists of matchings that are reachable from one another by successively satisfying blocking edges, but no other matching can be reached from this set by satisfying a blocking edge. These are in fact the *ergodic sets* of the corresponding Markov chain (see e.g. [27]), and a matching M is in an ergodic set if and only if the limit probability of M , starting from the empty matching, is positive. Moreover, Klaus *et al.* [30] prove that the absorbing sets consist of exactly those matchings that have positive probabilities in the limit distribution of a stochastic process where, starting from any matching, the agents make mistakes with small probabilities in their myopic blocking decisions. They called this process *perturbed blocking dynamics*. Similar stochastic systems have been studied for the Stable Marriage problem in the context of network formations by Jackson and Watts [26].

Ackermann *et al.* [1] study the convergence time of the stochastic processes occurring from stable marriage problems. They refer to the stochastic process, where in each step a blocking pair is chosen uniformly at random and satisfied, as the *random better response dynamics*. They demonstrate that, although the process converges to a stable matching, the expected convergence time is exponential for a family of SM instances. Our experiments conducted for the above family of instances confirm this finding, as we describe in Section 3. However, we also demonstrate that this behaviour is unexpected in an average market,

⁸ As Tan [46,47] referred to stable half-matchings as *stable partitions*, Inarra *et al.* [23] also used this terminology, and so instead of *h-stable matchings* they called the same notion *P-stable matchings*, relative to a stable partition P . We decided to change the original terminology because we believe that the concept of half-matchings is well-known in graph theory, see more explanation about this issue in [6].

⁹ This concept was originally proposed by Tan [46] as a method to find a matching as large as possible that is stable for the matched agents in an unsolvable instance.

since for the randomly generated instances the expected convergence time is significantly smaller.

The dynamics of matching markets have also been in focus in some recent engineering papers on P2P systems, see, e.g, [38] for an overview. In particular, Lebedev *et al.* [36] show that the convergence is fast for systems, modelled with SR instances, where the preferences are *acyclic*, i.e., the preferences are derived from some global rank function on the pairs. This is a realistic assumption in the case of some real P2P networks. Finally, Arcaute and Vassilvitskii [3] and Hoefer [22] studied similar stochastic market processes with the extra feature of an underlying social network that dynamically determine the accessible partners and the so-called locally stable matchings.

In our analysis we also suppose that the stochastic process follows the random better response dynamics. Does this model give a good description of decentralised matching markets? There are two very recent experimental studies that provide some positive evidence for that. Echenique and Yariv [19] conducted experimental tests with students who were allowed to make and accept proposals in a decentralised manner. Most of the outcomes in these games were stable matchings and when several stable matchings were possible then they recorded their distribution as outcomes. In particular, they found that, when the market had three stable matchings, then the median one emerged as the modal empirical outcome. They showed with simulations ([19, Section 7.1]) that this distribution of stable matchings was relatively close to those that the stochastic model with random better responses would predict. Pais *et al.* [40] received similar results regarding the likeliness of obtaining the median stable matching in their experiments.

Boudreau has also studied the random better response dynamics (referred to as *randomized tâtonnement process*) in several recent papers [14,15,16]. In [15] he showed that the median stable matching is not necessarily the most likely one for every market, which is in contrast with the above mentioned experimental results ([19,40]). While, in [14] Boudreau studied the effect of intercorrelated preferences on the expected convergence time. His simulations show that if preferences are negatively intercorrelated (i.e., agents prefer those on other side of the market who do not prefer them) then the path to stability tend to be (exponentially) long. This finding back the theoretical results of Ackermann *et al.* [1] on the exponential expected convergence time of ‘uncoordinated’ instances, that we also confirmed with simulations (described in Section 3). Boudreau [15] also demonstrate that the most likely outcome of such a decentralised market might not be the most *efficient* stable matching, which is defined as the minimum-choice-count matching (i.e., a matching, where the sum of the rankings that agents give their partners is minimal). In a very recent work [16] he identifies two measurable factors that do have significant, non-linear impacts on the likelihood that a market’s most likely outcome will be (in)efficient: the number of stable matchings that the market possesses and the level of correlation among the preferences of each side of the market.

To summarise, the contribution of this paper is the following. In Section 2 we give an alternative proof for the theorems of Diamantoudi *et al.* [18] and Inarra *et al.* [23]. This new proof, which is based on the Tan-Hsueh algorithm, is not only shorter and simpler than the originals, but also provides upper bounds on the number of steps needed to reach a stable (or h -stable) matching. In Section 3 we define the Stable Marriage and Stable Roommates Automata and then we demonstrate how the probabilistic model checker PRISM [35,51] can be used to analyse and compare the performance of different instances. In particular, we study two well-known SM instances, a SR instance and then present a case study involving structured and random SM instances. We believe that this approach will also have applications in the study the interaction of agents in real markets and networks for more complex settings. We describe some future research directions in Section 4.

2 Convergence to stability, an alternative proof

In this section we describe the Roth-Vande Vate and the Tan-Hsueh algorithms. We use the latter to give an alternative proof for the theorems of Diamantoudi *et al.* [18] and Inarra *et al.* [23]. That is, we show that starting from an arbitrary matching of a solvable SR instance one can always find a stable matching by successively satisfying blocking pairs; and that starting from an arbitrary matching of an instance of SR (solvable or unsolvable) one can always find an h -stable matching by successively satisfying blocking pairs. Note that these theorems were the main results of the above papers. Our proof is much shorter and it gives upper bounds for the number of blocking pairs that need to be satisfied to obtain a stable (or h -stable) matching. Also, it shows that the argument of Roth and Vande Vate for the marriage case can be extended for the roommates case in a natural way.

The Roth-Vande Vate algorithm. Suppose that we are given an instance I of SM together with a matching $M_0 = \{(m_1, w_1), \dots, (m_k, w_k)\}$. We shall show that we can reach a stable matching by successively satisfying blocking pairs. A variant of the Roth-Vande Vate algorithm works as follows.

During the procedure we gradually extend a set $S \subseteq (U \cup W)$ and a matching M_S that is stable in S . Initially let $S = \emptyset$ and $M_S = \emptyset$. For each index i ($i=1, \dots, k$), if $M_S \cup \{(m_i, w_i)\}$ is stable in $S \cup \{m_i, w_i\}$, then let $M_{S'} = M_S \cup \{(m_i, w_i)\}$ and $S' = S \cup \{m_i, w_i\}$ (i.e. we simply enlarge both S and M_S with a new pair). Otherwise we add m_i and w_i to S one by one as follows.

