The Stable Matching Problem

1 The Stable Matching Problem

The Stable Matching problem often arises in practice whenever there is a need to match a set of “objects” of one kind with “objects” of another. There are numerous examples of such matching problems. Content delivery networks (CDNs) need to continually match incoming client requests with suitable servers. Employers are often competing against one another for the same set of potential employees – so there is an inherent matching being determined between the applicants and jobs. The first study of the Stable Matching problem was, in fact, done in the context of National Resident Matching Program, which matches medical residents to hospitals. Indeed, the algorithm we are going to discuss is still in use today in this program.

In the Stable Matching Problem, we have a set $W = \{w_1, w_2, \ldots, w_n\}$ of $n$ women and a set $M = \{m_1, m_2, \ldots, m_m\}$ of $n$ men. The set $M \times W$ denotes the set of all possible ordered pairs of the form $(m, w)$, where $m \in M$ and $w \in W$. A matching $S$ is a set of ordered pairs from $M \times W$ such that each man in $M$ and each woman in $W$ appears in at most one pair in $S$. A perfect matching $S'$ is a matching in which each man and each woman appears in exactly one pair in $S'$.

To define the stable matching problem, we introduce a notion of preferences. Each man ranks all of the women, and each woman ranks all of the men. For simplicity, we assume that there are no ties in any of the rankings. We say that a man $m$ prefers $w$ to $w'$ if $w$ ranks higher than $w'$ in $m$'s list.

Now we are ready to define a stable matching. A matching $S$ is said to be stable if $S$ is perfect, and there do not exist men $m$ and $m'$ and women $w$ and $w'$ such that $(m, w)$ and $(m', w')$ are in $S$, but $m$ prefers $w'$ to $w$ and $w'$ prefers $m$ to $m'$.

The Stable Matching Problem is to find a stable matching for a given set of men, women, and their preference lists.

2 The Gale-Shapley Algorithm

In 1962, David Gale and Lloyd Shapley, two mathematical economists, designed an elegant algorithm for solving the stable matching problem.

Initially all $m \in M$ and all $w \in W$ are single
While there is a single man who has not proposed to every woman
Choose one such man $m$
Let $w$ be the highest ranked woman on $m$'s list to whom $m$ has not proposed
3 Analysis of Gale-Shapley

We now analyze the Gale-Shapley algorithm. We show that it terminates fast and always returns a stable matching.

We first prove that the algorithm terminates quickly.

**Theorem 1** The Gale-Shapley algorithm terminates after at most $n^2$ iterations of the While loop.

**Proof:** We argue that the algorithm continually makes progress. In each iteration of the while loop, a single man proposes to the next woman in his preference list, somebody he has never proposed to before. Since there are $n$ men and each preference list has length $n$, there are at most $n^2$ proposals that can occur. So the number of iterations that can take place is at most $n^2$.

We next prove that the matching returned is stable. To do so, we make two observations: the first on the sequence of men that a woman gets engaged to, and the second on single men.

**Lemma 1** Every woman $w$ remains engaged from the point at which she receives her first proposal; from that point on, the partner to whom she gets engaged gets better and better.

**Proof:** Once a woman $w$ receives a proposal, she is engaged. And she continued to be engaged; if the algorithm breaks this engagement, it is only to match $w$ with another man. Furthermore, if $(m, w)$ is replaced by $(m', w)$ it is only if $w$ prefers $m'$ to $m$; so the partner to whom $w$ gets engaged gets better.

**Lemma 2** If $m$ is single at some point in the execution of the algorithm, there is some woman to whom he has not proposed.

**Proof:** The proof is by contradiction. Suppose there is a particular time in the execution of the algorithm when a man $m$ is single, yet has proposed to every woman. This means that by this
time, every woman has been proposed to at least once. By Lemma 1, we obtain that every woman is engaged. Thus, we have $n$ engaged women and hence $n$ engaged men, which implies that $m$ is also engaged contradicting our assumption that $m$ is single.

**Theorem 2** At termination, the Gale-Shapley algorithm returns a stable matching.

**Proof:** We first show that the matching returned is a perfect matching. The proof is by contradiction. Suppose not, then there is a man who is single at the end of the algorithm. By Lemma 2, it means $m$ has not proposed to some woman. But then, the algorithm would not exited out of the while loop, yielding the desired contradiction.

Now we show that the matching returned is stable. Again the proof is by contradiction. Suppose there exist men $m$ and $m'$ and women $w$ and $w'$ such that $(m, w)$ and $(m', w')$ are in $S$, but $m$ prefers $w'$ to $w$ and $w'$ prefers $m$ to $m'$.

By the algorithm, $w$ is the last woman to whom $m$ proposed. Since $m$ prefers $w'$ to $w$, $m$ must have proposed to $w'$ prior to his proposal to $w$. At that time, or later, $w'$ was engaged to a man, say $m''$, whom she preferred more than $m$. At the end, $w'$ is engaged to $m'$. By Lemma 1, we find that $w'$ prefers $m'$ over $m''$ and prefers $m''$ over $m$; this implies that $w'$ prefers $m'$ over $m$, contradicting our assumption that $w'$ prefers $m$ to $m'$.