Study guide about Gale-Shapley proofs

The real subject of this material is how to do a proof by contradiction. These proofs are just nice illustrations of the general techniques.

We will have a quiz on them next week, and this material will likely come up on exams.

• G-S never leaves anybody single.

  **Proof.** Suppose the negation of this statement, that some person of one gender is still single after some run on some instance of the problem. Since the number of men is the same as the number of women, some person of the opposite gender is also still single. The single man proposed to the single woman, and she must have rejected him. In the algorithm, no single woman rejects a proposal, a contradiction.

  The one statement that doesn’t follow logically from others is the negation of the claim we’re trying to prove, so it must be false. If it is false, then the claim is true.

• G-S always returns a stable matching.

  **Proof.** Suppose the negation, that there is some instance of the problem and some run of the algorithm on it where the set \( S \) of couples it returns is not stable.

  There exist \((m, w), (m', w') \in S\) such that \( m \) likes \( w' \) more than \( w \) and \( w' \) likes \( m \) more than \( m' \). (This is the definition of an instability.)

  Since \( m \) proposes in descending order of preference and he likes \( w' \) more than \( w \), he proposed to \( w' \) before he proposed to \( w \). It must be that \( w' \) rejected him for a series of one or more higher men on her preference list, culminating in \( m' \). She likes \( m' \) more than \( m \), contradicting that she likes \( m \) more than \( m' \).

  The only claim that didn’t follow logically was the negation of the claim we were trying to prove. This must be false, so the claim is true.

A woman \( w \) is *attainable* for a man \( m \) if there exists some stable matching \( S \) (not necessarily one produced by G-S) in which they are married. This is also what it means for \( m \) to be attainable for \( w \).

Let’s show that G-S marries every man to his highest attainable woman by showing that no man is ever rejected by an attainable woman during execution of the algorithm. This will mean that his proposals stop at his highest attainable woman.

• During an execution of \( G - S \), a man is never rejected by an attainable woman.

  **Proof.** Suppose the negation, which is that in some run on some instance of the problem, one or more men is rejected by women that are attainable for them.

  There must be a first point when a man \( m \) is rejected by an attainable woman \( w \), in favor of a man \( m' \). Since \( m \) is the first, \( m' \) hasn’t been rejected by any attainable woman yet.

  Since \( w \) is attainable for \( m \), there must be a stable matching \( S' \) where \((m, w)\) are a couple. Let \( w' \) be the wife of \( m' \) in \( S' \). Because of \( S' \), \( w' \) is attainable for \( m' \). When \( m' \) proposed to \( w \), he hadn’t been rejected by any attainable woman, so he hadn’t gotten to \( w' \) yet. He likes \( w \) more than \( w' \).
Since \( w \) rejected \( m \) in favor of \( m' \), she likes \( m' \) more than \( m \).
This means \( m' \) and \( w \) are an instability in \( S' \), contradicting the fact that \( S' \) is stable.
Once again, the only statement that didn’t follow logically from previous ones is the initial negation of the claim. It must be false, so the claim is true.

- G-S assigns each woman her lowest attainable man.
  For the proof, see the short proof of 1.8 on page 11.