Modelling and Solving the Stable Marriage Problem Using Constraint Programming

David F. Manlove*,† and **Gregg O'Malley***

Department of Computing Science, University of Glasgow, Glasgow G12 8QQ, UK.
Email: {davidm,gregg}@dcs.gla.ac.uk.

Abstract

We study the Stable Marriage problem (SM), which is a combinatorial problem that arises in many practical applications. We present two new models of an instance I of SM with n men and n women as an instance J of a Constraint Satisfaction Problem. We prove that establishing arc consistency in J yields the same structure as given by the established Extended Gale/Shapley algorithm for SM as applied to I. Consequently, a solution (stable matching) of I can be derived without search. Furthermore we show that, in both encodings, *all* stable matchings in I may be enumerated in a failure-free manner. Our first encoding is of $O(n^3)$ complexity and is very natural, whilst our second model, of $O(n^2)$ complexity (which is optimal), is a development of the Boolean encoding in [Gent *et al.*, 2001], establishing a greater level of structure.

1 Introduction

The classical Stable Marriage problem (SM) has been the focus of much attention in the literature over the last few decades [Gale and Shapley, 1962; Knuth, 1976; Gusfield and Irving, 1989; Roth and Sotomayor, 1990]. An instance of SM comprises n men, $m_1, \ldots, m_n$, and n women, $w_1, \ldots, w_n$, and each person has a preference list in which they rank all members of the opposite sex in strict order. A matching M is a bijection between the men and women. We denote the partner in M of a person q by $M(q)$. A (man,woman) pair (m_i, w_j) *blocks* a matching M, or forms a *blocking pair* of M, if m_i prefers w_j to $M(m_i)$ and w_j prefers m_i to $M(w_j)$. A matching that admits no blocking pair is said to be *stable*, otherwise the matching is *unstable*. SM and its variants arise in important practical applications, such as the annual match of graduating medical students to their first hospital appointments in a number of countries (see e.g. [Roth, 1984]).

Gale and Shapley [Gale and Shapley, 1962] showed that every instance I of SM admits a stable matching, and gave an $O(n^2)$ algorithm, linear in the instance size, for finding such a matching in I. A modified version of this algorithm – the Extended Gale/Shapley (EGS) algorithm [Gusfield and Irving, 1989, Section 1.2.4] – avoids some unnecessary steps by deleting from the preference lists certain (man,woman) pairs that cannot belong to a stable matching. Moreover the EGS algorithm aids the development of some useful structural properties of SM [Gusfield and Irving, 1989, Section 1.2.4]. The *man-oriented* version of the EGS algorithm (henceforth referred to as the MEGS algorithm) involves a sequence of proposals from the men to the women, provisional engagements between men and women, and deletions from the preference lists. A pseudocode description of MEGS algorithm is given in Figure 1 (the term *delete the pair* (p, w) means that p should be deleted from w's list and vice versa.) The stable matching returned by the MEGS algorithm is called the *man-optimal* (or equivalently, *woman-pessimal*) stable matching, denoted by M_0, since each man has the best partner (according to his ranking) that he could obtain, whilst each woman has the worst partner that she could obtain, in any stable matching. A similar proposal sequence from the women to the men yields the *woman-oriented* EGS (WEGS) algorithm. This gives rise to the *woman-optimal* (or *man-pessimal*) stable matching, denoted by M_z, with analogous properties.

Upon termination of the MEGS algorithm, the reduced preference lists that arise following the deletions are referred to as the *MGS-lists*. Similarly, the *WGS-lists* arise upon termination of the WEGS algorithm. The intersection of the MGS-lists with the WGS-lists yields the *GS-lists* [Gusfield and Irving, 1989, p.16]. Some important structural properties of the GS-lists are given by the following theorem.

Theorem 1 ([Gusfield and Irving, 1989, Theorem 1.2.5]). *For a given instance of SM:*

(i) all stable matchings are contained in the GS-lists;

(ii) no matching M contained in the GS-lists can be blocked by a pair that is not in the GS-lists;

(iii) in the man-optimal (respectively woman-optimal) stable matching, each man is partnered by the first (respectively last) woman on his GS-list, and each woman by the last (respectively first) man on hers.

*Supported by Engineering and Physical Sciences Research Council grant GR/R84597/01.

†Supported by Royal Society of Edinburgh/Scottish Executive Personal Research Fellowship.

assign each person to be free;
while some man m is free **and** m has a nonempty list **loop**
  w := first woman on m's list; $\{m$ 'proposes' to $w\}$
  if some man p is engaged to w **then**
    assign p to be free;
  end if;
  assign m and w to be engaged to each other;
  for each successor p of m on w's list **loop**
    delete the pair (p, w);
  end loop;
end loop;

Figure 1: The man-oriented Extended Gale/Shapley algorithm for SM and SMI.

An example SM instance I is given in Figure 2. (We assume that a person's preference list is ordered with his/her most-preferred partner leftmost.) This figure also indicates those preference list entries that belong to the GS-lists. In I, the man-optimal stable matching M_0 and the woman-optimal stable matching M_z are as follows:

$$M_0 = \{(m_1, w_1), (m_2, w_3), (m_3, w_2), (m_4, w_4)\}$$
$$M_z = \{(m_1, w_3), (m_2, w_1), (m_3, w_4), (m_4, w_2)\}.$$

The extension SMI of SM arises when preference lists may be incomplete. This occurs when a person may find a member of the opposite sex *unacceptable*. If a person p finds a person q unacceptable, q does not appear on the preference list of p. In the SMI case, a matching M in an instance I of SMI is a one-one correspondence between a subset of the men and a subset of the women, such that $(m, w) \in M$ implies that each of m and w finds the other acceptable. Given a matching M in an SMI instance, a pair (m, w) blocks a matching M if each of m and w finds the other acceptable, and each is either unmatched in M or prefers the other to their partner in M. If a person p finds a person q unacceptable, then p and q cannot be paired in any stable matching, nor can they form a blocking pair. Hence, from the point of view of finding stable matchings, we lose no generality by assuming that q finds p unacceptable also, so that preference lists are *consistent*. It is straightforward to adapt the EGS algorithm to the SMI case [Gusfield and Irving, 1989, Section 1.4.2] – see Figure 1 for a pseudocode description. The woman-oriented algorithm is analogous. In the SMI context a stable matching need not be complete; however the same set of men and women are matched in all stable matchings [Gale and Sotomayor, 1985]. Furthermore, the concept of GS-lists can be extended to SMI, with analogous properties (for Property (ii) in Theorem 1, each person with a non-empty GS-list should be matched in M; for Property (iii), each person with an empty GS-list is unmatched in both stable matchings).