Without loss of generality suppose that m_i is involved in a blocking pair with an agent of S with respect to matching $M_S \cup \{(m_i, w_i)\}$, let w_{i_1} be the woman who is the best blocking partner of m_i and let $S' = S \cup \{m_i\}$. If w_{i_1} is unmatched in M_S , then $M_{S'} = M_S \cup \{(m_i, w_{i_1})\}$ is a stable matching in S' . Otherwise, let $m_{i_1} = M_S(w_{i_1})$ and $M_{S' \setminus \{m_{i_1}\}} = (M_S \setminus \{(m_{i_1}, w_{i_1})\}) \cup \{(m_i, w_{i_1})\}$ is stable for $S' \setminus \{m_{i_1}\}$. Now we let m_{i_1} re-enter the market. If m_{i_1} is not involved in any blocking pair, then $M_{S' \setminus \{m_{i_1}\}}$ is stable for S' . Otherwise we satisfy the best blocking pair m_{i_1} is involved in according to his preferences, and so on.

This process must terminate after satisfying at most m blocking pairs, since no woman ever receives a worse partner. We can also add w_i in a similar manner, reversing the role of men and women. (Note that if m_i was not involved in a blocking pair with an agent of S with respect to matching $M_S \cup \{(m_i, w_i)\}$ then w_i must have been involved in a blocking pair, so we start by adding w_i to S first, followed by m_i .)

After processing all pairs of M_0 , we add the remaining agents one by one in the same way. Therefore, we obtain the sequence of blocking pairs that we need to satisfy to reach a stable matching starting from M_0 . Since we never satisfy a pair twice when adding a new agent to S , it follows that the number of steps in the path to stability is at most mn .¹⁰

The Tan-Hsueh algorithm. The Tan-Hsueh algorithm deals with SR instances (rather than SM instances) and stable half-matchings (rather than stable matchings), and there is no starting matching M_0 . But otherwise it is based on the same idea as the Roth-Vande Vate algorithm: we gradually extend a set $S \subseteq V(G)$ and we restore the stability of a half-matching h_S in S .¹¹

Suppose that we are given an instance I of SR with an underlying graph $G(V, E)$. Let S denote a set of agents and let h_S denote a half-matching of the (entire) graph. Initially let $S = \emptyset$ and let $h_S(e) = 0$ for each $e \in E(G)$. Suppose that after adding k agents we have $S = \{v_1, v_2, \dots, v_k\}$ with a corresponding stable half-matching h_S on the subgraph induced by S , where each half-weighted cycle has odd length. Let $S' = S \cup \{v_{k+1}\}$. Now we describe how we can construct the new stable half-matching $h_{S'}$ in S' .

If v_{k+1} is not involved in any blocking pair in S' , then h_S remains stable in S' , obviously. Otherwise let v_j be the best blocking partner of v_{k+1} in S' . If v_j is unmatched (i.e., not matched and not covered by a half-weighted cycle either), then by setting $h_{S'}((v_{k+1}, v_j)) = 1$ and $h_{S'}(e) = h_S(e)$ for every other edge we obtain a new stable half-matching in S' . If v_j is covered by a half-weighted odd cycle, say by $(v_{c_1}, v_{c_2}, \dots, v_{c_{2l+1}})$ where $v_j = v_{c_1}$, then by setting $h_{S'}((v_{k+1}, v_j)) = 1$, $h_{S'}((v_{c_{2i}}, v_{c_{2i+1}})) = 1$ for $i = 1, \dots, l$, $h_{S'}((v_{c_{2i-1}}, v_{c_{2i}})) = 0$ for $i = 1, \dots, l$ and $h_{S'}((v_{c_{2l+1}}, v_{c_1})) = 0$ we obtain a new stable half-matching. The last case is when v_j is matched in h_S to an agent, say v_{a_1} . By setting $h_{S' \setminus \{v_{a_1}\}}((v_{k+1}, v_j)) = 1$ and $h_{S' \setminus \{v_{a_1}\}}((v_j, v_{a_1})) = 0$ we obtain a half-matching that is stable in $S' \setminus \{v_{a_1}\}$. (Note that each of the half-weighted cycles in the new stable half-matching has odd length.) Now, we restart the process with v_{a_1} .

In contrast with the SM context, it is possible that the latter case happens every time and the above process never ends, since in a sequence $v_{a_1}, v_{b_1}, \dots, v_{a_l}, v_{b_l}, v_{b_l}$ may be the same as v_{a_1} . Tan and Hsueh [48] showed that, if such a repe-

¹⁰ This version of the Roth-Vande Vate algorithm has been described by Ma [37]. Note that it slightly differs from the original method described in [43], but the difference is not substantial. In the original version, if no blocking pair involves an agent from S and another from outside S then a blocking pair involving two agents from outside S is satisfied and both agents are added to S .

¹¹ For any bipartite graph the Tan-Hsueh algorithm is identical to the Roth-Vande Vate algorithm if the starting matching of the latter algorithm is \emptyset .

tion occurs, then a subset of these agents (always the same agents after the repetition) will be involved in a never ending cycling and we can form a new half-weighted odd cycle on the corresponding edges resulting in a new stable half-matching $h_{S'}$ in S' .

In each phase of this incremental algorithm (when an agent is added to the market) we obtain a new stable half-matching in $O(m)$ steps, so the algorithm terminates in $O(mn)$ steps in total, where each step means that one blocking pair is satisfied.

Alternative proofs of [23] and [18]. Modifying the Tan-Hsueh algorithm slightly (with h -stable matchings rather than with stable half-matchings), we can obtain an alternative proof for the following theorem of Inarra *et al.* [23], with an upper bound on the number of steps needed to reach an h -stable matching.

Theorem 1. *Suppose that we are given an instance of SR and a matching M_0 , then one can always reach an h -stable matching starting from M_0 by successively satisfying at most mn blocking pairs.*

Proof. Let $M_0 = \{(v_1, v_2), (v_3, v_4), \dots, (v_{2k-1}, v_{2k})\}$. Just as in the proof of Roth and Vande Vate, we gradually extend a set $S \subseteq V(G)$ and a matching M_S in S , where initially $S = \emptyset$ and $M_S = \emptyset$.

Suppose that $S = \{v_1, v_2, \dots, v_{2i}\}$ and M_S is a h_S -stable matching relative to a stable half-matching h_S in the sub-instance induced by S . Recall that each edge e of weight 1 in h_S is represented in M_S and each half-weighted odd cycle $C = (v_{c_1}, v_{c_2}, \dots, v_{c_{2l+1}})$ is represented by l disjoint edges of C in M_S . Consider a half-matching h^* in $S \cup \{v_{2i+1}, v_{2i+2}\}$ where $h^*((v_{2i+1}, v_{2i+2})) = 1$ and the other weights are the same as in h_S . If h^* is stable in $S \cup \{v_{2i+1}, v_{2i+2}\}$, then it follows that $M_{S'} = M_S \cup \{(v_{2i+1}, v_{2i+2})\}$ is an h^* -stable matching in $S' = S \cup \{v_{2i+1}, v_{2i+2}\}$. Otherwise, if h^* is not stable in $S \cup \{v_{2i+1}, v_{2i+2}\}$, then we add v_{2i+1} and v_{2i+2} one by one to S as follows.

