Foundations
of Computing
Series

The Stable Marriage Problem

Structure and Algorithms

Dan Gusfield
and Robert W. Irving

The MIT Press

some initial insight into its algebraic structure. Simple variants of stable marriage are considered in Section 1.4, and in Section 1.5, lower bounds are established for a number of algorithmic problems associated with stable marriage, showing in particular that the Gale-Shapley algorithm is asymptotically optimal. Section 1.6 covers in some detail the college admissions or hospitals/residents problem (which we shall refer to henceforth, for consistency, as the hospitals/residents problem). Finally, Section 1.7 considers briefly some of the issues involved if one or more of the participants attempts to influence the outcome of the Gale-Shapley algorithm, or one of its variants, by falsifying preferences — issues of deceit, strategy, and coalition that have received a good deal of attention in the literature in recent years and that are of some practical importance in the context of the NRMP algorithm.

1.1.2 Stable Marriage: Basic Terminology and Notation

An instance of *size* n of the stable marriage problem involves two disjoint sets of size n, the men and the women. Associated with each person is a *strictly* ordered *preference list* containing *all* the members of the opposite sex. Person p prefers q to r, where q and r are of the opposite sex to p, if and only if q precedes r on p's preference list.

For such an instance, a matching M is a one-one correspondence between the men and the women. If man m and woman w are matched in M, then m and w are called *partners* in M, and we write $m = p_M(w)$, $w = p_M(m)$; $p_M(m)$ is the M-partner of m, and $p_M(w)$ the M-partner of w.

A man m and a woman w are said to *block* a matching M, or to be a *blocking pair* for M, if m and w are not partners in M, but m prefers w to $p_M(m)$ and w prefers m to $p_M(w)$. A matching for which there is at least one blocking pair is called *unstable*, and is otherwise *stable*.

The basic stable marriage problem involves the determination, for a given instance, of a stable matching (which, as already mentioned in Section 1.1.1, always exists). Of course, over and above this basic question, there are many other interesting questions that can be asked about stable matchings.

Example Consider the stable marriage instance of size 4 specified by the preference lists in Figure 1.1. Here, as throughout, it is assumed that the men and women are separately and arbitrarily labeled $1, \ldots, n$, and the men's and women's preference lists are arranged horizontally in two separate arrays.

The matching $\{(1,4),(2,3),(3,2),(4,1)\}$ is stable. Here, and elsewhere in the text, a matching is specified as a set of ordered man-woman pairs. Stability may be verified by considering each man in turn as a potential member of a blocking pair. Man 1 could form a blocking pair only with woman 2, but she prefers her partner,

1	2	4	1	3	1	2	1	4	3
2	3	1	4	2	2	4	3	1	2
3	2	3	1	4	3	1	4	3	2
4	4	1	3	2	4	2	1	4	3
	Men's Preferences					Women's Preferences			

Figure 1.1: A first stable marriage instance of size 4

man 3, to man 1. Each of men 2 and 3 is matched with his favorite woman, so neither can be in a blocking pair. Finally, man 4 could form a blocking pair only with woman 4, but she would rather stick with her partner, man 1.

A second example of a stable matching, indeed the only other stable matching in this case, is $\{(1,4),(2,1),(3,2),(4,3)\}$, as may be verified in a similar way. On the other hand, the matching $\{(1,1),(2,3),(3,2),(4,4)\}$, for example, is unstable because of the blocking pair $(1,4)$; man 1 prefers woman 4 to his partner, woman 1, and woman 4 prefers man 1 to her partner, man 4. Some other unstable matchings may have many more blocking pairs: for example, the matching $\{(1,1),(2,2),(3,4),(4,3)\}$ has six, which the reader may care to find.

Stability Checking

It may not be immediately obvious from the problem statement that a stable matching always exists, or how stable matchings may be found, but it should be obvious, as illustrated in the example, how a given matching may be checked for stability. It suffices to consider each member of one sex, say the men, as a potential member of a blocking pair. For each man, only the women that he prefers to his partner need be checked. More precisely, Figure 1.2 contains a stability checking,algorithm, and since there are n men in an instance of size n, and for each, at most $n-1$ women need be examined, it should be clear that with appropriate data structures, the algorithm has $O(n^2)$ worst-case complexity.

Note that here, as elsewhere in the book, algorithms are expressed in an informal Pascal-like language that should be self-explanatory.

Finally, we introduce some additional, fairly obvious, terminology. A man m and a woman w constitute a *stable pair* if and only if m and w are partners in some stable matching; in these circumstances, m is a *stable partner* of w, and vice versa. If some man m and woman w are partners in *all* stable matchings, then (m,w) is called a *fixed pair.*

For stylistic reasons, we use a number of phrases as synonyms for "m

```
for m := 1 to n do
    for each w such that m prefers w to p_M(m) do
        if w prefers m to p_M(w) then
        begin
            report matching unstable ;
            halt
        end ;
report matching stable
```

Figure 1.2: Simple stability-checking algorithm

prefers v to w"; these include "v is a *better* and w a *poorer* or *worse* partner for m" and "v is *more favored* and w *less favored* by m".