Men's lists	Women's lists
m_1: w_2 w_4 $\underline{w_1}$ $\underline{w_3}$	w_1: $\underline{m_2}$ $\underline{m_4}$ m_3 $\underline{m_1}$
m_2: $\underline{w_3}$ w_4 $\underline{w_1}$ w_2	w_2: $\underline{m_4}$ $\underline{m_3}$ m_1 m_2
m_3: $\underline{w_2}$ $\underline{w_4}$ w_1 w_3	w_3: m_3 m_4 $\underline{m_1}$ $\underline{m_2}$
m_4: $\underline{w_4}$ $\underline{w_1}$ $\underline{w_2}$ w_3	w_4: $\underline{m_3}$ $\underline{m_4}$ m_2 m_1

Figure 2: An SM instance with 4 men and 4 women; preference list entries that belong to the GS-lists are underlined.

1.1 Related work

The Stable Marriage problem has its roots as a combinatorial problem, but has also been the subject of much interest from the Game Theory and Economics community [Roth and Sotomayor, 1990] and the Operations Research community [Vate, 1989]. In recent years SM and SMI have also been the focus of interest from the Constraint Programming community [Aldershof and Carducci, 1999; Dye, 2001; Gent *et al.*, 2001; Lustig and Puget, 2001; Gent and Prosser, 2002a; 2002b; Green and Cohen, 2003; Thorn, 2003]. These papers have presented a range of encodings of SM and its variants as an instance of a Constraint Satisfaction Problem (CSP). In all references apart from [Gent *et al.*, 2001], structural relationships between the effect of Arc Consistency (AC) propagation [Bessière and Régin, 1997] and the GS-lists were not explored in detail, nor did the authors consider the aspect of failure-free enumeration.

However such issues were considered by Gent et al. [Gent *et al.*, 2001], who proposed two CSP encodings of SMI. For each model, it was shown that AC propagation can be used to achieve similar results to the EGS algorithm in a certain sense. The first encoding creates a CSP instance J_1 using a set of 'conflict matrices' to encode an SMI instance I. In J_1, AC may be established in $O(n^4)$ time, following which the variables' domains correspond to the GS-lists of I. The second encoding creates a Boolean CSP instance J_2. In J_2, AC may be established in $O(n^2)$ time, however the variables' domains after AC propagation only correspond to a weaker structure called the *XGS-lists* in I, which in general are supersets of the GS-lists in I. (The XGS-list for a person p consists of all entries in p's preference list between the first and last entries of his/her GS-list inclusive.) In both encodings the set of all stable matchings in I can be enumerated in a failure-free manner (using a value-ordering heuristic in the case of the first encoding).

1.2 Our contribution

The work of [Gent *et al.*, 2001] left open the question as to whether there exists an $O(n^2)$ CSP encoding of SM that captures exactly the structure of the GS-lists. In this paper we present two encodings of an instance I of SMI (and so of SM) as a CSP instance J. Again, for each encoding, we show that AC propagation achieves the same results as the EGS algorithm in a precise sense. The first model is a natural $(n + 1)$-valued encoding of SMI; it bears some resemblance to the encoding of SM given in [Lustig and Puget, 2001] and develops the 'conflict matrices' model of [Gent *et al.*, 2001]. In this model we show that AC propagation may be carried out in $O(n^3)$ time. Our model is more intuitive, and is more time and space-efficient, than the 'conflict matrices' model. Our second model is a more compact 4-valued encoding that develops the Boolean encoding from [Gent *et al.*, 2001] – in this case we show that AC propagation may be carried out in $O(n^2)$ time. For both models we prove that the GS-lists in I correspond to the domains remaining after establishing AC in J. Furthermore, we show that, for both encodings, we are guaranteed a failure-free enumeration of all stable matchings in I using AC propagation combined with a value-ordering heuristic in J. Our second encoding therefore answers the question left open by [Gent *et al.*, 2001].

Our results show that, provided the model is chosen carefully, AC propagation within a CSP formulation of SMI captures the structure produced by the EGS algorithm. Moreover our second encoding indicates that AC propagation can be achieved within the same time complexity as the (optimal) MEGS algorithm for SMI, producing equivalent structural results. This strengthens the assertion in [Gent *et al.*, 2001] regarding the applicability of constraint programming to the general domain of stable matching problems. Furthermore, in many practical situations there may be additional constraints that cannot be accommodated by a straightforward modification of the EGS algorithm. Such constraints could however be built on top of either of the two models that we present here. Possible extensions could arise from variants of SMI that are NP-hard [Ronn, 1990; Ng and Hirschberg, 1991; Kato, 1993; Manlove *et al.*, 2002].

We remark that, independently, Unsworth and Prosser have formulated a specialised n-ary constraint for SMI, such that AC propagation gives rise to the GS-lists, where the complexity of establishing AC is $O(n^2)$ [Unsworth and Prosser, 2005a]. They have also constructed a specialised binary constraint for SMI that yields the same structure, where AC may be established in $O(n^3)$ time [Unsworth and Prosser, 2005b]. In both cases, all stable matchings may be generated using a failure-free enumeration.

The remainder of this paper is organised as follows. Section 2 contains the $(n+1)$-valued encoding. We show that AC may be established in $O(n^3)$ time, proving the structural relationship between AC propagation and the GS-lists. This is followed by the failure-free enumeration result for this model. In Section 3 we present the 4-valued encoding, following a similar approach, however in this case we show that AC may be established in $O(n^2)$ time. Finally, Section 4 contains some concluding remarks.