Without loss of generality suppose that v_{2i+1} is involved in a blocking pair with an agent of S with respect to matching $M_S \cup \{(v_{2i+1}, v_{2i+2})\}$, let v_j be the best blocking partner of v_{2i+1} and let $S' = S \cup \{v_{2i+1}\}$. If v_j is unmatched in h_S (and so also in M_S), then $M_{S'} = M \cup \{(v_{2i+1}, v_j)\}$ is an $h_{S'}$ -stable matching in S' where $h_{S'}((v_{2i+1}, v_j)) = 1$ and otherwise it is the same as h_S . Note that (v_{2i+1}, v_j) must be a blocking pair for M_S too, so we may obtain $M_{S'}$ from M_S by satisfying (v_{2i+1}, v_j) .

If v_j is covered by a half-weighted odd cycle in h_S , say by $C = (v_{c_1}, \dots, v_{c_{2l+1}})$ where $v_{c_1} = v_j$, then we proceed as follows. Note v_j may be matched to her preferred partner among her two neighbours in C , say to v_{c_2} . In this case it may be possible that (v_{2i+1}, v_j) is blocking for h_S but it is not blocking for M_S . However, in this case, we can always rotate the edges of M_S in C by successively satisfying blocking pairs so that v_j becomes unmatched. Then we can satisfy (v_{2i+1}, v_j) and obtain an $h_{S'}$ -stable matching $M_{S'}$ where $h_{S'}$ is the stable half-matching that we get from h_S according to the corresponding Tan-Hsueh algorithm.

Finally, if v_j is matched in h_S (and also in M_S) to an agent v_{j_1} , then we satisfy (v_{2i+1}, v_j) obtaining a matching $M_{S' \setminus \{v_{j_1}\}} = (M_S \setminus \{(v_j, v_{j_1})\}) \cup \{(v_{2i+1}, v_j)\}$

which is an $h_{S' \setminus \{v_{j_1}\}}$ -stable matching in $S' \setminus \{v_{j_1}\}$, where $h_{S' \setminus \{v_{j_1}\}}$ is a stable half-matching in $S' \setminus \{v_{j_1}\}$ that we obtain in the corresponding Tan-Hsueh algorithm. Again, we continue the same process with v_{j_1} .

If a repetition occurs for the first time, namely, when an agent who became unmatched at a point of this process later gets involved in a blocking pair satisfied in the process, then in the corresponding Tan-Hsueh algorithm we form a new half-weighted odd cycle from the agents involved in the cycling, resulting in a new stable half-matching $h_{S'}$. The half-weighted edges of this new odd cycle correspond to those pairs that we would satisfy if we were to continue the proposal-rejection sequence. But regarding the matching $M_{S'}$, we can just stop after seeing the first repetition, and $M_{S'}$ will be an $h_{S'}$ -stable matching.

Note that if a repetition occurs, then we have to satisfy at most m blocking pairs (since each left-alone agent keeps getting worse partners, so no pair occurs twice as a blocking pair). Otherwise, if we have no repetition, then we also reach a new h -stable matching within m steps, since even if we have to rotate edges along a half-weighted odd cycle, the agents of this cycle could not be involved in any blocking pair satisfied before we rotated the edges along this cycle. Thus we can obtain the final h -stable matching in mn steps. \square

This result implies the theorem of Diamantoudi *et al.* [18] with an upper bound on the number of steps needed to reach a stable matching.

Corollary 1. *Suppose that we are given a solvable instance of SR and a matching M_0 , then one can always reach a stable matching starting from M_0 by successively satisfying at most mn blocking pairs.*

Finally in this section, we consider the differences between our proof technique and those of Diamantoudi *et al.* [18] and Inarra *et al.* [23]. Their proofs are based on the following idea. They suppose that a stable (or h -stable) matching is given together with the starting matching and then show that after satisfying a number of blocking pairs the initial matching can be transformed into another matching which shares more pairs with the given stable (or h -stable) matching than the initial one. This is not actually explicitly stated in the proof of Diamantoudi *et al.*, but it is not hard to see that in the above transformation no pair is satisfied twice, so we can obtain the same mn upper bound for the number of steps needed to reach a stable matching. In our proof, which is the extension of the Roth-Vande Vate technique for the roommates problem, we do not need to know a stable (or h -matching) in advance. The process works in an incremental way: successively enlarge a set of agents for which the corresponding matching is stable (or h -stable). Our proof also implies that if a new agent enters (or leaves) the market then the stability (or h -stability) of the matching can be restored after satisfying at most m blocking pairs. For more motivation with regard to the advantages of our alternative proof technique, please see the first paragraph of Section 4.

3 Analysing the market behavior with automata

If the input is random, then it is possible to simulate the dynamics of a matching market where two agents meet with each other randomly and behave in a myopic way (i.e. they form a new pair if they both would be better off). This is called the better response dynamics by Ackermann *et al.* [1]; whilst, Klaus *et al.* [30] refer to it as unperturbed blocking dynamics. What is the expected outcome of a matching market with myopic agents? To answer this question first we define the stable marriage and roommates automata as follows.

Definition 1. *Let I be a SR (SM) instance with underlying graph G . The stable roommates automaton (stable marriage automaton) of I , denoted $\text{SRA}(I)$ ($\text{SMA}(I)$) is the automaton $(\mathcal{M}(G), E(G), \delta, M_0, S_I)$ where:*

- the set of states is the set of all matchings $\mathcal{M}(G)$ of G ;
- the set of symbols is the set of edges $E(G)$ of G ;
- the transition function $\delta : \mathcal{M}(G) \times E(G) \rightarrow \mathcal{M}(G)$ is given by:

$$\delta(M, (v_i, v_j)) = \begin{cases} M^{(v_i, v_j)} & \text{if } (v_i, v_j) \text{ blocks } M \\ M & \text{otherwise} \end{cases}$$

- the initial state M_0 is any matching (e.g. the empty matching \emptyset);
- the set of accepting states equals the set S_I of stable matchings of I .

Recall, for a matching M and blocking pair (v_i, v_j) , $M^{(v_i, v_j)}$ is the matching such that $(v_i, v_j) \in M^{(v_i, v_j)}$ and for each $w \in \{v_i, v_j\}$, if w is matched in M , then $M(w)$ is unmatched in $M^{(v_i, v_j)}$. The set of symbols in the input of this automaton represents the pairs that are picked randomly in the stochastic process. The automaton *accepts* an input if the final state of the automaton is a stable matching.

Suppose that in each step of the process each blocking edge is chosen with equal probability, then starting from an arbitrary matching (e.g. the empty matching \emptyset) we can calculate the exact probabilities of particular matchings occurring after certain rounds. To be more precise, we will calculate these probabilities in the following absorbing Markov chain.