1.2 The Gale-Shapley Algorithm

1.2.1 The Basic Algorithm

We now develop the fundamental theorem, due to Gale and Shapley, that there always exists at least one stable matching in an instance of the stable marriage problem. To prove this theorem, we describe a version of the original Gale-Shapley algorithm. This simple algorithm always finds a stable matching, which, as mentioned earlier, turns out to be uniquely favorable to the men or to the women, depending on the respective roles of the two sexes in the algorithm. In our description of the algorithm, we will adopt the traditional approach, regarding the men as "suitors" in a "courtship" process, but analogous results may be obtained by reversing the roles of the sexes.

Informally, the algorithm may be expressed in terms of a sequence of "proposals" from men to women. At any point during the algorithm's execution, each person is either *engaged* or *free*; each man may alternate between being engaged and being free, but once a woman is engaged, she is never again free, although the identity of her fiancé may change. A man who is engaged more than once obtains fiancées who are successively less desirable to him, while each successive engagement brings a woman a more favored partner.

When a free woman receives a proposal, she will immediately accept it, becoming engaged to the proposer. When an engaged woman receives a

proposal, she compares the proposer with her current fiancé and rejects the less favored of the two men; that is, if she prefers her fiancé, she rejects the new proposal, but if she prefers the proposer, she breaks her current engagement, setting her ex-fiancé free, and becomes engaged to the current proposer.

Each man proposes to the women on his preference list, in their order of appearance, until he becomes engaged. If ever that engagement is broken (by the woman), then he becomes free again, and he resumes his sequence of proposals, starting with the next woman on his list. The algorithm terminates when everyone is engaged, and we will see that this will happen before any man exhausts his preference list. Furthermore, we will show that, on termination, the engaged couples constitute a stable matching.

The basic Gale-Shapley algorithm in which the men propose — the *man-oriented* version — is summarized in Figure 1.3.

```
assign each person to be free ;
while some man m is free do
begin
    w := first woman on m's list to whom m has not yet proposed ;
    if w is free then
        assign m and w to be engaged {to each other}
    else
        if w prefers m to her fiancé m' then
            assign m and w to be engaged and m' to be free
        else
            w rejects m {and m remains free}
end ;
output the stable matching consisting of the n engaged pairs
```

Figure 1.3: Basic Gale-Shapley algorithm

As expressed in Figure 1.3, the Gale-Shapley algorithm involves an element of nondeterminism, since the order in which the free men propose is not specified. However, it turns out, as we will see, that this nondeterminism is of no consequence: the order in which the free men propose is immaterial to the outcome.

The fundamental nature of the Gale-Shapley algorithm is summarized in the following theorem.

Theorem 1.2.1 *For any given instance of the stable marriage problem, the Gale-Shapley algorithm terminates, and, on termination, the engaged pairs constitute a stable matching.*

Proof First, we show that no man can be rejected by all the women. A woman can reject only when she is engaged, and once she is engaged she never again becomes free. So the rejection of a man by the last woman on his list would imply that all the women were already engaged. But since there are equal numbers of men and women, and no man has two fiancées, all the men would also be engaged, which is a contradiction. Also, each iteration involves one proposal, and no man ever proposes twice to the same woman, so the total number of iterations cannot exceed n^2 (for an instance involving n men and n women). Termination is therefore established.

It is clear that, on termination, the engaged pairs specify a matching, which we denote by M. If man m prefers woman w to $p_M(m)$, then w must have rejected m at some point during the execution of the algorithm. But this rejection implies that w was, or became, engaged to a man she prefers to m, and any subsequent change of her fiancé brings her a still better partner. So w cannot prefer m to $p_M(w)$, and therefore (m, w) cannot block M. It follows that there are no blocking pairs for M, and therefore that M is a stable matching. □

Example Consider the instance of size 4 defined by the preference lists in Figure 1.4.

1	4	1	2	3	1	4	1	3	2
2	2	3	1	4	2	1	3	2	4
3	2	4	3	1	3	1	2	3	4
4	3	1	4	2	4	4	1	3	2
	Men's Preferences					Women's Preferences			

Figure 1.4: A stable marriage instance of size 4

One possible execution of the algorithm results in the following sequence of proposals: man 1 to woman 4 (accepted); man 2 to woman 2 (accepted); man 3 to woman 2 (accepted, and woman 2 now rejects man 2); man 2 to woman 3 (accepted); man 4 to woman 3 (rejected, for woman 3 prefers man 2); man 4 to woman 1 (accepted). Hence the stable matching generated by the man-oriented version of the algorithm is $\{(1,4),(2,3),(3,2),(4,1)\}$.