2 $(n+1)$-valued encoding

2.1 Overview of the encoding

In this section we present an $(n+1)$-valued binary CSP encoding for an instance I of SMI. We assume that $\mathcal{M} = \{m_1, m_2, \ldots, m_n\}$ is the set of men and $\mathcal{W} = \{w_1, w_2, \ldots, w_n\}$ is the set of women in I (it is not difficult to extend our encoding to the case that the numbers of men and women are not equal, but for simplicity we assume that they are equal). For each man $m_i \in \mathcal{M}$ and woman $w_j \in \mathcal{W}$, the length of m_i's and w_j's preference list is denoted by l_i^m and l_j^w respectively. We let L denote the total length of the preference lists in I. Also, for any person $z \in \mathcal{M} \cup \mathcal{W}$, we let $PL(z)$ denote the set of persons on z's original preference list in I, and we let $GS(z)$ denote the set of persons on z's GS-list in I. For each man $m_i \in \mathcal{M}$ and woman $w_j \in PL(m_i)$, we denote the position of w_j on m_i's original preference list (regardless of any deletions that may be carried out by the MEGS/WEGS algorithms) by $rank(m_i, w_j)$, with $rank(w_j, m_i)$ being similarly defined. If $w_j \in \mathcal{W} \backslash PL(m_i)$, then $rank(m_i, w_j)$ and $rank(w_j, m_i)$ are undefined.

We define a CSP encoding J for an instance I of SMI by introducing $2n$ variables to represent the men and women in the original instance I. For each man $m_i \in \mathcal{M}$, we introduce a variable x_i in J whose domain, denoted by $dom(x_i)$, is initially defined as $dom(x_i) = \{1, 2, \ldots, l_i^m\} \cup \{n+1\}$. Similarly, for each woman $w_j \in \mathcal{W}$, we introduce a variable y_j in J whose domain, denoted by $dom(y_j)$, is initially defined as $dom(y_j) = \{1, 2, \ldots, l_j^w\} \cup \{n+1\}$.

An intuitive meaning of the variables is now given. Informally, if $x_i = p$ ($1 \leq p \leq l_i^m$), then m_i marries the woman w_j such that $rank(m_i, w_j) = p$, and similarly for the case that $y_j = q$ ($1 \leq q \leq l_j^w$). More formally, if $\min dom(x_i) \geq p$ ($1 \leq p \leq l_i^m$), then the pair (m_i, w_l) has been deleted as part of the MEGS algorithm applied to I, for all w_l such that $rank(m_i, w_l) < p$. Hence if w_j is the woman such that $rank(m_i, w_j) = p$, then either m_i proposes to w_j during the execution of the MEGS algorithm or the pair (m_i, w_j) will be deleted before the proposal occurs. Similarly if $\min dom(y_j) \geq q$ ($1 \leq q \leq l_j^w$), then the pair (m_k, w_j) has been deleted as part of the WEGS algorithm applied to I, for all m_k such that $rank(m_k, w_j) < q$. Hence if m_i is the man such that $rank(w_j, m_i) = q$, then either w_j proposes to m_i during the execution of the WEGS algorithm or the pair (m_i, w_j) will be deleted before the proposal occurs. If $x_i = n+1$ (respectively $y_j = n+1$) then m_i (respectively w_j) is unmatched upon termination of each of the MEGS or WEGS algorithms applied to I.

The constraints used for the $(n+1)$-valued encoding are shown in Figure 3. In the context of Constraints 1 and 4, j is the integer such that $rank(m_i, w_j) = p$; also $q = rank(w_j, m_i)$. In the context of Constraints 2 and 3, i is the integer such that $rank(w_j, m_i) = q$; also $p = rank(m_i, w_j)$.

An interpretation of Constraints 1 and 3 is now given (a similar interpretation can be attached to Constraints 2 and 4 with the roles of the men and women reversed). First consider Constraint 1, a stability constraint. This ensures that if a man m_i obtains a partner no better than his p^{th}-choice woman w_j, then w_j obtains a partner no worse than her q^{th}-choice man m_i. Now consider Constraint 3, a consistency constraint. This ensures that if man m_i is removed from w_j's list, then w_j is removed from m_i's list.

2.2 Arc consistency in the $(n+1)$-valued encoding

We now show that, given the above CSP encoding J of an SMI instance I, the domains of the variables in J following AC propagation correspond to the GS-lists of I. That is, we prove that, after AC is established, for any i, j ($1 \leq i, j \leq n$), $w_j \in GS(m_i)$ if and only if $p \in dom(x_i)$, and similarly $m_i \in GS(w_j)$ if and only if $q \in dom(y_j)$, where $rank(m_i, w_j) = p$ and $rank(w_j, m_i) = q$.

1.	$x_i \geq p \Rightarrow y_j \leq q$	$(1 \leq i \leq n, 1 \leq p \leq l_i^m)$
2.	$y_j \geq q \Rightarrow x_i \leq p$	$(1 \leq j \leq n, 1 \leq q \leq l_j^w)$
3.	$y_j \neq q \Rightarrow x_i \neq p$	$(1 \leq j \leq n, 1 \leq q \leq l_j^w)$
4.	$x_i \neq p \Rightarrow y_j \neq q$	$(1 \leq i \leq n, 1 \leq p \leq l_i^m)$

Figure 3: The constraints for the $(n+1)$-valued encoding of an instance SMI.

The proof is presented using two lemmas. The first lemma shows that the arc consistent domains are equivalent to subsets of the GS-lists. This is done by proving that the deletions made by the MEGS and WEGS algorithms applied to I are correspondingly made during AC propagation. The second lemma shows that the GS-lists correspond to a subset of the domains remaining after AC propagation. This is done by proving that the GS-lists for I give rise to arc consistent domains for the variables in J.

Lemma 2. *For a given i ($1 \leq i \leq n$), let p be an integer ($1 \leq p \leq l_i^m$) such that $p \in dom(x_i)$ after AC propagation. Then the woman w_j such that $rank(m_i, w_j) = p$ belongs to the GS-list of m_i. A similar correspondence holds for the women.*

Proof. The GS-lists are constructed as a result of the deletions made by the MEGS and WEGS algorithms applied to I. We show that the corresponding deletions are made to the relevant variables' domains during AC propagation. In the following proof, only deletions made by the MEGS algorithm are considered; a similar argument can be used to prove the result for an execution of the WEGS algorithm.

