The stable marriage problem with restricted pairs[*]

Vânia M. F. Dias[†] Guilherme D. da Fonseca[‡]
Celina M. H. de Figueiredo[§] Jayme L. Szwarcfiter[¶]

Abstract

A *stable matching* is a complete matching of men and women such that no man and woman who are not partners both prefer each other to their actual partners under the matching. In an instance of the STABLE MARRIAGE problem, each of the n men and n women ranks the members of the opposite sex in order of preference. It is well known that at least one stable matching exists for every STABLE MARRIAGE problem instance. We consider extensions of the STABLE MARRIAGE problem obtained by forcing and by forbidding sets of pairs. We present a characterization for the existence of a solution for the STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS problem. In addition, we describe a reduction of the STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS problem to the STABLE MARRIAGE WITH FORBIDDEN PAIRS problem. Finally, we also present algorithms for finding a stable matching, all stable pairs and all stable matchings for this extension. The complexities of the proposed algorithms are the same as the best known algorithms for the unrestricted version of the problem.

Keywords: algorithms, stable marriage, forced pairs, forbidden pairs, rotations.

[*]This work was partially supported by Conselho Nacional de Desenvolvimento Científico e Tecnológicoa - CNPq, Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro - FAPERJ, Coordenação de Aperfeiçoamento de Pessoal de nível Superior - CAPES, Brazilian research agencies.

[†]DCOP, Universidade Estadual de Londrina and COPPE, Universidade Federal do Rio de Janeiro. E-mail: `vaniad@cos.ufrj.br`.

[‡]COPPE, Universidade Federal do Rio de Janeiro. E-mail: `gfonseca@esc.microlink.com.br`.

[§]Instituto de Matemática and COPPE, Universidade Federal do Rio de Janeiro, Caixa Postal 68530, 21945-970 Rio de Janeiro, RJ, Brazil. E-mail: `celina@cos.ufrj.br`.

[¶]COPPE, Instituto de Matemática and Núcleo de Computação Eletrônica, Universidade Federal do Rio de Janeiro, Caixa Postal 68511, 21945-970 Rio de Janeiro, RJ, Brazil. E-mail: `jayme@nce.ufrj.br`.

1 Introduction

Given a set of n men and a set of n women, a *complete matching* is a set of n pairs, each pair containing one man and one woman, such that no person is in more than one pair. In an instance of the STABLE MARRIAGE problem, each of the n men and n women ranks the members of the opposite sex in order of preference. A *stable matching* is a complete matching of men and women for which there is no *blocking pair*: a pair of man and woman who are not partners and such that both prefer each other to their actual partners under the matching. It is well known that at least one stable matching exists for every STABLE MARRIAGE instance.

The Gale-Shapley algorithm [3] finds in time $O(n^2)$ a stable matching for a given STABLE MARRIAGE instance. A pair is *stable* if it is contained in some stable matching. Gusfield [4] gives algorithms for finding all stable pairs and all S (a number possibly exponential in n) stable matchings in $O(n^2)$ and $O(n^2+nS)$ time, respectively. The necessary background for the structure of the set of solutions and corresponding algorithms is presented in Section 2.

In this paper, we consider extensions of the STABLE MARRIAGE problem obtained by restricting pairs.

A set of pairs Q is *stable* if there is a stable matching M such that every pair in Q is a pair in M. We say that M is a *stable matching with forced pairs* Q. An algorithm to find in $O(n^2)$ time a stable matching with given forced pairs, if such a matching exists, was described by Knuth [8]. Gusfield and Irving [5] present a characterization for the existence of a solution of the STABLE MARRIAGE WITH FORCED PAIRS problem, and show how this characterization leads to an algorithm that tests for the existence of a solution of the STABLE MARRIAGE WITH FORCED PAIRS problem with forced pairs Q in $O(|Q|^2)$ time, after pre-processing the preference lists in $O(n^4)$ time.

Given a set of pairs P, we say that M is a *stable matching with forbidden pairs* P if every pair in P is not a pair of M. In Section 3, we present a characterization for the existence of a solution of the STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS problem, and we show how this characterization leads to an algorithm that tests for the existence of a solution of the STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS problem with forced pairs Q and forbidden pairs P in $O((|Q|+|P|)^2)$ time, after pre-processing the preference lists in $O(n^4)$ time. Such an algorithm can be useful if many sets of forced and forbidden pairs might be given. We end Section 3 by presenting a reduction of STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS to STABLE MARRIAGE WITH FORBIDDEN PAIRS. Given an instance of the STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS problem with forced pairs Q and forbidden pairs P, this reduction constructs an instance of the STABLE MARRIAGE WITH FORBIDDEN PAIRS problem with $(|P|+(n-1)|Q|)$ forbidden pairs. Note that this reduction increases the number of pairs that were previously forced by a factor of $n-1$, and this blow-up in the size of the instance justifies presenting the $O((|Q|+|P|)^2)$ algorithm (following $O(n^4)$ pre-processing time) in terms of the STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS problem, rather than assuming that $Q = \emptyset$ and presenting, for example, an $O(|P|^2)$ algorithm (following $O(n^4)$ pre-processing time) in terms of the STABLE MARRIAGE WITH FORBIDDEN PAIRS problem.

In Section 4 we describe algorithms which find, in case they exist, for the STABLE MARRIAGE WITH FORBIDDEN PAIRS problem: a stable matching, all stable matchings, and all stable pairs. The complexities of these algorithms are the same as Gusfield's algorithms for the unrestricted version of the STABLE MARRIAGE problem.