1.2.2 Man and Woman Optimal Stable Matchings

As already mentioned, all possible executions of the Gale-Shapley algorithm (with the men as proposers) lead to the same stable matching. Furthermore, this stable matching has the remarkable property that every man achieves in it the best partner that he can possibly have in any stable matching. It is perhaps surprising that all the men, who are essentially in competition with each other for the women, can agree on a stable matching that is simultaneously optimal for all of them. This result is stated formally in the next theorem, which also establishes the insignificance of the nondeterminism in the algorithm.

Theorem 1.2.2 *All possible executions of the Gale-Shapley algorithm (with the men as proposers) yield the same stable matching, and in this stable matching, each man has the best partner that he can have in any stable matching.*

Proof Suppose that an arbitrary execution E of the algorithm yields the stable matching M, and that, in contradiction of the theorem, there is a stable matching M' and a man m such that m prefers $w' = p_{M'}(m)$ to $w = p_M(m)$. Then during E, w' must have rejected m. Suppose, without loss of generality, that this was the first occasion, during E, that a woman rejected a stable partner, and suppose that this rejection took place because of the engagement of w' to m' (so that w' prefers m' to m). Then m' can have no stable partner whom he prefers to w' (for no woman had previously rejected a stable partner). So m' prefers w' to his partner in M', and the supposed stable matching M' is blocked by (m', w'). Each man m is therefore matched in M with his favorite stable partner w, and since E was an arbitrary execution of the algorithm, it follows that all possible executions of the algorithm leads to this same stable matching. □

This is a remarkable result. It implies that if each man is independently given his best stable partner, then the result is a stable matching. Yet there seems no a priori reason why this should even be a matching.

For obvious reasons, the stable matching generated by the man-oriented version of the Gale-Shapley algorithm is called *man-optimal*. If the roles of the sexes in the algorithm are interchanged, then the resulting *woman-optimal* stable matching, obtained by the *woman-oriented* version of the Gale-Shapley algorithm, is analogously optimal for the women. It may happen that the man and woman optimal stable matchings are identical, but this will not, in general, be the case. Throughout the book, we shall denote the man-optimal stable matching by M_0 and the woman-optimal by M_z.

It is perhaps not surprising that the optimality property from the point of view of the members of one sex is gained at the expense of the members of the other sex. Specifically, in the man-optimal stable matching, each woman has the worst partner that she can have in any stable matching, so that, to coin what seems an appropriate term, man-optimal is also *woman-"pessimal"*; likewise, woman-optimal is *man-pessimal.*

Theorem 1.2.3 *In the man-optimal stable matching, each woman has the worst partner that she can have in any stable matching.*

Proof Suppose not. Let M_0 be the man-optimal stable matching, and suppose there is a stable matching M' and a woman w such that w prefers $m = p_{M_0}(w)$ to $m' = p_{M'}(w)$. But then (m, w) blocks M' unless m prefers $p_{M'}(m)$ to $w = p_{M_0}(m)$, in contradiction of the fact that m has no stable partner better than his partner in M_0. □

Example The illustration in Figure 1.4 on page 10 shows that it can happen that the man-oriented and woman-oriented versions of the algorithm yield the same stable matching, in which case it is immediate, by combining the optimality and pessimality properties, that this is the unique stable matching for that instance.

The reader may verify that this is the case by executing the woman-oriented version of the algorithm.

Example The second illustration, this time of size 8, shows that different stable matchings can arise from the man-oriented and woman-oriented versions of the algorithm. The preference lists for this instance appear in Figure 1.5.

1	5	7	1	2	6	8	4	3	1	5	3	7	6	1	2	8	4
2	2	3	7	5	4	1	8	6	2	8	6	3	5	7	2	1	4
3	8	5	1	4	6	2	3	7	3	1	5	6	2	4	8	7	3
4	3	2	7	4	1	6	8	5	4	8	7	3	2	4	1	5	6
5	7	2	5	1	3	6	8	4	5	6	4	7	3	8	1	2	5
6	1	6	7	5	8	4	2	3	6	2	8	5	3	4	6	7	1
7	2	5	7	6	3	4	8	1	7	7	5	2	1	8	6	4	3
8	3	8	4	5	7	2	6	1	8	7	4	1	5	2	3	6	8

Men's Preferences | Women's Preferences

Figure 1.5: A stable marriage instance of size 8

The reader may verify, by applying the Gale-Shapley algorithm with men and then women as proposers, that the man-optimal and woman-optimal stable matchings are

$$M_0 = \{(1,5), (2,3), (3,8), (4,6), (5,7), (6,1), (7,2), (8,4)\}$$