We prove the following fact by induction on the number of proposals z during an execution E of the MEGS algorithm. If proposal z consists of man m_i proposing to woman w_j, with $rank(m_i, w_j) = p$ and $rank(w_j, m_i) = q$, then $x_i \geq p$, $y_j \leq q$ and for each man m_k such that $rank(w_j, m_k) = s$ ($q < s \leq l_j^w$), $x_k \neq r$, where $rank(m_k, w_j) = r$.

First consider the base case where $z = 1$. Then $p = 1$. Since $x_i \geq 1$, propagation of Constraint 1 yields $y_j \leq q$. Then for each s ($q < s \leq l_j^w$), propagation of Constraint 3 gives $x_k \neq r$ where $rank(w_j, m_k) = s$ and $rank(m_k, w_j) = r$.

Now suppose that $z = c > 1$ and that the result holds for $z < c$. We consider the cases where $p = 1$ and $p > 1$.
Case (i). For $p = 1$ the proof is similar to that of the base case.
Case (ii). Now suppose that $p > 1$. Let w_l be any woman such that $rank(m_i, w_l) = r < p$. Then w_l has been deleted from m_i's list during the MEGS algorithm. Now suppose $rank(w_l, m_i) = s_1$. Then m_i was deleted from w_l's preference list because she received a proposal from a man m_k whom she prefers to m_i, where $rank(w_l, m_k) = s_2 < s_1$. Since m_k proposed to w_l before the c^{th} proposal, we have by the induction hypothesis that $y_l \leq s_2$, so that $y_l \neq s_1$ and $x_i \neq r$. But w_l was arbitrary and hence $x_i \neq r$ for $1 \leq r \leq p-1$, so that $x_i \geq p$. The rest of the proof is similar to that of the base case. □

Lemma 3. *For each i ($1 \leq i \leq n$), define a domain of values $dom(x_i)$ for the variable x_i as follows: if $GS(m_i) = \emptyset$, then $dom(x_i) = \{n+1\}$; otherwise $dom(x_i) = \{rank(m_i, w_j) : w_j \in GS(m_i)\}$. The domain of each y_j ($1 \leq j \leq n$) is defined analogously. Then the domains so defined are arc consistent in J.*

Proof. To show that the variables' domains are arc consistent we consider each constraint in turn.

First consider Constraint 1 and suppose that $x_i \geq p$. Then during the execution of the MEGS algorithm applied to I, either (i) m_i proposed to w_j, or (ii) the pair (m_i, w_j) was deleted, where $rank(m_i, w_j) = p$ and $rank(w_j, m_i) = q$. We consider the two cases below:
Case (i) If m_i proposed to w_j during the execution of the MEGS algorithm, then all men ranked below m_i on w_j's list are deleted, i.e. $y_j \leq q$ as required.
Case (ii) If (m_i, w_j) was deleted during the execution of the MEGS algorithm then w_j must have received a proposal from a man m_k whom she prefers to m_i, where $rank(w_j, m_k) = s$ ($s < q$). Therefore the MEGS algorithm deletes all those men m_z from w_j's list such that $rank(w_j, m_z) > s$, i.e. $y_j \leq s < q$ as required.

Next consider Constraint 3. Suppose that $y_j \neq q$, so that during an execution of either the MEGS or WEGS algorithms, m_i is deleted from w_j's list, where $rank(w_j, m_i) = q$. To ensure that the preference lists are consistent, the same algorithm deletes w_j from m_i's list, i.e. $x_i \neq p$, where $rank(m_i, w_j) = p$, as required.

Verifying Constraints 2 and 4 is similar to the above with the roles of the men and women reversed and the MEGS algorithm exchanged for the WEGS algorithm. □

The two lemmas above, together with the fact that AC algorithms find the unique maximal set of arc consistent domains, lead to the following theorem.

Theorem 4. *Let I be an instance of SMI, and let J be a CSP instance obtained by the (n+1)-valued encoding. Then the domains remaining after AC propagation in J correspond to the GS-lists of I in the following sense: for any i, j ($1 \leq i, j \leq n$), $w_j \in GS(m_i)$ if and only if $p \in dom(x_i)$, and similarly $m_i \in GS(w_j)$ if and only if $q \in dom(y_j)$, where $rank(m_i, w_j) = p$ and $rank(w_j, m_i) = q$.*

The constraints shown in Figure 3 may be revised in $O(1)$ time during propagation, assuming that upper and lower bounds for the variables' domains are maintained. Hence the time complexity for establishing AC is $O(ed)$, where e is the number of constraints and d is the domain size [van Hentenryck *et al.*, 1992]. For this encoding we have $e = O(n^2)$ and $d = O(n)$, therefore AC may be established in $O(n^3)$ time; also the space complexity is $O(L)$. These complexities represent an improvement on the 'conflict matrices' encoding in [Gent *et al.*, 2001], whose time and space complexities are $O(n^4)$ and $O(L^2)$ respectively. Moreover we claim that the model that we present in this section is a very natural and intuitive encoding for SMI.

Theorems 4 and 1(iii) show that we can find a solution to the CSP giving the man-optimal stable matching M_0 without search: for each man $m_i \in \mathcal{M}$, we let $p = \min dom(x_i)$. If $p = n+1$ then m_i is unmatched in M_0, otherwise the partner of m_i is the woman $w_j \in \mathcal{W}$ such that $rank(m_i, w_j) = p$. Considering the y_j variables in a similar fashion gives the woman-optimal stable matching M_z.

In fact we may go further and show that the CSP encoding yields all stable matchings in I without having to backtrack due to failure. That is, we may enumerate all solutions of I in a failure-free manner using AC propagation in J combined with a value-ordering heuristic. The following theorem, proved in [Manlove and O'Malley, 2005], describes the enumeration procedure.