The extension STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS, where a set of forced pairs and a set of forbidden pairs are given, has been proposed and solved by Dias [1]. The reduction of STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS to STABLE MARRIAGE WITH FORBIDDEN PAIRS was considered by Fonseca [2] and applied to obtain algorithms which find, if they exist, in this extension: a stable matching, all stable matchings, and all stable pairs.

We conclude in Section 5 with a discussion on the optimality of our proposed algorithms. Usually, (see [6]) in generating combinatorial structures, listings with small prescribed differences between consecutive objects may allow their faster generation. The discussion and the example in Section 5 show that in any algorithm for explicitly finding all solutions of STABLE MARRIAGE the amount of computation between successive listed objects is $\Omega(n)$. As a consequence, no constant amortized time algorithm exists for the problem.

We end this introduction by describing an example showing that the marriage problem with restricted pairs cannot be reduced to the conventional problem, simply by changing the input data (as in the case of incomplete lists, for example). This fact justifies the more elaborate approach which has been taken in this paper. Superficially, STABLE MARRIAGE WITH FORBIDDEN PAIRS resembles STABLE MARRIAGE WITH INCOMPLETE LISTS – this is the variant of STABLE MARRIAGE in which persons may express unacceptable partners, so that preference lists may be incomplete (see [5], Section 1.4.2). The distinction between the variant proposed and solved in the present paper and STABLE MARRIAGE WITH INCOMPLETE LISTS is that, if a pair is a forbidden pair, it could still be a blocking pair with respect to a matching. Consider the following instance of STABLE MARRIAGE with three men m_1, m_2, m_3; three women w_1, w_2, w_3; and preference lists $m_1 : w_1w_2w_3$, $m_2 : w_2w_3w_1$, $m_3 : w_3w_1w_2$, $w_1 : m_2m_3m_1$, $w_2 : m_3m_1m_2$, $w_3 : m_1m_2m_3$. This instance admits three stable matchings: $M_0 = \{(m_1, w_1), (m_2, w_2), (m_3, w_3)\}$, $M_1 = \{(m_1, w_2), (m_2, w_3), (m_3, w_1)\}$, $M_z = \{(m_1, w_3), (m_2, w_1), (m_3, w_2)\}$. Now suppose we add as restriction the set of forbidden pairs $P = \{(m_1, w_1), (m_1, w_2)\}$. We have that just M_z is a stable matching for this instance of STABLE MARRIAGE WITH FORBIDDEN PAIRS. Consider the following instance of STABLE MARRIAGE WITH INCOMPLETE LISTS with three men m_1, m_2, m_3; three women w_1, w_2, w_3; and preference lists $m_1 : w_3$, $m_2 : w_2w_3w_1$, $m_3 : w_3w_1w_2$, $w_1 : m_2m_3$, $w_2 : m_3m_2$, $w_3 : m_1m_2m_3$. This instance admits two stable matchings: $M'_0 = \{(m_1, w_3), (m_2, w_2), (m_3, w_1)\}$, $M_z = \{(m_1, w_3), (m_2, w_1), (m_3, w_2)\}$. Note that M'_0 is not a stable matching for the original unrestricted case, since it has (m_1, w_2) as blocking pair. So M'_0 is not a stable matching for the restricted case obtained by forbidding the set P of pairs either. With respect to this restricted case, pair (m_1, w_2) is both forbidden and blocking. The reason why a pair (m, w) could be a forbidden pair, but could still form a blocking pair could be justified by considering a centralized matching scheme in which an administrator wishes to forbid (for whatever reason) two agents from becoming matched. Yet these two agents could find each other acceptable, leading to the possibility that they could form a blocking pair with respect to the constructed matching.

2 The Lattice of Stable Matchings

The following background for the structure of the set of solutions and corresponding algorithms of the STABLE MARRIAGE problem has been fully developed and described in [5]. We repeat some of these results here as they will be referred to in the sections that follow.

The Gale-Shapley algorithm [3] yields in time $O(n^2)$ what is called the *man-optimal stable matching*, denoted M_0, with the property that every man has the best partner he can have in any stable matching. If applied with the roles of men and women interchanged, the algorithm yields the *woman-optimal stable matching*, denoted M_z, which similarly favours the women.

Let M and M' be two stable matchings, and let $max_i(M, M')$ be the woman whom man i prefers between his two assigned partners in M and M'. Let $min_i(M, M')$ denote the other woman. Let $max(M, M')$ (respectively $min(M, M')$) be the mapping of each man i to $max_i(M, M')$ (respectively $min_i(M, M')$). Say that stable matching M *dominates* stable matching M' (from the perspective of men) if and only if $M = max(M, M')$. Say that a stable matching X is *between* M and M' if and only if M dominates X and X dominates M', while X differs from both M and M'. It is surprising but easy to show that $max(M, M')$ and $min(M, M')$ are both stable matchings. Hence, under the relation of dominance, the set of all stable matchings forms a lattice Λ where the join and meet operations are the max and min operations above. The unique maximum (most dominant) element of Λ is the man-optimal stable matching M_0, and the unique minimum (most dominated) element of Λ is the woman-optimal stable matching M_z.

The concept of *rotation* is crucial for understanding the structure of the lattice of solutions Λ of a STABLE MARRIAGE instance. Let M be a stable matching. Let w be the first woman in the list of m after his partner in M such that w prefers m to her partner in M. Let $next(m)$ be the partner of w in M. Then there is a sequence, called *rotation*, of pairs of M, say $\pi = (m_0, w_0), (m_1, w_1), \ldots, (m_{r-1}, w_{r-1})$ in the stable matching M, such that for each i, $0 \leq i \leq r-1$, m_{i+1} is equal to $next(m_i)$, where $i+1$ is taken modulo r. We say that rotation π is *exposed* in M. Denote by M/π the stable matching obtained by *elimination* of π, i.e., the stable matching where each $m_i \in \pi$ is married to w_{i+1}, while the remaining pairs are the same as in M.

