Stable marriage problem and Gale-Shapley algorithm

Problem: There are n men and n women. Each man has a preference list/ranking of the women and each woman has a preference list of the men.

The goal is to find a *stable matching*, where a *matching* is a one-to-one correspondence between men and women. Informally, a matching is stable if there is no pair of a man and a woman that are not married in the matching, but that both could improve by marrying each other.

Formally & Notation: We denote men by capital letters and women by small letters.

A *matching* $\mathcal{M}$ is a one-to-one correspondence/bijection between men and women.

Aa (in $\mathcal{M}$) ... man A is married to woman a in the matching $\mathcal{M}$

AaB ... woman a prefers man A to man B

aAb ... man A prefers woman a to woman b

(B, a) is a *blocking pair* for a matching $\mathcal{M}$ if Ba (in $\mathcal{M}$) is not true but rather

$$Aa \text{ (in } \mathcal{M}), Bb \text{ (in } \mathcal{M}) \text{ and } BaA, aBb.$$

A *stable matching* is a matching without any blocking pair.

Application: E.g., assigning TAs to courses.

Let us look at an example. We are given four men A, B, C, D and four women a, b, c, d with the following preference lists.

A	c	b	d	a
B	b	a	c	d
C	b	d	a	c
D	c	a	d	b

a	A	B	D	C
b	C	A	D	B
c	C	B	D	A
d	B	A	C	D

$\mathcal{M}_1 = (Aa, Bb, Cc, Dd)$ is a matching, but it is not stable because (A, b) is a blocking pair for $\mathcal{M}_1$. Indeed, we have Aa (in $\mathcal{M}_1$), Bb (in $\mathcal{M}_1$) and AbB, bAa.

$\mathcal{M}_2 = (Ad, Ba, Cb, Dc)$ is a stable matching as can be verified by checking all man-woman pairs as a candidate for a blocking pair (running time of checking is in $\mathcal{O}(n^2)$).

Does there always exist a stable matching? Is it uniquely defined? We will show that there always exists a stable matching by providing an algorithm to compute one (Gale-Shapley algorithm[1]). A stable matching is not necessarily uniquely defined as the following example shows:

A	a	b
B	b	a

a	B	A
b	A	B

Both (Aa, Bb) and (Ab, Ba) are stable matchings as can be easily verified.

The following algorithm by Gale & Shapley computes a stable matching.

[1] D. Gale & L. S. Shapley. *College Admissions and the Stability of Marriage.* The American Mathematical Monthly Vol. 69, No. 1 (Jan., 1962), pp. 9-15.

Algorithm 1 Gale-Shapley algorithm (man-oriented version)

```
1  assign each person to be free
2  while some man M is free
3      set w as the first woman on M's preference list to whom M has not yet proposed
4      if w is free
5          assign M and w to be married to each other and M and w to be not free anymore
6      else
7          if w prefers M to her current husband M'
8              assign M and w to be married to each other, M' to be free and M and w to be not
               free anymore ("w breaks up with M' and marries M instead")
9          else
10             w rejects M and M remains free
11         end if
12     end if
13 end while
14 return the matching consisiting of n married pairs
```

Note that we did not specify in Algorithm 1 which free man to choose in Line 2 in case there are several free men. We will see below that the output of Algorithm 1 is always the same no matter which free man we choose. Also note that we will have to show that w in Line 3 is well defined, that is there is always at least one woman on M's preference list to whom M has not yet proposed.

Let us apply Algorithm 1 to the example involving four men and four women from above. If there are several free men from which we can choose, we choose the first one with respect to the order of the alphabet.

A	B	C	D	a	b	c	d
~~free~~	~~free~~	~~free~~	~~free~~	~~free~~	~~free~~	~~free~~	~~free~~
~~free~~	~~free~~						

A	$w = c$	Ac	
B	$w = b$	Bb	
C	$w = b$	Cb	B free
B	$w = a$	Ba	
D	$w = c$	Dc	A free
A	$w = b$		
A	$w = d$	Ad	

Algorithm 1 returns (Ad, Ba, Cb, Dc) as output.

Now we want to show correctness of Algorithm 1.

Theorem 1 *For any given instance of the stable marriage problem, Algorithm 1 terminates (after not more than n^2 iterations of the while-loop) and the married pairs that it returns form a stable matching.*

Proof. The proof consists of several steps.