Theorem 5. *Let I be an instance of SMI and let J be a CSP instance obtained using the $(n+1)$-valued encoding. Then the following search process enumerates all solutions in I without repetition and without ever failing due to an inconsistency:*

- *AC is established as a preprocessing step, and after each branching decision, including the decision to remove a value from a domain;*
- *if all domains are arc consistent and some variable x_i has two or more values in its domain, then the search proceeds by setting x_i to the minimum value p in its domain. On backtracking, the value p is removed from the domain of x_i;*
- *when a solution is found, it is reported and backtracking is forced.*

3 4-valued encoding

3.1 Overview of the encoding

In this section we present a CSP encoding of SMI that is more complex but more efficient than the $(n+1)$-valued encoding given in Section 2.1. We assume the notation as defined for an instance of SMI in the first paragraph of Section 2.1.

We construct a CSP encoding J for an SMI instance I by introducing L variables, each of which represents a preference list entry. For each man m_i $(1 \leq i \leq n)$ we introduce l_i^m variables $x_{i,p}$ $(1 \leq p \leq l_i^m)$, corresponding to the members of $PL(m_i)$. Similarly for each woman w_j $(1 \leq j \leq n)$ we introduce l_j^w variables $y_{j,q}$ $(1 \leq q \leq l_j^w)$. As before the domain of a variable z is denoted by $dom(z)$; initially each variable is given the domain $\{0,1,2,3\}$.

An intuitive meaning of the variables' values is given in Figure 4. The table indicates that deletions carried out by the MEGS and WEGS algorithms applied to I are reflected by the removal of elements from the relevant variables' domains. In particular, removal of the value 2 (respectively 3) from a variable's domain corresponds to a preference list entry being deleted by the MEGS (respectively WEGS) algorithm applied to I. Note that potentially a given preference list entry could be deleted by both algorithms. Also, if the value 0 is removed from $dom(x_{i,p})$ $(1 \leq i \leq n$, $1 \leq p \leq l_i^m)$, then either m_i proposes to w_j during the MEGS algorithm (where $rank(m_i, w_j) = p$) or the entry is deleted prior to the proposal occurring. Similarly if the value 0 is removed from $dom(y_{j,q})$ $(1 \leq j \leq n$, $1 \leq q \leq l_j^w)$, then either w_j proposes to m_i during the WEGS algorithm (where $rank(w_j, m_i) = q$) or the entry is deleted prior to the proposal occurring.

The constraints for this encoding are listed in Figure 5. In the context of Constraints 4 and 10, j is the integer such that $rank(m_i, w_j) = p$; also $q = rank(w_j, m_i)$. In the context of Constraints 5 and 9, i is the integer such that $rank(w_j, m_i) = q$; also $p = rank(m_i, w_j)$. Further, we remark that Constraints 4 and 9 are present only if $q+1 \leq l_j^w$ and $p+1 \leq l_i^m$ respectively.

An interpretation of each constraint is now given. Firstly consider Constraint 1. This constraint is used to start the proposal sequence and can be interpreted as each man initially proposing to the first woman on his list during the MEGS algorithm. Constraint 2 states that if (m_i, w_l) has been deleted by the MEGS algorithm for all w_l such that $rank(m_i, w_l) < p$, and (m_i, w_j) has also been deleted, where $rank(m_i, w_j) = p$, then (m_i, w_l) has been deleted by the by MEGS algorithm for all w_l such that $rank(m_i, w_l) \leq p$. Hence, if $p+1 \leq l_i^m$, m_i will subsequently propose to the woman w_l such that $rank(m_i, w_l) = p+1$ during the MEGS algorithm, or the pair (m_i, w_l) will be deleted before the proposal occurs. Constraint 3 states that if a woman's q^{th}-choice partner is deleted during an iteration of the MEGS algorithm, then her $(q+1)^{th}$-choice partner should also be deleted. Constraint 4 shows a stability constraint: this ensures that if man m_i obtains a partner no better than w_j, then w_j obtains a partner no worse than m_i. Lastly Constraint 5 is a consistency constraint: this ensures that if m_i is removed from w_j's list during the MEGS algorithm then w_j is also removed from m_i's list. Constraints 6-10 have a similar meaning with the roles of the men and women reversed, and with MEGS replaced by WEGS.

3.2 Arc consistency in the 4-valued encoding

We now prove that, given the above CSP encoding J of an SMI instance I, the domains of the variables in J following AC propagation correspond to the GS-lists of I. That is, we show that, after AC is established, for any i, j $(1 \leq i, j \leq n)$, $w_j \in GS(m_i)$ if and only if $\{2,3\} \subseteq dom(x_{i,p})$, and similarly $m_i \in GS(w_j)$ if and only if $\{2,3\} \subseteq dom(y_{j,q})$, where $rank(m_i, w_j) = p$ and $rank(w_j, m_i) = q$.

In order to establish this correspondence, we define the *GS-domains* for the variables in J as follows. Initially let each variable in J have domain $\{0,1,2,3\}$. Run the MEGS algorithm on instance I. Then use rules (i), (ii) and (v) in Figure 4 to remove 0's and 2's from the appropriate domains, obtaining CSP instance J' from J. Next run the WEGS algorithm on the original instance I. Now use rules (iii), (iv) and (vi) in Figure 4 to remove 0's and 3's from the appropriate domains in J', obtaining CSP instance J''. The domains of the variables in J'' are referred to as the *GS-domains*.

As in Section 2.2, two lemmas are used to prove that enforcing AC gives the GS-lists. The first lemma shows that the domains remaining following AC propagation are equivalent to subsets of the GS-lists. This is done by proving that if a deletion is made as part of either the MEGS or WEGS algorithms, then a corresponding deletion is made during AC propagation. The second lemma shows that the GS-lists correspond to a subset of the domains remaining after AC is enforced. This is done by proving that the GS-domains for J are arc consistent.