Once an exposed rotation has been identified and eliminated, then one or more rotations may be exposed in the resulting matching. A rotation π is said to be an *explicit predecessor* of rotation $\rho = (m_0, w_0), (m_1, w_1), \ldots, (m_{r-1}, w_{r-1})$ if, for some $i, 0 \leq i \leq r-1$, and for some woman $w_q (\neq w_i)$, π is the eliminating rotation for (m_i, w_q) and m_i prefers w_q to w_{i+1}. Clearly a rotation cannot become exposed until all of its explicit predecessors have been eliminated. Further, the reflexive transitive closure $\preceq$ of the explicit predecessor relation is a partial order on the set of rotations, called the *rotation poset* denoted by $\Pi(\Lambda)$, and $\pi \prec \rho$ if and only if π must be eliminated before ρ becomes exposed.

A *closed set* in a poset $\Pi(\Lambda)$ is a subset S of $\Pi(\Lambda)$ such that if $\rho \in S$ and $\pi \prec \rho$ then $\pi \in S$. The following theorem was shown in [9].

Theorem 1 *The stable matchings of a given instance are in one-to-one correspondence with the closed subsets of the rotation poset.* ■

The set of all rotations can be found in time $O(n^2)$ and the explicit construction of the poset $\Pi(\Lambda)$ requires time $O(n^4)$. A compact representation of $\Pi(\Lambda)$ is achieved by constructing a digraph $G(\Lambda)$ which contains a subset of pairs of rotations such that the transitive closure of $G(\Lambda)$ is the poset $\Pi(\Lambda)$. The construction of $G(\Lambda)$ from the set of all rotations can be carried out in time $O(n^2)$.

Another important digraph is $\tilde{G}(\Lambda)$, a subgraph of $G(\Lambda)$. $\tilde{G}(\Lambda)$ can also be constructed in $O(n^2)$ and its transitive closure is also $\Pi(\Lambda)$ but the outdegree of every rotation is at most n. This upper bound on the outdegree is important to guarantee the $O(n^2 + nS)$ time complexity of the algorithm that lists all stable matchings. For details about the structure and properties of $\Pi(\Lambda)$, $G(\Lambda)$ and $\tilde{G}(\Lambda)$, we refer the reader to Gusfield and Irving [5].

3 A Characterization for Restricted Pairs

The following characterization was given by Gusfield in [4].

Theorem 2 *A pair (m, w) is a stable pair if and only if it is a pair in M_z or it is a pair in some rotation. Equivalently, (m, w) is stable if and only if it is a pair in M_0, or for some rotation $(m_0, w_0), (m_1, w_1), \ldots, (m_{r-1}, w_{r-1})$ and some i, $m = m_i$ and $w = w_{i+1}$.* ∎

We remark that the above characterization yields an $O(n^2)$ time algorithm that given an instance of STABLE MARRIAGE finds all stable pairs through the compact representation of $\Pi(\Lambda)$ by $G(\Lambda)$.

Let Q be a stable set of pairs. Hence, by definition, there exists a stable matching M such that every pair in Q is married in M. For each pair $(m, w) \in Q$ and not married under M_0, let $\gamma(m, w)$ be the unique rotation that *moves m to w*, i.e., $m = m_i$ and $w = w_{i+1}$, for some i, in the rotation $\pi = \gamma(m, w)$. For each pair $(m, w) \in Q$ and not married under M_z, let $\theta(m, w)$ be the unique rotation that *moves m from w*, the pair (m, w) belongs to the rotation $\pi = \theta(m, w)$. Note that, by Theorem 2, every stable pair (m, w) that is not in M_0 has a corresponding rotation $\gamma(m, w)$, and that every stable pair (m, w) that is not in M_z has a corresponding rotation $\theta(m, w)$.

In [5], the following characterization for stable sets is given. A corresponding algorithm is also presented, for deciding in $O(|Q|^2)$ time after pre-processing the preference lists in $O(n^4)$ time whether a given set Q of pairs is stable. The algorithm first constructs $\Pi(\Lambda)$ explicitly.

Theorem 3 *A set Q of pairs is stable if and only if each of the pairs is stable, and there are no two pairs (m,w) and (m',w') in Q such that $\theta(m,w) \preceq \gamma(m',w')$ in $\Pi(\Lambda)$.* ■

The following theorem is an extension of Theorem 3. Theorem 4 gives a characterization for determining whether, given two given sets of pairs Q and P, there exists a stable matching with set of forced pairs Q and set of forbidden pairs P.

Theorem 4 *Let P and Q be two sets of stable pairs. There exists a stable matching with set of forced pairs Q and set of forbidden pairs P if and only if there exists a set X of rotations such that:*
(i) for every pair $(m,w) \in Q$, we have either $(m,w) \in M_0$, or $\gamma(m,w) \in X$. In both cases, there is no $\rho \in X$ such that $\theta(m,w) \preceq \rho$;
(ii) for every pair $(m,w) \in P$, we have that if $(m,w) \in M_0$ or $\gamma(m,w) \preceq \rho$, for $\rho \in X$, then $\theta(m,w) \in X$.