Definition 2. *Let I be a SR instance with corresponding automaton $\text{SRA}(I) = (\mathcal{M}(G), E(G), \delta, M_0, S_I)$. The Markov chain of I is given by $(\mathcal{M}(G), M_0, \mathbf{P})$ where the set of states and initial state are taken from $\text{SRA}(I)$ and the probability transition matrix $\mathbf{P} : \mathcal{M}(G) \times \mathcal{M}(G) \rightarrow [0, 1]$ is such that for $M, M' \in \mathcal{M}(G)$:*

- if M is stable, then $\mathbf{P}(M, M')$ equals 1 if $M=M'$ and 0 otherwise;
- if M is not stable, then

$$\mathbf{P}(M, M') = \frac{|\{(v, v') \in E(G) \mid (v, v') \text{ blocks } M \text{ and } \delta(M, \sigma) = M'\}|}{|\{(v, v') \in E(G) \mid (v, v') \text{ blocks } M\}|}.$$

For any SM instance or solvable SR instance, the stochastic process is an absorbing Markov chain where the absorbing states are the stable matchings.

We now report on our experiments to construct and analyse the Markov chain of a number of different instances with the probabilistic model checking tool PRISM [35,51]. PRISM is a tool for the formal modelling and analysis of systems that exhibit random or probabilistic behaviour. It supports a wide range of probabilistic models and has been used to analyse systems from many different application domains, including communication and multimedia protocols, randomised distributed algorithms, security protocols and biological systems.

Models are described using the PRISM language, a simple, state-based language based on guarded commands. The basic components of the modelling language are *modules* and *variables*. A system model is defined by specifying a set of modules, with the state of each module represented by a finite number of variables. The behaviour of each module is given by a set of guarded commands of the form:

$$[\langle \text{action} \rangle] \langle \text{guard} \rangle \rightarrow \langle \text{prob} \rangle : \langle \text{update} \rangle + \dots + \langle \text{prob} \rangle : \langle \text{update} \rangle;$$

The action label is used to force modules to synchronise (i.e. execute their commands simultaneously) and the guard is a predicate over all the variables of the model, indicating when the command is enabled. The updates describe the probabilistic transitions that the module can make when the command is executed, i.e. the changes made to its own variables; primed variables indicate the next values of variables.

PRISM provides support for automated analysis of a wide range of quantitative properties such as “*what is the probability of a failure within 4 hours?*”, “*what is the worst-case probability of the protocol terminating in error, over all possible initial configurations?*”, “*what is the expected size of the buffer after 30 minutes?*”, or “*what is the worst-case expected time taken for the program to terminate?*”. The property specification language incorporates the temporal logics PCTL, CSL and LTL, as well as extensions for quantitative specifications and rewards. PRISM incorporates state-of-the art symbolic data structures and algorithms, based on BDDs (Binary Decision Diagrams) and MTBDDs (Multi-Terminal Binary Decision Diagrams) [34,41]. It also includes a discrete-event simulation engine, providing support for approximate and statistical verification, and implementations of various different analysis techniques, such as quantitative abstraction refinement and symmetry reduction.

Below we give an overview of the PRISM model for an SM instance with three men and three women. To represent the preference lists of the men and the women the model includes constants $m_{i,j}, w_{i,j} \in \{1, 2, 3\}$ for $i, j=1, \dots, 3$, where $m_{i,j}$ equals the j th preference of man i and $w_{j,i}$ equals the i th preference of woman j . There is a module M_i ($i \in \{1, \dots, 3\}$) for each man and a module W_j ($j \in \{1, \dots, 3\}$) for each woman. Each module has an integer variable (denoted m_i and w_j respectively) in the range $\{0, 1, 2, 3\}$ representing the current matching of the man or woman (where 0 corresponds to the case when the man or woman is unmatched).

The commands of the modules correspond to the men and women trying to improve their current matching by synchronising on actions, more precisely, when man i and woman j form a blocking pair, the action $b_{i,j}$ can be performed and performing the action updates the matching so that i and j become matched. For example, the commands of man 1 (in module M_1) take the form:

$$\begin{aligned} [b_{1,1}] \quad & m_1=0 \vee (m_1=2 \wedge m_{1,1} > m_{1,2}) \vee (m_1=3 \wedge m_{1,1} > m_{1,3}) \rightarrow (m'_1=1); \\ [b_{1,2}] \quad & m_1=0 \vee (m_1=1 \wedge m_{1,2} > m_{1,1}) \vee (m_1=3 \wedge m_{1,2} > m_{1,3}) \rightarrow (m'_1=2); \\ [b_{1,3}] \quad & m_1=0 \vee (m_1=1 \wedge m_{1,3} > m_{1,1}) \vee (m_1=2 \wedge m_{1,3} > m_{1,2}) \rightarrow (m'_1=3); \end{aligned}$$

where the second command states: if either man 1 is unmatched or would prefer to be matched to woman 2 than his current matching (i.e. either matched to woman 1 and woman 2 is higher than woman 1 in his preference list, or is matched to woman 3 and woman 2 is higher than woman 3 in his preference list), then the module for man 1 is able to perform the action $b_{1,2}$ and update its matched partner to woman 2. The commands of the women are similar with the indexing of action names reversed. For example, the commands of woman 2 (in module W_2) are given by:

$$\begin{aligned} [b_{1,2}] \quad & w_2=0 \vee (w_2=2 \wedge w_{2,1} > w_{2,2}) \vee (w_2=3 \wedge w_{2,1} > w_{2,3}) \rightarrow (w'_2=1); \\ [b_{2,2}] \quad & w_2=0 \vee (w_2=1 \wedge w_{2,2} > w_{2,1}) \vee (w_2=3 \wedge w_{2,2} > w_{2,3}) \rightarrow (w'_2=2); \\ [b_{3,2}] \quad & w_2=0 \vee (w_2=1 \wedge w_{2,3} > w_{2,1}) \vee (w_2=2 \wedge w_{2,3} > w_{2,2}) \rightarrow (w'_2=3); \end{aligned}$$

in this case, the first command states: if either woman 2 is unmatched or would prefer to be matched to man 1 than her current matching (i.e. either matched to man 2 and man 1 is higher than man 2 in her preference list, or is matched to man 3 and man 1 is higher than man 3 in her preference list), then the module for woman 2 is able to perform the action $b_{1,2}$ and update its matched partner to man 1. Notice that for *both* man 1 and woman 2 are able to perform the action $b_{1,2}$ they must both prefer each other to their current matching, i.e. the action can only be performed when they form a blocking pair.

When a matching is updated to remove a blocking pair i and j , then the people previously matched to i and j must also update their preference lists, and therefore additional commands (synchronising on $b_{i,j}$) are included in each module to perform these updates. Further details of the models and experiments are available from the PRISM website [52]. For small instances, we also exported the PRISM models to the symbolic solver Maple [39] and computed the exact rational values. Note that in Examples 1-3 the absorption probabilities are calculated with the empty matching as starting state.