Lemma 6. *For a given i $(1 \leq i \leq n)$, let p be an integer such that $\{2,3\} \subseteq dom(x_{i,p})$ after AC propagation. Then the woman w_j such that $rank(m_i, w_j) = p$ belongs to the GS-list of m_i. A similar correspondence holds for the women.*

Proof. The GS-lists are obtained through deletions made by the MEGS and WEGS algorithms. We prove that the corresponding deletions are made to the relevant variables' domains during AC propagation. In particular, suppose that

(i)	$0 \notin dom(x_{i,p})$	$\Leftrightarrow$	$p = 1$ or $2 \notin dom(x_{i,r})$ for all r $(1 \leq r < p)$ (i.e. man m_i's r^{th}-choice woman is removed from his list as part of the MEGS algorithm applied to I, for all r $(1 \leq r < p)$);
(ii)	$2 \notin dom(x_{i,p})$	$\Leftrightarrow$	man m_i's p^{th}-choice woman is removed from his list as part of the MEGS algorithm applied to I;
(iii)	$3 \notin dom(x_{i,p})$	$\Leftrightarrow$	man m_i's p^{th}-choice woman is removed from his list as part of the WEGS algorithm applied to I;
(iv)	$0 \notin dom(y_{j,q})$	$\Leftrightarrow$	$q = 1$ or $3 \notin dom(y_{i,s})$ for all s $(1 \leq s < q)$ (i.e. woman w_j's s^{th}-choice man is removed from her list as part of the WEGS algorithm applied to I, for all s $(1 \leq s < q)$);
(v)	$2 \notin dom(y_{j,q})$	$\Leftrightarrow$	woman w_j's q^{th}-choice man is removed from her list as part of the MEGS algorithm applied to I;
(vi)	$3 \notin dom(y_{j,q})$	$\Leftrightarrow$	woman w_j's q^{th}-choice man is removed from her list as part of the WEGS algorithm applied to I.

Figure 4: Intuitive variable meanings for the 4-valued SMI encoding.

$m_i \in \mathcal{M}$ and $w_j \in PL(m_i)$. Let $p = rank(m_i, w_j)$ and $q = rank(w_j, m_i)$. Then we prove:

– (m_i, w_j) deleted during MEGS algorithm $\Leftrightarrow x_{i,p} \neq 2$ and $y_{j,q} \neq 2$.

– (m_i, w_j) deleted during WEGS algorithm $\Leftrightarrow x_{i,p} \neq 3$ and $y_{j,q} \neq 3$.

In this proof, only deletions made by the MEGS algorithm are considered; a similar argument can be used for deletions made by the WEGS algorithm.

It suffices to prove the following by induction on the number of proposals z during an execution E of the MEGS algorithm. If proposal z consists of man m_i proposing to woman w_j, with $rank(m_i, w_j) = p$ and $rank(w_j, m_i) = q$, then $x_{i,p} > 0$, $y_{j,s} \neq 2$ $(q < s \leq l_j^w)$, and for each man m_k such that $rank(w_j, m_k) = s$ $(q < s \leq l_j^w)$, $x_{k,r} \neq 2$, where $rank(m_k, w_j) = r$.

First consider the base case where $z = 1$. Then $p = 1$. By Constraint 1, $x_{i,1} > 0$, and by Constraint 4 we have $y_{j,q+1} \neq 2$. Hence by Constraint 3, it follows that $y_{j,s} \neq 2$ for each s $(q < s \leq l_j^w)$. Also for each such s, propagation of Constraint 5 ensures that $x_{k,r} \neq 2$, where $rank(w_j, m_k) = s$ and $rank(m_k, w_j) = r$.

Now suppose that $z = c > 1$ and that the result holds for $z < c$. We consider the cases where $p = 1$ and $p > 1$.

Case (i) For $p = 1$ the proof is similar to that of the base case.

Case (ii) Now assume that $p > 1$. Let w_l be any woman such that $rank(m_i, w_l) = r < p$. Then w_l has been deleted from m_i's list during the MEGS algorithm. Now suppose that $rank(w_l, m_i) = s_1$. Then m_i was deleted from w_l's list because she received a proposal from a man m_k whom she prefers to m_i, where $rank(w_l, m_k) = s_2 < s_1$. Since m_k proposed to w_l before the c^{th} proposal, by the induction hypothesis it follows that $x_{i,r} \neq 2$. However since w_l was arbitrary, it follows that $x_{i,r} \neq 2$ for $1 \leq r \leq p - 1$. From Constraint 1 we have $x_{i,1} > 0$, and hence the propagation of Constraint 2 ($p - 1$ times) yields $x_{i,p} > 0$. The rest of the proof is similar to that of the base case. □

Lemma 7. *The GS-domains (corresponding to the GS-lists in I) are arc consistent in J.*

Proof. We consider each constraint in turn to show that the GS-domains are arc consistent.

Clearly Constraint 1 is satisfied, as $p = 1$ in rule (i) of Figure 4, i.e. $x_{i,1} > 0$. Now consider Constraint 4 and suppose that $x_{i,p} > 0$. Then during the execution of the MEGS algorithm, either (i) m_i proposed to w_j, or (ii) the pair (m_i, w_j) was deleted, where $rank(m_i, w_j) = p$ and

1.	$x_{i,1} > 0$	$(1 \leq i \leq n)$
2.	$(x_{i,p} \neq 2 \wedge x_{i,p} > 0) \Rightarrow x_{i,p+1} > 0$	$(1 \leq i \leq n, 1 \leq p \leq l_i^m - 1)$
3.	$y_{j,q} \neq 2 \Rightarrow y_{j,q+1} \neq 2$	$(1 \leq j \leq n, 1 \leq q \leq l_j^w - 1)$
4.	$x_{i,p} > 0 \Rightarrow y_{j,q+1} \neq 2$	$(1 \leq i \leq n, 1 \leq p \leq l_i^m)$
5.	$y_{j,q} \neq 2 \Rightarrow x_{i,p} \neq 2$	$(1 \leq j \leq n, 1 \leq q \leq l_j^w)$
6.	$y_{j,1} > 0$	$(1 \leq j \leq n)$
7.	$(y_{j,q} \neq 3 \wedge y_{j,q} > 0) \Rightarrow y_{j,q+1} > 0$	$(1 \leq j \leq n, 1 \leq q \leq l_j^w - 1)$
8.	$x_{i,p} \neq 3 \Rightarrow x_{i,p+1} \neq 3$	$(1 \leq i \leq n, 1 \leq p \leq l_i^m - 1)$
9.	$y_{j,q} > 0 \Rightarrow x_{i,p+1} \neq 3$	$(1 \leq j \leq n, 1 \leq q \leq l_j^w)$
10.	$x_{i,p} \neq 3 \Rightarrow y_{j,q} \neq 3$	$(1 \leq i \leq n, 1 \leq p \leq l_i^m)$

Figure 5: The constraints for the 4-valued encoding of an instance SMI.

