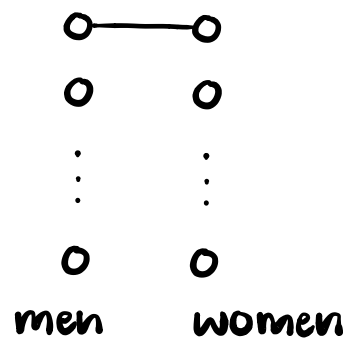
Ch 1. Stable Matching



Problem Definition:

$M = \{m_1, m_2, \ldots m_n\}$ n men

$W = \{w_1, w_2, \ldots w_n\}$ n women

Def. A "perfect" matching (1-1):
each man is matched with a woman.

$\{(m_1, w_1), (m_2, w_2), \ldots\}$

$\{(m_1, w_2), (m_2, w_3), \ldots\}$

Each perfect matching can be represented by a set of n pairs.

Each m/w has a preference list.

Def. A perfect matching is "unstable" if there exist $(m, w), (m', w') \in S$ s.t. m prefers w' to w, w' prefers m to m'.

Rmk. Can have >1 stable matching.

$m_1 : w_1 > w_2$ $\qquad$ $w_1 : m_2 > m_1$

$m_2 : w_2 > w_1$ $\qquad$ $w_2 : m_1 > m_2$

$$\Downarrow$$

both stable

Stable Matching Problem

Input: n & preference list of each m, w

Output: a stable matching S

Gale-Shapley Algorithm (G-S)

```
S = ϕ
while (∃ unmatched m):
    w ← highest ranked w to m to whom m has not proposed
      if (w is unmatched):
          add (m, w) to S
      else if (w prefers m to her current partner m')
          replace (m, w') by (m, w) in S
      else
          w reject m
return S
```

Observations

1. Once w is matched, she will never become unmatched.
2. m's partner ranking is always decreasing.
3. w's partner ranking is always increasing.

G-S algorithm will stop in n^2 steps.

Thm. G-S algorithm will output a perfect matching.

Pf. (By contradiction)

If the output S is not a perfect matching,
$\exists$ an unmatched m, w.
From the def. of alg.,

m has proposed to all the women,
$\Rightarrow$ m has proposed to w.
w is unmatched during the whole procedure
$\Rightarrow$ when m proposed to w, she is unmatched
$\Rightarrow$ w should be matched with m when he was proposing. ↯

Thm. G-S algorithm will output a stable matching.
Pf. (By contradiction)

If $\exists$ unstable edge in S,
(m, w), (m', w')

$$\begin{tikzcd} m \arrow[r, dash] & w \\ m' \arrow[r, dash] \arrow[ur, dotted, dash, "\text{unstable}"'] & w' \end{tikzcd}$$

m' prefers w to w',
w prefers m' to m.
m proposed to w' before w
w's partner ranking is always increasing
$\Rightarrow$ w prefers m to m'. ↯

Recall: there can be >1 stable matching for a problem.

$A: Y > X \qquad X: A > B$

$B: X > Y \qquad Y: B > A$

$A - X$	$A - Y$
$B - Y$	$B - X$
stable	stable

Different ordering can be used to execute the G-S algorithm.

Def. w is a valid partner of a m if $\exists$ stable matching S s.t. $(m, w) \in S$

Def. best(m) : best valid partner of m

Thm. G-S algorithm outputs sol. S^* s.t.

$S^* = \{(m_1, best(m_1)), (m_2, best(m_2)), \ldots$
$(m_n, best(m_n))\}$

Pf. Assume G-S outputs something other than S^*.

Assume in G-S execution,

m is the first man rejected by a valid woman w (w: best(m))

Case 1: m propose to w, but w is already matched w/ m'.
w prefers $m' > m$.

Case 2: m' propose to w,
w' prefers $m' > m$.

* replace (m, w) by (m^*, w)

$\exists$ m' s.t. w prefers $m' > m$

Since w is a valid partner of m,

$\exists$ stable S: $m - w$
unstable
$m' - w'$

Since m is the first man rejected by a valid partner, m' has not been rejected by his (valid partner) before m' matched w/ w.

including w' $\Rightarrow$ m' prefers $w > w'$ ↯
(contradicts def. that S = stable)

Thm. G-S algorithm
each w will be matched with the "worst" valid partner

Pf. (by ↯)

G-S outputs S $(m, w) \in S$

$\exists$ another m' which is w's worst valid partn

In the stable matching S' including (m', w)

$$\left.\begin{array}{l} m' - w \\ m - w' \end{array}\right\} \text{unstable} \rightarrow ↯$$

① w prefers m to m'

② G-S outputs (m, w) $\Rightarrow$ w = best(m)

$\Rightarrow$ m prefers w > w'