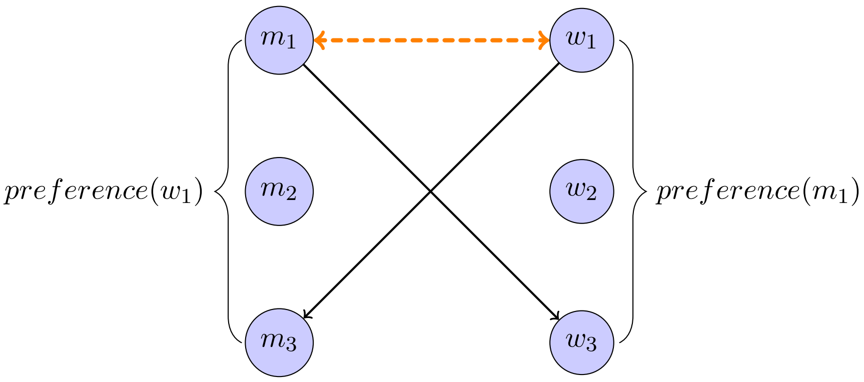
Stable Matching

1 Problem

Given two sets M of "men" and W of "women", where for each $m \in M$ and $w \in W$, there exists a complete sequence of preferences of members of the other set, find a mutually agreeable perfect matching.

Stability Mutually agreeable in this context is defined to be a matching with no **instability**. An unstable matching is defined as one with at least 1 pair of matches (m, w) and (m', w') such that m prefers w' to w and m' prefers w to w'. In this instance, all parties involved have an incentive to work outside the system and "elope".

Consider the following diagram:



In this case, because m_1 and w_1 mutually prefer each other to their currently matched partners, the matching is instable.

2 Algorithm

The Gale-Shapley algorithm (1962) computes a perfect stable matching for the problem in polynomial (technically linear) time.

```
Initialize all m ∈ M and w ∈ W to free
while ∃ free m who still has a woman w to propose to:
   if w is free:
      engage (m, w)
   else:
      let m’ be current partner of w
      if w prefers m to m’:
         free m’
         engage (m, w)
```

Essentially, the men iteratively take turns proposing to women by order of preference, and the women can reject, accept, and switch partners based on their preferences. Note that while the solution is stable, it may not necessarily produce an optimal matching from the women's point of view, because while the men have the entire set of women to choose from, the women only get to choose between a limited subset of the men at any given time.

3 Runtime

Once a man proposes to a woman, he will never propose to her again. She may reject his proposal right now or in the future, but he will only move further down his preference list. Therefore, each man will propose at most n times (all the way through the preference list). Because there are n men in the input, the algorithm runs in $O(n^2)$ time.

However, because each man and woman has a complete preference list, we can say that the input size is actually $2n^2$. Therefore, the Gale-Shapley algorithm runs in linear time.

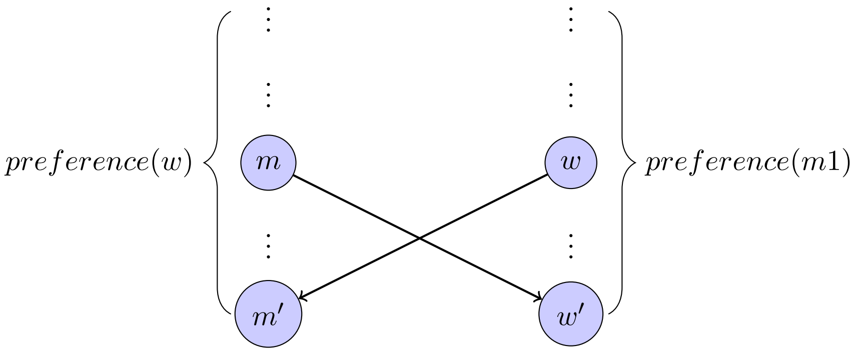
4 Proof

The proof comes in two parts: first, prove that the G-S algorithm produces a perfect matching (no unmatched people), and next, prove that the matching will always be stable.

Proof of Perfect Matching On termination, if some man m ends up without a woman, he must have proposed to and been rejected by every woman in his preferece list. There will also exist some $w \in W$ that is unengaged. Note that once a woman is engaged, she stays engaged for the whole algorithm (although her man can change), so this means that w must have been unproposed to.

However, because preference lists must be complete, w must have been in m's preference list, which leads to a contradiction because m must have proposed to every woman in his list for an imperfect matching to exist. By contradiction, the G-S algorithm therefore produces a perfect matching every time.

Proof of Stability Suppose, in the output, there exists some instability as below:



Before proposing to w', m must have proposed to w in this situation, because a man's partner only gets worse as the algorithm goes on. This means that w must have rejected m initially because she was paired with a more preferential mate m'', or broke off her engagement to m in favor of m''.

However, $m' < m < m''$, and since w would have been engaged to m'' at some point, she could never be proposed to m' because she either would have rejected him immediately or switched partners. By contradiction therefore, the G-S algorithm produces a stable matching.