$rank(w_j, m_i) = q$. Assuming $q + 1 \leq l_j^w$, we consider the two cases separately.
Case (i) If m_i proposed to w_j during the execution of the MEGS algorithm, then w_j deletes all those men ranked below m_i on her preference list, so that in particular, $y_{j,q+1} \neq 2$.
Case (ii) If the pair (m_i, w_j) was deleted during the execution of the MEGS algorithm, then w_j must have received a proposal from a man m_k whom she prefers to m_i. Consequently, all men ranked below m_k on w_j's list are deleted by the MEGS algorithm, so that in particular, $y_{j,q+1} \neq 2$.

Now suppose that $y_{j,q} \neq 2$. Then by construction of the GS-domains, the MEGS algorithm deleted the man m_i such that $rank(w_j, m_i) = q$. So in addition, 2 is removed from the domain of $x_{i,p}$, where $rank(m_i, w_j) = p$, satisfying Constraint 5. Also, as in Case (ii) above, $y_{j,q+1} \neq 2$, satisfying Constraint 3.

Now consider Constraint 2 and suppose that $x_{i,p} \neq 2$ and $x_{i,p} > 0$. Then w_j has been removed from the list of m_i, where $rank(m_i, w_j) = p$. Also $x_{i,p} > 0$ implies that either (i) $p = 1$, or (ii) $x_{i,r} \neq 2$ $(1 \leq r < p)$. We consider the two cases separately.
Case (i) If $p = 1$, we have $x_{i,1} \neq 2$, and hence $x_{i,2} > 0$ by construction of the GS-domains.
Case (ii) As $x_{i,p} > 0$, it follows that $x_{i,r} \neq 2$ $(1 \leq r < p)$. Also $x_{i,p} \neq 2$. Hence $x_{i,r} \neq 2$ $(1 \leq r \leq p)$, so that $x_{i,p+1} > 0$ by construction of the GS-domains.

A similar argument can be used to verify that Constraints 6-10 are satisfied. Here the roles of the men and women are reversed and MEGS is replaced by WEGS. □

The two lemmas above, together with the fact that AC algorithms find the unique maximal set of arc consistent domains, lead to the following theorem.

Theorem 8. *Let I be an instance of SMI, and let J be a CSP instance obtained by the 4-valued encoding. Then the domains remaining after AC propagation in J correspond to the GS-lists of I in the following sense: for any i, j ($1 \leq i, j \leq n$), $w_j \in GS(m_i)$ if and only if $\{2, 3\} \subseteq dom(x_{i,p})$, and similarly $m_i \in GS(w_j)$ if and only if $\{2, 3\} \subseteq dom(y_{j,q})$, where $rank(m_i, w_j) = p$ and $rank(w_j, m_i) = q$.*

In general AC may be established in $O(ed^r)$ time, where e is the number of constraints, d the domain size, and r the arity of each constraint [Bessière and Régin, 1997]. In the context of the 4-valued encoding, it follows that $e = O(L)$, $d = 4$ and $r = 2$, and hence AC may be enforced in time $O(L) = O(n^2)$. The time complexity of $O(L)$ is linear in the size of I and gives an improvement over the encoding presented in Section 2.1. Moreover $O(L)$ is also the time complexity of the EGS algorithm, which is known to be optimal [Ng and Hirschberg, 1990]. The space complexity of the 4-valued encoding is also $O(L)$.

Theorems 8 and 1(iii) show that we can find a solution to the CSP giving the man-optimal stable matching M_0 without search: for each man $m_i \in \mathcal{M}$, if $\{2, 3\} \not\subseteq dom(x_{i,r})$ for each r $(1 \leq r \leq l_i^m)$ then m_i is unmatched in M_0, otherwise we let p be the unique integer such that $dom(x_{i,p}) = \{1, 2, 3\}$ and define the partner of m_i to be the woman $w_j \in \mathcal{W}$ such that $rank(m_i, w_j) = p$. Considering the y_j variables in a similar way gives the woman-optimal stable matching M_z.

As in Section 2, we may go further and show that the CSP encoding yields all stable matchings in I without having to backtrack due to failure. As before we enumerate all solutions of I in a failure-free manner using AC propagation in J combined with a value-ordering heuristic, however in this case, maintenance of AC is much less expensive. The following theorem, proved in [Manlove and O'Malley, 2005], describes the enumeration strategy in this context.

Theorem 9. *Let I be an instance of SMI and let J be a CSP instance obtained from I using the 4-valued encoding. Then the following search process enumerates all solutions in I without repetition and without ever failing due to an inconsistency:*

- *AC is established as a preprocessing step, and after each branching decision, including the decision to remove a value from a domain;*
- *if all domains are arc consistent and some variable $x_{i,r}$ has $\{0, 1, 2, 3\}$ in its domain, then we let p be the unique integer such that $dom(x_{i,p}) = \{1, 2, 3\}$ and we choose p' to be the minimum integer $(p < p')$ such that $dom(x_{i,p'}) = \{0, 1, 2, 3\}$;*
- *the search proceeds by removing the value 3 from the domain of $x_{i,p'}$. On backtracking, the value 2 is removed from the domain of $y_{j,q}$, where $rank(m_i, w_j) = p$ and $rank(w_j, m_i) = q$;*
- *when a solution is found, it is reported and backtracking is forced.*

4 Concluding remarks

In this paper we have described two models for the Stable Marriage problem and its variant SMI as a CSP. Our first encoding is very natural and may be used to derive the GS-lists following AC propagation, although the time complexity for establishing AC is worse than that of the EGS algorithm. Our second encoding, whilst more complex, again yields the GS-lists, but this time the time complexity for AC propagation is optimal. Using both encodings we are able to find all stable matchings for a given instance of SMI using a failure-free enumeration without search.