Example 1. We start with a classical instance by Gale and Shapley [20] with three men and three women and the following preferences:

$$\begin{array}{lll} m_1 : w_1, w_2, w_3 & m_2 : w_2, w_3, w_1 & m_3 : w_1, w_2, w_3 \\ w_1 : m_2, m_3, m_1 & w_2 : m_3, m_1, m_2 & w_3 : m_1, m_2, m_3 \end{array}$$

Here, the Markov chain has 34 states and 123 transitions, and the following three absorbing states (stable matchings):

$$\begin{aligned} M_m &= \{(m_1, w_1), (m_2, w_2), (m_3, w_3)\} && \text{(man-optimal)} \\ M_w &= \{(m_1, w_3), (m_2, w_1), (m_3, w_2)\} && \text{(woman-optimal)} \\ M_e &= \{(m_1, w_2), (m_2, w_3), (m_3, w_1)\} && \text{(egalitarian)} \end{aligned}$$

Calculating the absorption probabilities we find:

$$x^*(M_m) = x^*(M_w) = \frac{299}{1362} \sim 0.2195301028 \quad \text{and} \quad x^*(M_e) = \frac{382}{681} \sim 0.5609397944.$$

The egalitarian stable matching is therefore more likely than both the extreme solutions together. This differs from using the random order mechanism, since in this case the egalitarian stable matching is not achievable (as nobody gets their best stable partner) and the remaining stable matchings have probability $\frac{1}{2}$.

Example 2. The following classical instance was proposed by Knuth [31] with four men and four women and the following preferences:

$$\begin{aligned} m_1:w_1, w_2, w_3, w_4 \quad m_2:w_2, w_1, w_4, w_3 \quad m_3:w_3, w_4, w_1, w_2 \quad m_4:w_4, w_3, w_2, w_1 \\ w_1:m_4, m_3, m_2, m_1 \quad w_2:m_3, m_4, m_1, m_2 \quad w_3:m_2, m_1, m_4, m_3 \quad w_4:m_1, m_2, m_3, m_4 \end{aligned}$$

In this case, the Markov chain has 209 states, 1280 transitions, and the following 10 absorbing states (stable matchings):

$$\begin{aligned} M_1 &= \{(m_1, w_1), (m_2, w_2), (m_3, w_3), (m_4, w_4)\} \\ M_2 &= \{(m_2, w_1), (m_1, w_2), (m_3, w_3), (m_4, w_4)\} \\ M_3 &= \{(m_1, w_1), (m_2, w_2), (m_4, w_3), (m_3, w_4)\} \\ M_4 &= \{(m_2, w_1), (m_1, w_2), (m_4, w_3), (m_3, w_4)\} \\ M_5 &= \{(m_3, w_1), (m_1, w_2), (m_4, w_3), (m_2, w_4)\} \\ M_6 &= \{(m_2, w_1), (m_4, w_2), (m_1, w_3), (m_3, w_4)\} \\ M_7 &= \{(m_3, w_1), (m_4, w_2), (m_1, w_3), (m_2, w_4)\} \\ M_8 &= \{(m_4, w_1), (m_3, w_2), (m_1, w_3), (m_2, w_4)\} \\ M_9 &= \{(m_3, w_1), (m_4, w_2), (m_2, w_3), (m_1, w_4)\} \\ M_{10} &= \{(m_4, w_1), (m_3, w_2), (m_2, w_3), (m_1, w_4)\} \end{aligned}$$

and calculating the absorption probabilities we find:

$$\begin{aligned} x^*(M_1) &= x^*(M_{10}) = \frac{549582018404187049}{9518428268802561564} \sim 0.0577387362 \\ x^*(M_2) &= x^*(M_3) = x^*(M_8) = x^*(M_9) = \frac{1582747100504304809}{19036856537605123128} \sim 0.0831412002 \\ x^*(M_4) &= x^*(M_7) = \frac{61576717268573787}{528801570489031198} \sim 0.1164457912 \\ x^*(M_5) &= x^*(M_6) = \frac{253084017443076793}{1586404711467093594} \sim 0.1595330722 . \end{aligned}$$

Using the random order mechanism we find M_4, \dots, M_7 are not achievable, whilst in our case it is more likely one of these matchings will be reached.¹²

¹² The probabilities of getting these six matchings by the random order mechanism are as follows [29]: $p(M_1)=p(M_{10})=\frac{9600}{40320}$ and $p(M_2)=p(M_3)=p(M_8)=p(M_9)=\frac{5280}{40320}$.

Example 3. In this example we consider the roommates instance from [30, Example 3, page 25], provided by Elena Molis. This instance concerns eight agents with the following preferences:

$$\begin{aligned}
a_1 &: a_2, a_3, a_4, a_6, a_5, a_7, a_8 \\
a_2 &: a_3, a_1, a_4, a_5, a_6, a_8, a_7 \\
a_3 &: a_1, a_2, a_4, a_5, a_6, a_7, a_8 \\
a_4 &: a_6, a_3, a_5, a_1, a_2, a_7, a_8 \\
a_5 &: a_4, a_7, a_1, a_2, a_3, a_6, a_8 \\
a_6 &: a_7, a_4, a_2, a_3, a_1, a_5, a_8 \\
a_7 &: a_5, a_6, a_1, a_2, a_3, a_4, a_8 \\
a_8 &: a_3
\end{aligned}$$

It is an unsolvable instance, it admits two stable half-matchings (with no even cycles), namely h_1 and h_2 , where

$$\begin{aligned}
h_1((4, 5)) = h_1((6, 7)) = 1 \quad \text{and} \quad h_1((1, 2)) = h_1((2, 3)) = h_1((3, 1)) = \frac{1}{2} \\
h_2((4, 6)) = h_2((5, 7)) = 1 \quad \text{and} \quad h_2((1, 2)) = h_2((2, 3)) = h_2((3, 1)) = \frac{1}{2}
\end{aligned}$$

The h_1 -stable matchings are:

$$\begin{aligned}
M_1 &= \{(2, 3), (4, 5), (6, 7)\} \\
M_2 &= \{(1, 2), (4, 5), (6, 7)\} \\
M_3 &= \{(1, 3), (4, 5), (6, 7)\}
\end{aligned}$$

while the h_2 -stable matchings are:

$$\begin{aligned}
M_4 &= \{(2, 3), (4, 6), (5, 7)\} \\
M_5 &= \{(1, 2), (4, 6), (5, 7)\} \\
M_6 &= \{(1, 3), (4, 6), (5, 7)\}
\end{aligned}$$

Theorem 1 states that, starting from any matching, we can always reach one of these matchings by successively satisfying blocking pairs. This implies that any ergodic set (which is called *absorbing set* in [24] and [30]) of the corresponding Markov chain must contain some of the above matchings. Constructing this instance in PRISM, we find there are 308 matchings and a single ergodic set which consists of the matchings $\{M_4, M_5, M_6, M_7\}$, where $M_7 = \{(1, 2), (3, 8), (4, 6), (5, 7)\}$. This corresponds to the results presented in [30]. Computing the long-term likelihood of being in any one of the matching (i.e. the steady state probabilities of the Markov chain) we find:

$$x^*(M_4) = x^*(M_5) = x^*(M_6) = \frac{2}{7} \sim 0.285714 \quad \text{and} \quad x^*(M_7) = \frac{1}{7} \sim 0.142857.$$

Case study. We now compare the performance characteristics of a number of different instances of the SM problem, as the number of men and women $k(=n/2)$ varies between 4 and 8.