Proof. ($\Rightarrow$) Suppose there exists a stable matching M with set of forced pairs Q and set of forbidden pairs P. Let S be the closed subset of $\Pi(\Lambda)$ corresponding to M. The existence of stable matching M says every pair $(m,w) \in Q \setminus M_0$ is a stable pair and so admits a rotation $\gamma(m,w)$. Let Γ be the set of γ-rotations for all pairs in $Q \setminus M_0$. Also the existence of M implies there is no pair $(m,w) \in P \cap M_0 \cap M_z$, as such a pair would be present in every stable matching. Let Θ be the set of θ-rotations for all stable pairs $(m,w) \in P \setminus M_z$. Clearly, $X = (\Theta \cup \Gamma) \cap S$ satisfies the requirements (i) and (ii).

($\Leftarrow$) Let S be the closed subset of $\Pi(\Lambda)$ such that the maximal rotations in S are the rotations that are maximal in X with respect to the predecessor relation $\preceq$. Let M be the stable matching that corresponds to S. Clearly, M is the desired stable matching with set of forced pairs Q and set of forbidden pairs P. ■

Next we show how the characterization presented in the above theorem leads to an algorithm that tests for the existence of a solution of the STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS problem with forced pairs Q and forbidden pairs P in $O((|Q| + |P|)^2)$ time, after pre-processing the preference lists in $O(n^4)$ time.

We can test within this time bound whether the desired set X of rotations used in the characterization of Theorem 4 exists by processing a list $\mathcal{L}$ of rotations as follows. First we deal with some trivial situations, where the answer is obtained in the pre-processing phase and there is no need to construct a set X. Clearly, we may assume we have a set of forced pairs Q such that every pair is stable. Also, a non stable pair in P is a forbidden pair for any stable matching. Thus, we may remove from P all non stable pairs and assume we have a set of forbidden pairs P such that every pair is stable. In addition, if P contains a pair present both in M_0 and in M_z (this means the pair belongs to every stable matching), then clearly there is no solution.

Denote by $(m_i, w_i), 1 \leq i \leq q$, the forced pairs in Q. Denote by γ_i the γ-rotation of pair $(m_i, w_i) \in Q \setminus M_0$, and call this set of rotations Γ. Denote by θ_i the θ-rotation of pair $(m_i, w_i) \in Q \setminus M_z$, and call this set of rotations Θ. Denote by $(m_i', w_i'), 1 \leq i \leq p$, the forbidden pairs in P. Denote by θ_i' the θ-rotation of a stable pair $(m_i', w_i') \in P \setminus M_z$, and call this set of rotations Θ'. Denote by γ_i' the γ-rotation of a stable pair $(m_i', w_i') \in P \setminus M_0$, and call this set of rotations Γ'. By hypothesis, no pair of P belongs both to M_0 and M_z. Clearly, as in the case of the algorithm for forced pairs presented in [5], the pre-processing of the preference lists identifies the stable pairs, completely constructs $\Pi(\Lambda)$, and determines $\gamma(m, w)$ and $\theta(m, w)$, for each stable pair (m, w).

Now construct and process a list $\mathcal{L}$ of rotations as follows. Begin by adding to $\mathcal{L}$ all $\gamma(m, w)$, for every pair $(m, w) \in Q \setminus M_0$, and all $\theta'(m, w)$, for every pair $(m', w') \in P \cap M_0$. Process each rotation ρ of $\mathcal{L}$ by adding ρ to set X, and by testing, for each θ_i, a θ-rotation in Θ, whether $\theta_i \preceq \rho$. If yes, then stop: there is no desired set X. Else, test, for each γ_i', a γ-rotation in Γ', whether $\gamma_i' \preceq \rho$. If yes and $(m_i', w_i') \in M_z$, then stop: there is no desired set X. If yes and $(m_i', w_i') \notin M_z$, then move θ_i' from set Θ' to $\mathcal{L}$ and remove γ_i' from set Γ'. Continue by processing the rotations according to their rank in $\mathcal{L}$. In the case that all rotations in list $\mathcal{L}$ are successfully processed, then we have the desired set X built in time $O((|Q| + |P|)^2)$. As in the case of forced pairs [5], this algorithm first constructs $\Pi(\Lambda)$ explicitly, and so pre-processes the preference lists in $O(n^4)$ time.

We proceed to the transformation from STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS to STABLE MARRIAGE WITH FORBIDDEN PAIRS. Given an instance (n, L) of the original STABLE MARRIAGE problem, n is the number of men and L is the set of $2n$ preference lists. In an instance (n, L, Q, P) of STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS, Q is the set of forced pairs and P is the set of forbidden pairs. We reduce STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS to STABLE MARRIAGE WITH FORBIDDEN PAIRS as follows [2].

Let (n, L, Q, P) be an instance of STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS. Begin by setting $P' = P$, and for each pair $(m, w) \in Q$, add (m, w') to P', for all $w' \neq w$. A matching is stable with respect to (n, L, Q, P), if and only if it is stable with respect to $(n, L, \emptyset, P')$.

Note that the above reduction constructs an instance of STABLE MARRIAGE WITH FORBIDDEN PAIRS with $(|P| + (n-1)|Q|)$ forbidden pairs. This observation justifies the characterization of Theorem 4 being stated for STABLE MARRIAGE WITH FORCED AND FORBIDDEN PAIRS.

In Section 4, we focus on the STABLE MARRIAGE WITH FORBIDDEN PAIRS problem. We denote an instance of STABLE MARRIAGE WITH FORBIDDEN PAIRS by (n, L, P), where n is the number of men, L is the set of $2n$ preference lists, and P is the set of forbidden pairs. We shall describe in Section 4 algorithms for STABLE MARRIAGE WITH FORBIDDEN PAIRS that find a stable matching, if it exists, in time $O(n^2)$.