1. EACH ITERATION OF THE WHILE-LOOP IS WELL DEFINED, THAT IS IF A MAN M IS FREE, THERE IS A WOMAN w ON HIS PREFERENCE LIST TO WHOM HE HAS NOT YET PROPOSED.

   We make two observations:

   - Once a woman has been proposed to for the first time, she is always married to some man for the rest of the execution of the algorithm.
   - No man is married to two women at the same time in the execution of the algorithm.

   Assume now that M is free and there is no woman to whom he has not proposed yet. Then he must have proposed to all women before already and hence all women are married. But then n women are married to $n - 1$ men, none of which has two women. $\lightning$

2. The algorithm terminates after not more than n^2 iterations of the while-loop.

   In each iteration there is a new proposal (a man to a woman to whom he has not proposed yet) and there are only n^2 possible proposals.

3. When the algorithm terminates, it outputs a matching.

   When the algorithm terminates, every man is married with one woman since no man is free anymore and a man is never married to more than one woman (see above). Also, a woman is never married to more than one man in the execution of the algorithm, and hence the married pairs must form a matching.

4. The algorithm outputs a stable matching.

   Assume the matching is not stable. Then there exists a blocking pair (B, a). Let b' be the wife of B and A' be the husband of a in the output. Then we have BaA' and aBb'. Since B prefers a to b', B must have proposed to a at some point in the execution of the algorithm and either a rejected B immediately or B and a were married for some time and a broke up with B later. In both cases, we can conclude that a must have been married to some man C with CaB for some time. CaB and BaA' implies CaA', but in the execution of the algorithm, a woman can only improve her situation and get married to a better man. ↯(a cannot be married to A' at the end of the execution of the algorithm)

■

The next theorem shows that the output of Algorithm 1 does not depend on which free man we choose in case there are several free men. The stable matching returned by Algorithm 1 has the property that each man is married to the best woman he can have in any stable matching (*man-optimal* stable matching). Note that we can easily interchange the roles of men and women in Algorithm 1. This would yield the woman-oriented version of the Gale-Shapley algorithm, which returns the woman-optimal stable matching.

Theorem 2

(i) The output of Algorithm 1 does not depend on which free man we choose in case there are several free men from which we can choose.

(ii) In the output, each man is married to the best woman that he can have in any stable matching.

Proof. We will show that *(ii)* is true independent of which free man we choose. This immediately implies *(i)*.

Assume that in the output $\mathcal{U}$ of the algorithm man A is married to woman a', that is Aa' (in $\mathcal{U}$), but there exists a stable matching $\mathcal{M}$ with $A\hat{a}$ (in $\mathcal{M}$) and $\hat{a}Aa'$. Then in the execution of the algorithm, A must have proposed to $\hat{a}$ and either she rejected immediately or A and $\hat{a}$ were married for some time and $\hat{a}$ broke up with A later.

W.l.o.g. we may assume that when $\hat{a}$ rejected or broke up with A, it happened for the first time in the execution of the algorithm that a woman $\hat{a}$ rejected or broke up with a man A such that A and $\hat{a}$ are married in some stable matching (then, necessarily, A prefers $\hat{a}$ to his wife in $\mathcal{U}$).

The woman $\hat{a}$ rejected or broke up with A because of some man B with $B\hat{a}A$. Let b' be the wife of B in $\mathcal{M}$. Since $A\hat{a}$ (in $\mathcal{M}$), we have $b' \neq \hat{a}$. We must have $\hat{a}Bb'$ because otherwise $b'B\hat{a}$ and then b' would have rejected or broke up with B in the execution of the algorithm before $\hat{a}$ rejected or broke up with A. But since Bb' (in $\mathcal{M}$), then this contradicts our assumption from the previous paragraph.

Hence, we have $A\hat{a}$ (in $\mathcal{M}$), Bb' (in $\mathcal{M}$) and $B\hat{a}A$, $\hat{a}Bb'$. This means $(B, \hat{a})$ is a blocking pair for the stable matching $\mathcal{M}$. ↯ ■

It is not difficult to show that in the man-optimal stable matching, each woman is married to the worst man that she can have in any stable matching.