How to Have a Stable Relationship with Your Girlfriend/Boyfriend Using Linear Programming

Arnab Mitra

Setting: We have a group of \( n \) men and \( n \) women who want to be paired off. We assume:

1. Everyone has a complete preference list of partners (with no ties).

2. Everyone is heterosexual and being in a relationship is preferred to being single.

In the above setting if we just pair men and women arbitrarily, there can always be two couples \((m, w)\) and \((m', w')\) such that \( m \) rates \( w' \) higher than \( w \) in his preference list and \( w' \) rates \( m \) higher than \( m' \) in her list. Obviously such a situation is not stable as nothing stops \( m \) and \( w' \) from breaking off from their respective partners and getting paired themselves.

Definition 0.1. A stable matching is an assignment of \( n \) men to \( n \) women so that no two people prefer each other to their respective partners.

Now if \( m \) and \( w' \) pair up, this leaves \( m' \) and \( w \) free. Due to assumption (2) it is natural for them to pair up but this may lead to new clashes in the preference lists. This leads to the following question.

Question: Given a group of \( n \) men and \( n \) women satisfying the above two assumptions, can one always find a stable matching?

The answer is yes. Consider the following algorithm due to Gale and Shapley (see [1]).

Each man proposes, in order, to the women on his list, pausing when a woman agrees to consider his proposal, but continuing if a proposal is either immediately or subsequently rejected. When a woman receives a proposal, she rejects it if she already holds a better proposal, but otherwise agrees to hold it for consideration, simultaneously rejecting any poorer proposal that she may currently hold. (Here the better proposal is the one according to her preference list.) The algorithm stops when every woman holds a proposal.
To illustrate the workings of the above algorithm let us consider the following example. Let \( n = 4 \) and \( \{A, B, C, D\}, \{W, X, Y, Z\} \) be the set of males and females respectively. Let the following be the preference lists:

\[
\begin{align*}
A &\rightarrow (X, W, Y, Z), \\
B &\rightarrow (W, Y, X, Z), \\
C &\rightarrow (X, W, Z, Y), \\
D &\rightarrow (Z, Y, X, W)
\end{align*}
\]

\[
\begin{align*}
W &\rightarrow (A, C, B, D), \\
X &\rightarrow (B, D, A, C), \\
Y &\rightarrow (C, A, B, D), \\
Z &\rightarrow (C, A, D, B)
\end{align*}
\]

It is easy to check that the final matching is going to be

\[
(A, X), (B, Y), (C, W), (D, Z).
\]

Observe that:

1) A woman \( w \) remains engaged from the point at which she receives her first proposal and the sequence of partners to which she is engaged gets better and better (in terms of her preference list).

2) The sequence of women to whom a man \( m \) proposes gets worse and worse (in terms of his preference list).

It is clear from the above observations that the G-S algorithm terminates. Let us now prove that the above algorithm always results in a stable matching.

**Proposition 0.2.** Consider an execution of the G-S algorithm that returns a set of pairs \( S \). The set \( S \) is a stable matching.

**Proof.** It is easy to see that the set \( S \) is a perfect matching. Let if possible it is not stable. Then there exists two pairs \((m, w)\) and \((m', w')\) such that \( m \) prefers \( w' \) to \( w \) and \( w' \) prefers \( m' \) to \( m \). This means \( m \) must have proposed \( w' \) and got rejected in favor of some man \( m'' \). Now since \( w' \) is with \( m' \), this means either \( m' = m'' \) or \( m' \) is higher than \( m'' \) in \( w' \)'s preference list. Since \( w' \) prefers \( m \) to \( m' \) this leads to a contradiction. \(\square\)

Call a woman \( w \) a valid partner of a man \( m \) (and vice versa) if there exists a stable matching that contains the pair \((m, w)\). We will say that \( w \) is the best valid partner of \( m \) if \( w \) is a valid partner of \( m \), and no woman whom \( m \) prefers more than \( w \) is a valid partner of his. We will denote best valid partner of \( m \) by \( \text{best}(m) \). Let \( S^* \) be the matching \( \{(m, \text{best}(m))\} \). The next proposition offers a nice description of the stable matching obtained by the G-S algorithm.

**Proposition 0.3.** Every execution of the G-S algorithm results in the set \( S^* \).

**Proof.** Let us suppose if possible that the algorithm results in a matching \( S \) in which some man is paired with a woman who is not his best valid partner. Since men propose in the decreasing order of preference
this means that some man is rejected by a valid partner during execution. So consider the first step during the execution in which some man, say $m$, is rejected by a valid partner $w$. Since this is the first time such a rejection has occurred $w = \text{best}(m)$. Let at the end of this step $w$ hold a proposal from $m'$ who she prefers to $m$.