4 Optimal Algorithms for Restricted Pairs

Algorithm for finding a stable matching

We use the operation *breakmarriage* [7] to decide in time $O(n^2)$, given an instance of STABLE MARRIAGE WITH FORBIDDEN PAIRS, whether it admits a stable matching. Given a stable matching M, containing the pair (m, w), operation $breakmarriage(M, (m, w))$ returns the man-optimal stable matching which is dominated by M and does not contain the pair (m, w), if it exists.

The following algorithm [2] finds the man-optimal stable matching with a set of forbidden pairs P. We call this matching M_0^P. Note that Algorithm 1 may also find, if changed accordingly, the woman-optimal stable matching with a set of forbidden pairs P. We call this matching M_z^P.

```
Algorithm 1
Input: (n, L, P)
Output: The man-optimal stable matching with a set of forbidden pairs P,
if it exists, and "There is no solution" otherwise
M ← man-optimal solution without considering P
while there is a forbidden pair (m, w) in M
    M ← breakmarriage(M, (m, w))
    if M is not a matching
        return "There is no solution"
return M
```

Theorem 5 *Algorithm 1 decides in time $O(n^2)$ whether a given instance of* STABLE MARRIAGE WITH FORBIDDEN PAIRS *admits a stable matching, and returns the man-optimal solution if it exists.*

Proof. The proof of correctness is straightforward. Let $M_1, M_2, ..., M_k$ be the matchings assumed by variable M during the execution of the algorithm. If the solution exists, every matching M_i dominates or is equal to it. As M_i dominates M_{i+1}, matching M_k is the solution. If there is no solution we will certainly try to break a forbidden pair of P that is in the unrestricted woman-optimal solution M_z and *breakmarriage* will return an error.

For the complexity analysis, first note that we can determine in constant time whether a given pair is forbidden by checking a pre-built boolean matrix. We can maintain a list of all forbidden pairs in the current matching by checking the boolean matrix during all changes of pairs in the matching and adding or removing a pair from the list accordingly. To add or remove these elements in constant time, it is necessary to maintain another matrix, which points to the position of each forbidden pair in the list. It is clear by [7] that the total time spent in the breakmarriage operation is bounded by the total number of proposals performed within the operation. Since the operation does not make the same proposal twice, this number is $O(n^2)$. ■

Algorithm for all stable pairs

The relevant results from Section 1.3.1 of [5] can be extended to STABLE MARRIAGE WITH FORBIDDEN PAIRS so that the set of stable matchings form a lattice. Given an instance (n, L, P) of STABLE MARRIAGE WITH FORBIDDEN PAIRS, call Λ_P the lattice of solutions of the version with forbidden pairs and Λ the lattice of solutions of the unrestricted version obtained by removing the set P.

First, we use Algorithm 1 to obtain stable matchings M_0^P and M_z^P (we assume that M_0^P and M_z^P exist, otherwise we may halt immediately). Then, we consider only rotations in the maximal chains in Λ between M_0^P and M_z^P to construct the corresponding subgraph of $G(\Lambda)$. We construct the digraph $G'(\Lambda_P)$ by adding edges to this subgraph of $G(\Lambda)$. The digraph $G'(\Lambda_P)$ contains, for each forbidden pair $(m, w) \in P$, the directed edge $(\theta(m, w), \gamma(m, w))$, if $\theta(m, w)$ and $\gamma(m, w)$ are rotations in the maximal chains in Λ between M_0^P and M_z^P. Note that these additional edges add cycles to the acyclic digraph $G(\Lambda)$. The number of edges added to the subgraph of $G(\Lambda)$ to obtain $G'(\Lambda_P)$ is $O(n^2)$.

We extend the definition of closed set to digraphs as follows: a *closed* set in a digraph G is a subset S of the vertex set of G such that if $v \in S$ and there is a directed path from vertex w to vertex v, then $w \in S$.

Theorem 6 *There is a one to one correspondence between the stable matchings of Λ_P and the closed subsets of $G'(\Lambda_P)$.*

Proof. Given a closed subset of $G'(\Lambda_P)$, the corresponding stable matching is obtained by the elimination of every rotation in the subset starting from M_0^P. First, we show that the corresponding matchings are, in fact, stable. Then we show that every stable matching can be generated this way.

Clearly, by Corollary 3.2.2 of [5] and Theorem 1, all these matchings are in Λ. If a stable matching generated in this way contained a pair (m, w) from P, then the corresponding subset would contain $\gamma(m, w)$ and would not contain $\theta(m, w)$. Since there is an edge $(\theta(m, w), \gamma(m, w))$, this subset is not closed.

To show that every matching in Λ_P has a corresponding closed subset of $G'(\Lambda_P)$, we suppose there is a matching $M \in \Lambda_P$ that contradicts this assumption. There is a closed subset S of $G(\Lambda)$ that corresponds to M. Consequently, for some rotations $\pi \in S$ and $\pi' \notin S$, the edge (π', π) is in $G'(\Lambda_P)$, but not in $G(\Lambda)$. So, (π', π) is $(\theta(m, w), \gamma(m, w))$ for a forbidden pair (m, w). Therefore, the forbidden pair $(m, w) \in M$, contradicting its stability. ■

Let M be a stable matching and consider a set of rotations S that can be eliminated in M consecutively, resulting in the stable matching M'. A *transformation* τ is a set of triples (m, w, w') corresponding to S. For each man m which is married to a woman w in M and another woman $w' \neq w$ in M', the corresponding transformation contains a triple (m, w, w'). We denote by M/τ the stable matching M' obtained by elimination of τ: for each $(m, w, w') \in \tau$, we have $(m, w) \in M$ and $(m, w') \in M/\tau$, the other pairs are the same in M and M/τ. If there is just one rotation in the set, say $\pi = (m_0, w_0), (m_1, w_1), \ldots, (m_{r-1}, w_{r-1})$, then the corresponding transformation is $\tau = \{(m_0, w_0, w_1), (m_1, w_1, w_2), \ldots, (m_{r-1}, w_{r-1}, w_0)\}$. We shall study next the lattice of solutions Λ_P of a STABLE MARRIAGE WITH FORBIDDEN PAIRS instance.