Model	k	states	transitions	expected time	
				av.	max.
<i>Symmetric</i>	4	208	1,433	3.595	5.713
	5	1,545	15,901	5.456	7.919
	6	13,326	189,691	7.692	11.05
	7	130,921	2,450,001	10.30	14.09
	8	1,441,728	34,194,273	13.27	17.97
<i>Uncoord</i>	4	208	1,268	18.97	25.22
	5	1,545	14,205	84.23	93.74
	6	13,326	170,886	399.2	413.5
	7	130,921	2,222,745	2,197	2,216
	8	1,441,728	31,209,032	14,361	14,385
<i>Uniform</i>	4	87	369	6.822	9.160
	5	665	4,746	12.04	14.92
	6	5,972	64,341	19.08	22.73
	7	61,215	926,095	28.18	32.42
	8	702,311	14,175,310	39.61	44.56
<i>1000 random samples (min)</i>	4	102	461	4.735	6.932
	5	993	8,524	8.082	11.21
	6	9,272	119,035	11.61	15.59
	7	130,884	2,378,889	15.93	20.89
<i>1000 random samples (average)</i>	4	193	1,247	8.032	10.65
	5	1,562	15,618	13.83	17.34
	6	13,317	192,465	22.84	27.28
	7	130,918	2,524,157	37.34	42.74
<i>1000 random samples (max)</i>	4	208	1,460	17.25	20.30
	5	1,545	16,660	46.28	50.68
	6	13,326	202,560	115.8	121.1
	7	130,921	2,657,024	164.9	170.7

Table 1. Expected time to reach a stable matching from a complete initial matching

- *Symmetric*: in this instance the preferences of the men and women are of the form $m_j : w_j, \dots, w_k, w_1, \dots, w_{j-1}$ and $w_j : m_j, \dots, m_k, m_1, \dots, m_{j-1}$.
- *Uncoord*: this instance is used in [1] to show an exponential lower bound for the convergence time. The preference lists in this instance are given by $m_j : w_{j+1}, \dots, w_k, w_1, \dots, w_j$ and $w_j : m_j, m_{j+1}, \dots, m_k, m_1, \dots, m_{j-1}$.
- *Uniform*: in this case the preference lists of all men and all women are the same and equal w_1, w_2, \dots, w_k and m_1, m_2, \dots, m_k respectively.

In our experiments we consider both the case when we start with a random (complete) matching and the empty matching. Tables 1 and 2 report on the model statistics (states and transitions) of the Markov chains generated with PRISM. Table 1 includes both the average and the maximum expected time to reach a stable matching when starting from a complete matching, while Table 2 the expected time when starting from the empty matching and number of stable matchings. For comparison, the tables also includes the minimum, average and maximum values obtained from a sample of 1,000 random instances.¹³

The number of states reported in Tables 1 and 2 demonstrate that, when starting from a randomly chosen complete matching, the number of reachable matchings is dependent on the particular instance. We also see that for the *Symmetric* and *Uncoord* instances all matchings (except the empty matching) are

¹³ Since for $k=8$ each instance takes over 20 minutes to analyse, it was not feasible to study 1,000 different instances.

Model	k	states	transitions	expected time	no. of stable matchings
<i>Symmetric</i>	4	209	1,449	7.469	1
	5	1,546	15,926	10.51	1
	6	13,327	189,727	13.95	1
	7	130,922	2,450,050	17.78	1
	8	1,441,729	34,194,337	21.99	1
<i>Uncoord</i>	4	209	1,284	28.04	4
	5	1,546	14,230	97.16	5
	6	13,327	170,922	416.5	6
	7	130,922	2,222,794	2,220	7
	8	1,441,729	31,209,096	14,388	8
<i>Uniform</i>	4	209	1,421	11.31	1
	5	1,546	15,926	17.66	1
	6	13,327	192,862	25.82	1
	7	130,922	2,525,804	36.03	1
	8	1,441,729	35,686,961	48.56	1
<i>1000 random samples (min)</i>	4	209	1,421	7.851	1
	5	1,546	15,926	11.53	1
	6	13,327	192,862	15.77	1
	7	130,922	2,525,804	21.04	1
<i>1000 random samples (average)</i>	4	209	1,421	11.02	1.506
	5	1,546	15,926	17.54	1.657
	6	13,327	192,862	27.39	1.961
	7	130,922	2,525,804	42.80	2.187
<i>1000 random samples (max)</i>	4	209	1,421	20.35	5
	5	1,546	15,926	50.61	5
	6	13,327	192,862	121.2	7
	7	130,922	2,525,804	170.8	9

Table 2. Expected time to reach a stable matching from the empty initial matching

reachable. The results for the *Uncoord* instances are far slower than for the other instances, corresponding to the fact that [1] uses this instance to demonstrate an exponential lower bound on the convergence time. Considering the random sample results, we see that the performance of the *Uncoord* instance is unlikely to be seen in practice. These results also indicate that the number of stable matches does not seem to be the cause of the slow convergence time demonstrated by the *Uncoord* instance, since the number of stable matchings for some random matchings were even greater than for the *Uncoord* instances of the same size, but the expected convergence times were significantly smaller. To further demonstrate how PRISM can be used to analyse instances, Figure 1 plots the probability of reaching a stable matching within R rounds when starting from the empty configuration.

4 Further remarks

First, we would like to propose an aspect of our alternative proof (described in Section 2) for further consideration, that may deepen our understanding of the behaviour of decentralised matching markets. There is a slight contradiction in the theoretical results with regard to the convergence time of stochastic processes in matching markets. Our theorem (and implicitly also the proof of Diamantoudi *et al.* [18]) says that a stable matching can be always reached after satisfying at

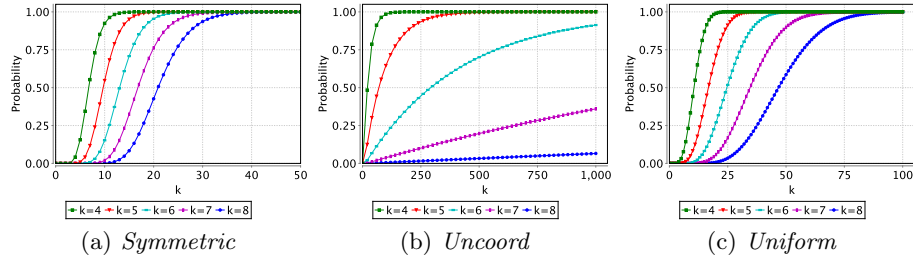


Fig. 1. Probability of reaching a stable matching within R rounds.