Since $w$ is a valid partner of $m$, there exists a stable matching $S'$ containing $(m, w)$. Let $m'$ be paired to $w' (\neq w)$ in this matching. We know $w$ prefers $m'$ to $m$. If $m'$ also prefers $w$ to $w'$, then this contradicts the stability of $S'$ (as $(m', w) \notin S'$) which in turn contradicts our initial assumption. Thus to prove the proposition it suffices to show that $m'$ prefers $w$ to $w'$. Since the rejection $m$ by $w$ is the first rejection of any valid partner, $m'$ has not been rejected by $w'$ (who clearly is a valid partner of his) till that step. This implies $m'$ proposed $w$ before $w'$ and we are done. □

Proposition 0.3 indicates that the G-S algorithm favors the group that makes the proposals. The next proposition further reinforces the viewpoint.

**Proposition 0.4.** In the stable matching $S^*$, each woman is paired with her worst valid partner.

**Proof.** Suppose there were a pair $(m, w)$ in $S^*$ such that $m$ is not the worst valid partner of $w$. Then there is a stable matching $S'$ in which $w$ is paired with $m'$ such that $w$ prefers $m$ to $m'$. In $S'$, $m$ is paired with a woman $w' (\neq w)$; since $w = \text{best}(m)$ this means $m$ prefers $w$ to $w'$. This contradicts the stability of $S'$ (as $(m, w) \notin S'$). □

Because of this inherent bias in the G-S algorithm we need to look for more equitable algorithms for stable matching. Define the following quantities:

$$\begin{align*}
mr(i, k) &= j, \text{ if the woman } k \text{ is the } j \text{-th choice of the man } i \\
wr(i, k) &= j, \text{ if the man } k \text{ is the } j \text{-th choice of the woman } i.
\end{align*}$$

For a given stable matching $S = \{(m_1, w_1), ..., (m_n, w_n)\}$ define the value $c(S)$ of $S$ by

$$c(S) = \sum_{i=1}^{n} mr(m_i, w_i) + \sum_{i=1}^{n} wr(w_i, m_i).$$

We would like a stable matching $S$ which has the minimum possible value $c(S)$. More generally we can attach any positive value to every possible pairing and find a stable matching with the maximum (or minimum) total value. Call such a matching, an optimal stable matching.

Irving, Leather and Gusfield (in [2]) provide an algorithm to find an optimal stable matching by exploiting the one-to-one correspondence
between stable matchings and closed subsets of a certain partially ordered sets. We now give a formulation of the stable matching problem in terms of linear programming which gives us an easier way of obtaining an optimal stable matching. Before that we need to introduce notation.

Write \( x >_z y \) to denote that the person \( z \) prefers person \( x \) to \( y \). Let \( \mu \) denote the characteristic function of a matching \( S \) i.e. \( \mu(m) = w \) and \( \mu(w) = m \) if \((m, w) \in S\). Thus a matching \( S \) is stable if and only if there is no pair \( m \) and \( w \) such that both \( w >_m \mu(m) \) and \( m >_w \mu(w) \). Let \( M \) and \( W \) denote the set of men and women respectively. The incidence vector of \( \mu \), \( I \in \{0, 1\}^{|M| \times |W|} \) is such that \( I(m, w) = 1 \) if \( \mu(m) = w \) and \( I(m, w) = 0 \) otherwise. Define \( I(m >, w) \) to be \( \sum I(i, w) \), where the sum is over all men \( i \) such that \( m >_w i \), and \( I(M, w) = \sum I(i, w) \), where the sum is over all men. Similarly define \( I(m, > w) \) and \( I(M, W) \).

With this notation, we can characterize the incidence vector of a stable matching as an integer vector satisfying

\[
\begin{align*}
I(m, W) &= 1 \quad \forall m \in M \\
I(M, w) &= 1 \quad \forall w \in W \\
I(m >, w) - I(m, > w) &\leq 0 \quad \forall (m, w) \in M \times W \\
I(m, w) &\geq 0 \quad \forall (m, w) \in M \times W
\end{align*}
\]

Note that Condition (3) (along with Condition (1)) is the stability condition. Also it is clear that an integer vector \( I \in \mathbb{R}^{|M| \times |W|} \) is the incidence vector of a stable matching if and only if it satisfies (1)-(4).

The following theorem is due to Vande Vate ([3]).

**Theorem 0.5.** For any vector \( c = (c_{mw} : m \in M \text{ and } w \in W) \) of positive values, an extreme point optimal solution to the linear program

\[
\max cI
\]

such that

\[
\begin{align*}
I(m, W) &= 1 \quad \forall m \in M \\
I(M, w) &= 1 \quad \forall w \in W \\
I(m >, w) - I(m, > w) &\leq 0 \quad \forall (m, w) \in M \times W \\
I(m, w) &\geq 0 \quad \forall (m, w) \in M \times W
\end{align*}
\]

solves the optimal stable matching problem.

We refer the reader to the reference mentioned above for a proof.
References