We define the poset of transformations $\Pi(\Lambda_P)$ and construct its compact representations $G(\Lambda_P)$ and $\tilde{G}(\Lambda_P)$ analogously to the poset of rotations $\Pi(\Lambda)$ and its compact representations $G(\Lambda)$ and $\tilde{G}(\Lambda)$. The elements of the poset $\Pi(\Lambda_P)$, and the vertices of the digraphs $G(\Lambda_P)$ and $\tilde{G}(\Lambda_P)$ are the transformations corresponding to the strongly connected components of $G'(\Lambda_P)$.

Transformation τ precedes τ' in $\Pi(\Lambda_P)$ if and only if π precedes π' in $\Pi(\Lambda)$, for some rotation π belonging to τ and for some rotation π' belonging to τ'.

Theorem 7 *There is a one to one correspondence between the stable matchings of Λ_P and the closed subsets of $\Pi(\Lambda_P)$.*

Proof. It is easy to verify that there is a one to one correspondence between the closed subsets of $\Pi(\Lambda_P)$ and the closed subsets of $G'(\Lambda_P)$. Now Theorem 6 implies that there is also a one to one correspondence between the closed subsets of $\Pi(\Lambda_P)$ and the stable matchings of Λ_P. ■

We establish a result analogous to Theorem 2.

Theorem 8 *A pair (m, w) is a stable pair in Λ_P if and only if it is a pair in M_0^P or (m, w', w) belongs to a transformation in $\Pi(\Lambda_P)$. Equivalently, (m, w) is stable in Λ_P if and only if it is a pair in M_z^P or (m, w, w') belongs to a transformation in $\Pi(\Lambda_P)$.*

Proof. It is enough to prove the first version of the theorem. The proof of the second version is analogous. Let (m, w) be a pair such that (m, w', w) belongs to a transformation τ. Let M be the stable matching corresponding to the smallest closed subset of Λ_P containing τ. The stable matching M is a proof of the stability of the pair (m, w).

Conversely, let (m, w) be a stable pair that does not belong to M_0. Let M be a stable matching in Λ_P containing (m, w). By Theorem 7, there exists a closed subset S of $\Pi(\Lambda_P)$ corresponding to M. Now since $(m, w) \notin M_0$, there exists a transformation in S containing (m, w', w). ■

The above characterization yields an $O(n^2)$ time algorithm that given an instance of STABLE MARRIAGE WITH FORBIDDEN PAIRS finds all stable pairs. Notice that, in order to find all stable pairs, it is not necessary to construct $G(\Lambda_P)$, but only to determine its vertices, the transformations corresponding to the strongly connected components of $G'(\Lambda_P)$.

Given a strongly connected component S of $G'(\Lambda_P)$, to construct the corresponding transformation it is first necessary to find a valid order by which the rotations of S can be eliminated. To do that, we must consider the subgraph of $G(\Lambda)$ induced by the vertices of S (in other words, we must remove from consideration the edges which created cycles in S). Any topological order of the vertices of this acyclic digraph is a valid order by which the rotations can be eliminated. To construct the actual transformation it is sufficient to simulate the elimination of these rotations and list the modified pairs.

Algorithm for all stable matchings

The definition of the edges of $G(\Lambda_P)$ is analogous to the definition of the edges of $G(\Lambda)$ given in [5]. There are two types of edges:

Type 1: If $(m, w', w) \in \tau$ and $(m, w, w'') \in \tau'$, then (τ, τ') is a type 1 edge.

Type 2: If the transformation τ moves a woman w from a man worse than m to a man better than m and the transformation τ' moves m from a woman better than w to a woman worse than w, then (τ, τ') is a type 2 edge.

Theorem 9 *If (τ, τ') is in $G(\Lambda_P)$, then (τ, τ') is in $\Pi(\Lambda_P)$.*

Proof. We follow a similar argument to the proof of Lemma 3.2.3 in [5]. If (τ, τ') is a type 1 edge in $G(\Lambda_P)$, it is clear that τ must be eliminated before τ', so (τ, τ') is in $\Pi(\Lambda_P)$.

If (τ, τ') is a type 2 edge in $G(\Lambda_P)$, there is a pair (m, w) such that τ takes w from a man worse than m to a man better than m and τ' takes m from a woman better than w to a woman worse than w. The pair (m, w) blocks any matching obtained by the elimination of τ' without the elimination of τ, so (τ, τ') is in $\Pi(\Lambda_P)$. ■

We say that τ is an *immediate predecessor* of τ' in $\Pi(\Lambda_P)$ if there is no τ'' such that τ precedes τ'' and τ'' precedes τ'.

Theorem 10 *If τ is an immediate predecessor of τ' in $\Pi(\Lambda_P)$, then (τ, τ') is in $G(\Lambda_P)$.*

Proof. We follow a similar argument to the proof of Lemma 3.2.4 in [5]. By Theorem 7, let M be the stable matching corresponding to the closed set of all transformations t such that $(t, \tau) \in \Pi(\Lambda_P)$. M/τ is also a stable matching. As τ is a immediate predecessor of τ', $M/\tau/\tau'$ is also a stable matching, but M/τ' is not.