most mn blocking pairs, whilst the theoretical results by Ackermann *et al.* [1] and the simulations by Boudreau [14] show that the expected convergence time can be exponential for ‘uncoordinated’ markets under the better response dynamics. How do real markets work? Recent experimental papers ([19,40]) show that in laboratory environment the frequency of getting different stable matchings are close to the absorption probabilities, even if only one side of the market make proposals [19]. However, there are some other papers ([11,12]) where the authors argue that real two-sided job markets (especially with regard to senior positions) work differently, and instead of the better response dynamics, they rather follow the proposal-rejection sequences of Roth and Vande Vate.¹⁴ If the latter conjecture is true then this can clarify why some decentralised job markets are able to reach a stable state relatively quickly. Our proof technique, which is an extension of the Roth-Vande Vate argument for the roommates problem, can also have similar policy implications. If we request the agents of a roommates market to enter (or leave) the market one by one and encourage them to make offers to the best available mate according to their preferences, then these kind of stochastic markets would be likely to reach a stable (or close to stable) state more quickly than the ones with no coordination.¹⁵ These conjectures would be interesting to investigate using both laboratory experiments and by inspecting real decentralised matching markets. In fact, the above best response policy could also be used as a protocol in P2P systems, as Lebedev *et al.* [36] suggested for

¹⁴ We might think about a new opening in the senior academic market. If this position is linked to a well-specified research topic, then it is a realistic assumption that the potential candidates will be reached by the advertisements and the selection process ensures that the best applying candidate will be selected. Therefore, we can perhaps assume that in a real (sub-)market only one new position makes the market instable, and also the stochastic process might be better described with the best response dynamics rather than better responses.

¹⁵ Note that path to stability constructed in the proof of Diamantoudi *et al.* [18] cannot be translated into a policy for agents to follow, as the construction is based on the knowledge of a stable matching, which could be obtained only by a central coordinator.

special roommates settings, which can make a decentralised system to converge to a stable state rapidly.

Calculations of the absorption probabilities have been conducted in several recent papers ([14,16,19]) by extensive simulations. Doing these simulations is not an easy task¹⁶. As we demonstrate in Section 3, PRISM is a very useful tool for analysing stochastic processes of this kind. The reader is welcome to download this open-source model checker tool and repeat our experiments or try different ones with the code we have made available from the PRISM website [52].

As an extension of the approach presented in this paper, it would be interesting to study stochastic processes occurring in more general settings, for example, in *hedonic coalition formation games*, where the size of possible coalitions can be larger than two, see e.g. [4] and [13]. However, in this case the existence of a stable solution (i.e., a nonempty core) does not guarantee that there is a convergence to a stable solution when starting from any unstable state, as illustrated by Diamantoudi *et al.* [18]. So in this case absorbing states and ergodic sets may appear together in the Markov chain. Yet, one could investigate the structure of absorbing and ergodic states for special classes of coalition formation games, and analyse particular games in a similar framework as we did here, using PRISM.

Furthermore, more general network formation games and matching problems could also be analysed with this technique. Kojima and Ünver [33] extended the results of Roth and Vande Vate to many-to-many markets. Another example for such problems is the resident allocation problem with couples, where the existence of a stable matching is not guaranteed in general (for a survey on this problem, see [8]). However, for particular preference structures Klaus and Klijn [28] show that not only that the existence of a stable solution can be guaranteed, but also that the path to stability from any starting matching.

Finally, the same questions can be asked for cooperative games with transferable utilities, such as the matching game [9], or the slightly different stable matching problem with payments [10], where the agents who are matched together may share the value of their cooperation between each other. In a very recent paper Biró *et al.* [10] showed that starting from any solution in a roommates market with payments we can reach a stable solution by satisfying at most $2m$ blocking pairs, if a stable solution exists. Similar result has been shown for two-sided markets by Chen *et al.* [17]. Regarding general TU-games, there

¹⁶ For example, Boudreau [15] wrote the following in a recent paper: “*Calculating the probability of each stable outcome for a given market under the randomized tâtonnement process is extremely difficult due to the tremendous number of paths that can be involved (see [14] for an idea of just how long random paths to stability are likely to be). Loops in the process mean that a closed form solution is virtually impossible to obtain. Fortunately, the process can easily be programmed into software such as Matlab (copyright The Mathworks). Trials of the process can thereby be repeated over and over again by starting from the empty matching, assigning uniform probability to all blocking pairs existing at each step, and keeping track of which particular stable outcome prevails each time. After one million such trials, it is then possible to obtain a probability distribution for a given markets set of stable outcomes based on the simulated experimental data.*”

are also results on the *accessibility of the core* and the number of blocks needed to access the core (or some other desired set of imputations), see e.g. [32], [5] and [50]. There can be seen as counterpart of the path to stability results for matching markets.

Acknowledgement. We would like to thank three anonymous referees for giving helpful comments. We also thank the feedbacks given by the participants of the 17th Coalition Theory Network workshop (Paris, France), the Workshop on Frontiers of Mechanism Design: Matching Markets (Monte Verit, Switzerland), the 8th Spain-Italy-Netherlands Meeting on Game Theory (Budapest, Hungary), the Corvinus Game Theory Seminars and the Granada Economics Seminars.

References

1. H. Ackermann, P.W. Goldberg, V. Mirrokni, H. Röglin and B. Vöcking. Uncoordinated two-sided matching markets. *Proc. 9th ACM Conference on Electronic Commerce (EC'08) Chicago, USA*, pages 256–263, 2008.
2. R. Aharoni and T. Fleiner. On a lemma of Scarf. *J. Combin. Theory Ser. B*, 87(1):72–80, 2003. Dedicated to Crispin St. J. A. Nash-Williams.
3. E. Arcaute and S. Vassilvitskii. Social networks and stable matchings in the job market. *Proc. 5th Intl. Workshop Internet and Network Economics (WINE'09)*, pages 220–231, 2009.
4. S. Banerjee, H. Konishi and T. Sönmez. Core in a Simple Coalition Formation Game. *Social Choice Welfare*, 18:135–153, 2001.
5. S. Béal, E. Rémila and P. Solal. On the number of blocks required to access the coalition structure core. *Working paper, Munich Personal RePEc Archive, MPRA Paper No. 29755*, 2011.
6. P. Biró, K. Cechlárová and T. Fleiner. The dynamics of stable matchings and half-matchings for the stable marriage and roommates problems. *Int. Journal of Game Theory*, 36:333–352, 2008.
7. P. Biró and T. Fleiner. Fractional solutions for NTU-games. *Proc. 3rd Workshop on Computational Social Choice*, 2010.
8. P. Biró and F. Klijn. Matching with Couples: a Multidisciplinary Survey. *To appear in International Game Theory Review*, 2012.
9. P. Biró, W. Kern, and D. Paulusma: Computing solutions for matching games. *Int. Journal of Game Theory*, 41:75–90, 2012.
10. P. Biró, M. Bomhoff, P.A. Golovach, W. Kern, and D. Paulusma: Solutions for the Stable Roommates Problem with Payments. *To appear in the proceedings of WG 2012: 38th International Workshop on Graph Theoretic Concepts in Computer Science*, 2012.
11. Y. Blum, A.E. Roth and U.G. Rothblum. Vacancy chains and equilibration in senior-level labor markets. *Journal of Economic Theory*, 76:362–411, 1997.
12. Y. Blum and U.G. Rothblum. “Timing is everything” and marital bliss. *Journal of Economic Theory*, 103:429–443, 2002.
13. A. Bogomolnaia and M.O. Jackson. The Stability of Hedonic Coalition Structures. *Games and Economic Behavior*, 38:201–230, 2002.
14. J. Boudreau. Preference Structure and Random Paths to Stability in Matching Markets. *Economics Bulletin*, 67:1–12, 2008.