A natural extension of this work is to the case where there is indifference in the preference lists. It has already been demonstrated [Gent and Prosser, 2002a; 2002b] that the earlier encodings of [Gent *et al.*, 2001] can be extended to the case where preference lists in a given SMI instance may include ties, suggesting that the same should be possible with the models that we present here. Another direction is to consider the Hospitals / Residents problem (HR) (a many-one generalisation of SMI). The $(n + 1)$-valued encoding from this paper, and the specialised constraints from [Unsworth and Prosser, 2005a; 2005b], have already been generalised to the HR case (see [Manlove *et al.*, 2005] for further details).

Finally, it remains to conduct an empirical investigation of the encodings presented in this paper, based on randomly-generated and real-world data. Such investigations have already been carried out for other encodings for SM and its variants [Gent and Prosser, 2002a; 2002b; Unsworth and Prosser, 2005a; 2005b].

Acknowledgements

We would like to thank Rob Irving and Patrick Prosser for helpful comments on earlier drafts of this paper. We would also like to thank Patrick Prosser for suggesting that previous (man-oriented and woman-oriented) versions of the models that we present here could be amalgamated.

References

[Aldershof and Carducci, 1999] B. Aldershof and O.M. Carducci. Refined inequalities for stable marriage. *Constraints*, 4:281–292, 1999.

[Bessière and Régin, 1997] C. Bessière and J-C. Régin. Arc consistency for general constraint networks: Preliminary results. In *Proceedings of IJCAI '97*, volume 1, pages 398–404. Morgan Kaufmann, 1997.

[Dye, 2001] J. Dye. A constraint logic programming approach to the stable marriage problem and its application to student-project allocation. BSc Honours project report, University of York, Department of Computer Science, 2001.

[Gale and Shapley, 1962] D. Gale and L.S. Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, 69:9–15, 1962.

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

[Gent and Prosser, 2002a] I.P. Gent and P. Prosser. An empirical study of the stable marriage problem with ties and incomplete lists. In *Proceedings of ECAI '02*, pages 141–145. IOS Press, 2002.

[Gent and Prosser, 2002b] I.P. Gent and P. Prosser. SAT encodings of the stable marriage problem with ties and incomplete lists. In *Proceedings of SAT '02*, 2002. `http://gauss.ececs.uc.edu/Conferences/SAT2002/Abstracts/gent.ps`.

[Gent *et al.*, 2001] I.P. Gent, R.W. Irving, D.F. Manlove, P. Prosser, and B.M. Smith. A Constraint Programming Approach to the Stable Marriage Problem. In *Proceedings of CP '01*, volume 2239 of *Lecture Notes in Computer Science*, pages 225–239. Springer-Verlag, 2001.

[Green and Cohen, 2003] M.J. Green and D.A. Cohen. Tractability by approximating constraint languages. In *Proceedings of CP '03*, volume 2833 of *Lecture Notes in Computer Science*, pages 392–406. Springer-Verlag, 2003.

[Gusfield and Irving, 1989] D. Gusfield and R.W. Irving. *The Stable Marriage Problem: Structure and Algorithms*. MIT Press, 1989.

[Kato, 1993] A. Kato. Complexity of the sex-equal stable marriage problem. *Japan Journal of Industrial and Applied Mathematics*, 10:1–19, 1993.

[Knuth, 1976] D.E. Knuth. *Mariages Stables*. Les Presses de L'Université de Montréal, 1976. English translation in *Stable Marriage and its Relation to Other Combinatorial Problems*, volume 10 of CRM Proceedings and Lecture Notes, American Mathematical Society, 1997.

[Lustig and Puget, 2001] I.J. Lustig and J. Puget. Program does not equal program: constraint programming and its relationship to mathematical programming. *Interfaces*, 31:29–53, 2001.

[Manlove and O'Malley, 2005] D.F. Manlove and G. O'Malley. Modelling and solving the stable marriage problem using constraint programming. Technical Report TR-2005-192, University of Glasgow, Department of Computing Science, 2005.

[Manlove *et al.*, 2002] D.F. Manlove, R.W. Irving, K. Iwama, S. Miyazaki, and Y. Morita. Hard variants of stable marriage. *Theoretical Computer Science*, 276(1-2):261–279, 2002.

[Manlove *et al.*, 2005] D.F. Manlove, G. O'Malley, P. Prosser, and C. Unsworth. A Constraint Programming Approach to the Hospitals / Residents Problem. Technical Report TR-2005-196, University of Glasgow, Department of Computing Science, 2005.

[Ng and Hirschberg, 1990] C. Ng and D.S. Hirschberg. Lower bounds for the stable marriage problem and its variants. *SIAM Journal on Computing*, 19:71–77, 1990.

[Ng and Hirschberg, 1991] C. Ng and D.S. Hirschberg. Three-dimensional stable matching problems. *SIAM Journal on Discrete Mathematics*, 4:245–252, 1991.

[Ronn, 1990] E. Ronn. NP-complete stable matching problems. *Journal of Algorithms*, 11:285–304, 1990.

[Roth and Sotomayor, 1990] A.E. Roth and M.A.O. Sotomayor. *Two-sided matching: a study in game-theoretic modeling and analysis*, volume 18 of *Econometric Society Monographs*. Cambridge University Press, 1990.

[Roth, 1984] A.E. Roth. The evolution of the labor market for medical interns and residents: a case study in game theory. *Journal of Political Economy*, 92(6):991–1016, 1984.

[Thorn, 2003] M. Thorn. A constraint programming approach to the student-project allocation problem. BSc Honours project report, University of York, Department of Computer Science, 2003.

[Unsworth and Prosser, 2005a] C. Unsworth and P. Prosser. An n-ary constraint for the stable marriage problem. To appear in *Proceedings of the Fifth Workshop on Modelling and Solving Problems with Constraints*, 2005.

[Unsworth and Prosser, 2005b] C. Unsworth and P. Prosser. A specialised binary constraint for the stable marriage problem. To appear in *Proceedings of SARA '05*, *Lecture Notes in Computer Science*. Springer-Verlag, 2005.

[van Hentenryck *et al.*, 1992] P. van Hentenryck, Y. Deville, and C-M. Teng. A generic arc-consistency algorithm and its specializations. *Artificial Intelligence*, 57:291–321, 1992.

[Vate, 1989] J.E. Vande Vate. Linear programming brings marital bliss. *Operations Research Letters*, 8(3):147–153, 1989.