As $M/\tau/\tau'$ is a stable matching, but M/τ' is not, one of the following conditions occurs: There is a pair created by τ and broken by τ' or there is a pair (m, w) such that τ takes w from a man worse than m to a man better than m and τ' takes m from a woman better than w to a woman worse than w. In the former case, (τ, τ') is a type 1 edge in $G(\Lambda_P)$ and in the latter case (τ, τ') is a type 2 edge in $G(\Lambda_P)$. ■

An immediate consequence of the last two theorems is:

Theorem 11 *The transitive closure of $G(\Lambda_P)$ is $\Pi(\Lambda_P)$. Consequently, there exists a one to one correspondence between the closed subsets of $G(\Lambda_P)$ and the stable matchings of Λ_P.*

The following algorithm constructs $G(\Lambda_P)$ by extending to the context of transformations the algorithm for the construction of $G(\Lambda)$ suggested by the proof of Lemma 3.3.2 of [5]. An argument similar to the one in [5] establishes the time complexity bound of $O(n^2)$ for the construction of $G(\Lambda_P)$.

```
Algorithm 2
Input: (n, L, P), M_z^P and the set of transformations
Output: The edges of G(Λ_P)
(Phase 1)
V[m,w] ← 0, for every pair (m,w)
For each transformation τ
    For each (m,w,w') ∈ τ
        V[m,w] ← 1
        T[m,w] ← τ
    For each (m,w) such that τ moves w from a man
    worse than m to a man better than m
        V[m,w] ← 2
        T[m,w] ← τ
For each (m,w) ∈ M_z^P
    V[m,w] ← #
(Phase 2)
For each man m
    t ← 0
    For each woman w following the order of preference of m
        If V[m,w] = #
            Proceed to the next man
        If V[m,w] = 1
            If t ≠ 0
                Output type 1 edge (t, T[m,w])
            t ← T[m,w]
        If V[m,w] = 2
            If t ≠ 0
                Output type 2 edge (T[m,w], t)
```

Phase 1 of the algorithm assigns labels to the pairs. During phase 2, we scan these labels on the preference lists of each man. When $V[m,w] = 1$ (the same for $V[m,w] = 2$) and $T[m,w] = \tau$ we say that there is a type 1 (type 2) label of τ.

Exploring the closed subsets of $G(\Lambda_P)$ using the same algorithm used in [5] will list all stable matchings, but the time complexity will not be optimal, because some vertices may have an outdegree greater than n. To solve this problem we extend to the context of transformations the method used in [5] to obtain $\tilde{G}(\Lambda)$ from $G(\Lambda)$ and define another digraph, $\tilde{G}(\Lambda_P)$, which has some of the type 2 edges of $G(\Lambda_P)$ removed. The only difference between $\tilde{G}(\Lambda_P)$ and $G(\Lambda_P)$ is that, in phase 2 of the algorithm, if we find two labels of the same transformation τ during the scan of the preference list of a man m, we only consider the first label and the additional type 1 labels. In other words, we do not consider type 2 labels that come after a type 1 or type 2 label of the same transformation. Phase 2 of Algorithm 2 should be rewritten as follows:

```
Algorithm 3
Input: (n, L, P), M_z^P and the set of transformations
Output: The edges of G~(Λ_P)

(Phase 1)
The same as Algorithm 2
(Phase 2)
p[τ] ← 0, for every transformation τ
For each man m
    t ← 0
    For each woman w following the order of preference of m
        If V[m,w] = #
            Proceed to the next man
        If V[m,w] = 1
            If t ≠ 0
                Output type 1 edge (t, T[m,w])
            t ← T[m,w]
            p[T[m,w]] ← 1
            Add T[m,w] to a list
        If V[m,w] = 2
            If t ≠ 0 and p[T[m,w]] = 0
                Output type 2 edge (T[m,w], t)
            p[T[m,w]] ← 1
            Add T[m,w] to a list
    p[τ] ← 0, for every τ on the list
    Empty the list
```

The next two theorems may be proved in a similar manner to Parts (i) and (ii) of Theorem 3.3.1 in [5].

Theorem 12 *The outdegree of any node in $\tilde{G}(\Lambda_P)$ is at most n.*

Proof. Type 1 labels of a transformation τ can appear only once in the preference list of any man. So, at most one type 1 edge is created during the scan of the preference list of each man. Since type 2 labels of τ are ignored if preceded by earlier labels of τ and a type 2 label of τ cannot precede the unique type 1 label of τ, at most one edge out of τ is created during the scan of the preference list of each man. ■

Theorem 13 *The transitive closure of $\tilde{G}(\Lambda_P)$ is $\Pi(\Lambda_P)$.*

Proof. It is sufficient to prove that if (τ, τ') is a type 2 edge of $G(\Lambda_P)$, but not of $\tilde{G}(\Lambda_P)$, there is a path from τ to τ' in $\tilde{G}(\Lambda_P)$. This edge (τ, τ') has been ignored when we were scanning the preference list of man m in Algorithm 3 because there is another label of τ before the type 1 label of τ'. But there is a type 1 edge (τ, τ'') created because there is a type 1 label of τ'' before the type 1 label of τ'. As there is a path through type 1 edges from τ'' to τ', there is a path from τ to τ' in $\tilde{G}(\Lambda_P)$. ■

Exploring the closed subsets of $\tilde{G}(\Lambda_P)$ involves extending to the context of transformations the algorithm of Figure 3.8 of [5] which will list all stable matchings in optimal worst case time. The only necessary change is that instead of eliminating rotations, we must eliminate transformations. The space complexity is $O(n^2)$. Summarizing, the proposed algorithm to find all stable matchings consists of:

Algorithm 4
Input: (n, L, P)
Output: All stable matchings with set of forbidden pairs P
Construct $G(\Lambda)$
Add edges constructing $G'(\Lambda_P)$
Find the strongly connected components of $G'(\Lambda_P)$
and the corresponding transformations
Construct $\tilde{G}(\Lambda_P)$
Explore all closed subsets of $\tilde{G}(\Lambda_P)$
and list the corresponding matchings

5 Concluding remarks

We have described an algorithm for finding all S solutions, given an instance of STABLE MARRIAGE WITH FORBIDDEN PAIRS, with n men and n women. The time complexity of the algorithm is $O(n^2+nS)$ while the space complexity is $O(n^2)$.