15. J. Boudreau. A note on the efficiency and fairness of decentralized matching. *Operations Research Letters*, 39:231-233, 2011.
16. J. Boudreau. An exploration into why some matchings are more likely than others. *To appear in the proceedings of MATCH-UP 2012: The Second International Workshop on Matching Under Preferences*, 2012.
17. B. Chen, S. Fujishige and Z. Yang. Decentralized Market Processes to Stable Job Matchings with Competitive Salaries. *Working paper, Kyoto University, RIMS-1715*, 2011.
18. E. Diamantoudi, E. Miyagawa and L. Xue. Random paths to stability in the roommates problem. *Games and Economic Behavior*, 48:18–28, 2004.
19. F. Echenique and L. Yariv. An Experimental Study of Decentralized Matching. *working paper*, 2011.
20. D. Gale and L.S. Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, 69:9–15, 1962.
21. D. Gusfield and R.W. Irving. *The Stable Marriage Problem: Structure and Algorithms*. MIT Press, 1989.
22. M. Hoefer. Local matching dynamics in social networks. *Proc. 38th Intl. Coll. Automata, Languages and Programming (ICALP 2011)*, LNCS 6756, pages 113–124, 2011.
23. E. Inarra, C. Larrea and E. Molis. Random paths to P -stability in the roommate problem. *Int. Journal of Game Theory*, 36:461–471, 2008.
24. E. Inarra, C. Larrea and E. Molis. The stability of the roommate problem revisited. *Proceedings of Matching Under Preferences (MATCH-UP) Satellite workshop of ICALP 2008*, 114–125, 2008.
25. R.W. Irving. An efficient algorithm for the “stable roommates” problem. *Journal of Algorithms*, 6:577–595, 1985.
26. M.O. Jackson and A. Watts. The evolution of social and economic networks. *Journal of Economic Theory*, 106(2):265–295, 2002.
27. J.G. Kemeny and J.L. Snell. *Finite Markov chains*. D. Van Nostrand Company, Inc., 1960.
28. B. Klaus and F. Klijn. Path to stability for matching markets with couples. *Games and Economic Behavior*, 58:154–171, 2007.
29. B. Klaus and F. Klijn. Corrigendum to “On randomized matching mechanisms” [Economic Theory 8(1996)377–381]. *Economic Theory*, 32:411–416, 2007.
30. B. Klaus, F. Klijn and M. Walzl. Stochastic stability for roommate markets. *Journal of Economic Theory*, 145:2218–2240, 2010.
31. D.E. Knuth. Mariages stable et leurs relations avec d’autres problèmes combinatoires. *Les Presses de l’Université de Montréal, Montréal*, 1976.
32. L.Á. Kóczy and L. Lauwers. The coalition structure core is accessible. *Games and Economic Behavior*, 48:86–93, 2004.
33. F. Kojima and Ü. Ünver. Random Paths to Pairwise Stability in Many-to-Many Matching Problems: A Study on Market Equilibration. *International Journal of Game Theory*, 36:473–488, 2008.
34. M.Z. Kwiatkowska, G. Norman, and D. Parker. Probabilistic Symbolic Model Checking with PRISM: A Hybrid Approach. *International Journal on Software Tools for Technology Transfer (STTT)*, 6(2):128–142, 2004.
35. M.Z. Kwiatkowska, G. Norman, and D. Parker. PRISM 4.0: Verification of Probabilistic Real-time Systems. *Proc. 23th Int. Conf. Computer Aided Verification (CAV’11)*, volume 6806 of LNCS, Springer, pages 585–591, 2011.

36. D. Lebedev, F. Mathieu, L. Viennot, A.-T. Gai, J. Reynier and F. de Montgolfier. On using matching theory to understand P2P network design. *Proc. INOC 2007: Int. Network Optimization Conference.*, pages 1–6, 2007.
37. J. Ma. On randomized matching mechanisms. *Economic Theory*, 8:377–381, 1996.
38. F. Mathieu. Acyclic preference-based systems. In Xuemin Shen, Heather Yu, John Buford, and Mursalin Akon, editors, *Handbook of Peer-to-Peer Networking*. Springer Verlag, Berlin Heidelberg, January 2010.
39. MapleSoft Corporation. Maple computer algebra system (www.maplesoft.com).
40. J. Pais, Á. Pintér and R. Veszteg. Decentralized Matching Markets: A Laboratory Experiment. *working paper*, 2011.
41. D. Parker. Implementation of Symbolic Model Checking for Probabilistic Systems. Ph.D. thesis, University of Birmingham. August 2002.
42. A.E. Roth and M.A.O. Sotomayor. Two-sided matching: a study in game-theoretic modeling and analysis. *Cambridge: Econometric Society monographs*, 1990.
43. A.E. Roth and J.H. Vande Vate. Random paths to stability in two-sided matching. *Econometrica*, 58:1475–1480, 1990.
44. H. E. Scarf. The core of an N person game. *Econometrica*, 35:50–69, 1967.
45. A. Tamura. Transformation from arbitrary matchings to stable matchings. *Journal of Combinatorial Theory, Series A*, 62:310–323, 1993.
46. J.J. Tan. A maximum stable matching for the roommates problem. *BIT Numerical Mathematics*, 30:631–640, 1990.
47. J.J. Tan. A necessary and sufficient condition for the existence of a complete stable matching. *Journal of Algorithms*, 12:154–178, 1991.
48. J.J. Tan and J.C. Hsueh. A generalisation of the stable matching problem. *Discrete Applied Mathematics*, 59:87–102, 1995.
49. J.J. Tan and W.C. Su. On the divorce digraph of the stable marriage problem. *Proc. National Science Council, Republic of China*, 19:342–354, 1995.
50. Y.-Y. Yang. Accessible outcomes versus absorbing outcomes. *Mathematical Social Sciences*, 62:65–70, 2011.
51. PRISM website. www.prismmodelchecker.org/
52. PRISM website, our experiments. www.prismmodelchecker.org/casestudies/stable_matching.php