It would be interesting to know whether there is an algorithm that could solve the above problem in less than $O(n)$ amortized time per solution for a sufficiently large value of S. In general, most of the algorithms (e.g. [6]) for enumerating the set of size n objects of a desired collection achieving efficient amortized time bounds work under the following model: *the objects of the collection are enumerated in a same memory space, of size n.* The following argument shows that any algorithm for explicitly finding all solutions of STABLE MARRIAGE (with or without forbidden pairs) requires $\Omega(n)$ amortized time per solution, under the above model.

Denote by $\eta(n)$ the following instance of STABLE MARRIAGE with n men and n women. Let $L(m,k)$ be the k-th woman on man m's list and $L(w,k)$ the k-th man on woman w's list. The preference lists in $\eta(n)$ are: $L(m_i,k) = w_{i+k-1}$, $L(w_i,k) = m_{i+k}$, where indices are taken modulo n.

The n stable matchings for instance $\eta(n)$ are precisely, for each fixed value of $k = 1,\ldots,n$, the set of n pairs: $\{(m, L(m,k)),$ for every man $m\}$. First, it is clearly true that for $k = 1$, we have a stable matching because every man is married to the first woman on his list and every man is married to a distinct woman. By searching an exposed rotation in this matching, we find the rotation $((m_1,w_1),(m_2,w_2),\ldots,(m_n,w_n))$. The following matching will have the exposed rotation $((m_1,w_2),(m_2,w_3),...,(m_n,w_1))$, and so on. We generate in this way n distinct stable matchings for instance $\eta(n)$. Note that each one of the possible n^2 pairs is a stable pair, and that any pair is in precisely one stable matching. Therefore instance $\eta(n)$ admits precisely n distinct stable matchings, and is such that any of its two stable matchings have no common pairs, i.e., any two stable matchings differ by n pairs.

Next, we show how to construct an instance of STABLE MARRIAGE with $n = n_1 n_2$ men, $n_1^{n_2}$ stable matchings and such that any two distinct solutions differ by n_1 pairs, for any n_1 and n_2. By considering n_2 to be a constant, we need $\Omega(n)$ time to write the different pairs in memory.

To construct this instance we take n_2 instances $\eta(n_1)$ for different sets of n_1 men. Only the first n_1 positions on each list are filled, but the other ones can be filled arbitrarily, because they will not be used in any stable matching. The n_1 distinct stable matchings for each instance can be freely combined yielding the claimed $n_1^{n_2}$ stable matchings.

Some variations of the STABLE MARRIAGE problem (like incomplete lists or different sized sets) can be reduced to the conventional problem by changing the preference lists and possibly adding auxiliary men and women. The solutions of the variations are found by removing these auxiliary people from the obtained stable matchings. We show next that this kind of simple reduction is not possible for STABLE MARRIAGE WITH FORBIDDEN PAIRS. The set of solutions for the instance $\eta(4)$ described above with forbidden pairs $P = \{(1,2),(1,4)\}$ is: $\{\{(1,1),(2,2),(3,3),(4,4)\},\{(1,3),(2,4),(3,1),(4,2)\}\}$. If these two matchings were solutions of the STABLE MARRIAGE problem, then they would be connected by a single rotation, which is not the case. A rotation is a cyclic permutation of one subset of women among one subset of men. In this example, two rotations are necessary to exchange the wives of men 1 and 3, and of men 2 and 4. The need of two rotations remains if auxiliary men and women are added.

Acknowledgements We are grateful to the referee for his careful reading and many suggestions which helped to improve the content and organization of the paper.

References

[1] V. M. F. Dias (2000). "Stable Matchings with Restricted Pairs", Master dissertation, COPPE, Universidade Federal do Rio de Janeiro (In Portuguese).

[2] G. D. da Fonseca (2000). "Stable Marriages with Forbidden Pairs", Graduation dissertation, Instituto de Matemática, Universidade Federal do Rio de Janeiro (In Portuguese).

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

[4] D. Gusfield, *Three fast algorithms for four problems in stable marriage*, SIAM J. Comput., 16 (1987), pp. 111–128.

[5] D. Gusfield and R. W. Irving (1989). "The Stable Marriage Problem – Structure and Algorithms", The MIT Press.

[6] M. Habib, R. Medina, L. Nourine, G. Steiner, *Efficient algorithms on distributive lattices*, Discrete Appl. Math., 110 (2001), pp. 169–187.

[7] D. McVitie and L. Wilson, *The stable marriage problem*, Comm. ACM, (July 1971), pp. 486–490.

[8] D. E. Knuth (1997). "Stable Marriage and Its Relation to Other Combinatorial Problems", American Mathematical Society.

[9] R. W. Irving and P. Leather, The complexity of counting stable marriages, SIAM J. Comput., 15 (1986), pp. 654–667.