Lecture 3 – The Stable Matching Problem
Reading: KT Sections 1.2 and 2.3

Partial content of these slides have been obtained from the official lecture slides that accompany the textbook. A complete set of slides can be found at:
http://www.cs.princeton.edu/~wayne/kleinberg-tardos/

Matching Residents to Hospitals

- **Goal.** Given a set of preferences among hospitals and medical school students, design a self-reinforcing admissions process.

- **Unstable pair:** applicant x and hospital y are unstable if:
  - x prefers y to its assigned hospital.
  - y prefers x to one of its admitted students.

- **Stable assignment.** Assignment with no unstable pairs.
  - Natural and desirable condition.
  - Individual self-interest will prevent any applicant/hospital deal from being made.

Stable Matching Problem

- **Input:**
  - Given n men and n women, with their rating of the opposite sex.
  - Each man lists women in order of preference from best to worst.
  - Each woman lists men in order of preference from best to worst.

- **Goal:**
  - Find a *suitable* matching.

Stable Matching Problem

- **Perfect matching:** everyone is matched monogamously.
  - Each man gets exactly one woman.
  - Each woman gets exactly one man.

- **Stability:** no incentive for some pair of participants to undermine assignment by joint action.
  - In matching M, an unmatched pair m-w is unstable if man m and woman w prefer each other to current partners.
  - Unstable pair m-w could each improve by eloping.

- **Stable matching:** perfect matching with no unstable pairs.

- **Stable matching problem.** Given the preference lists of n men and n women, find a stable matching if one exists.
Stable Matching Problem

- Q. Is assignment X-C, Y-B, Z-A stable?
- A. No. Bertha and Xavier would elope.

An unstable pair (m,w) could each improve by joint action.

Stable Matching Problem

- Q. Is assignment X-A, Y-B, Z-C stable?
- A. Yes.

Stable Roommate Problem

- Q. Do stable matchings always exist?
- A. Not obvious a priori.

Stable roommate problem.
- 2n people, each person ranks others from 1 to 2n-1.
- Assign roommate pairs so that no unstable pairs.

Observation. Stable matchings do not always exist for stable roommate problem.
Propose-And-Reject Algorithm

- The [Gale-Shapley 1962] deferred acceptance algorithm is an intuitive method that guarantees to find a stable matching.

Proof of Correctness: Termination

- Observation 1. Men propose to women in decreasing order of preference.

- Observation 2. Once a woman is matched, she never becomes unmatched; she only "trades up."

- Claim. Algorithm terminates after at most $n^2$ iterations of while loop.

- Pf. Each time through the while loop a man proposes to a new woman. There are only $n^2$ possible proposals.

<table>
<thead>
<tr>
<th>Man</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wyatt</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Victor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Proof of Correctness: Perfection

- Claim. All men and women get matched.

- Pf. (by contradiction)
  - Suppose, for sake of contradiction, that Zeus is not matched upon termination of algorithm.
  - Then some woman, say Amy, is not matched upon termination.
  - By Observation 2, Amy was never proposed to.
  - But, Zeus proposes to everyone, since he ends up unmatched.
Proof of Correctness: Stability

• Claim. No unstable pairs in a matching $S^*$.  
• Pf. (by contradiction)  
  • Suppose A-Z is an unstable pair: each prefers each other to partner in Gale-Shapley matching $S^*$.  
  • Case 1: Z never proposed to A.  
    $\Rightarrow$ Z prefers his GS partner to A.  
    $\Rightarrow$ A-Z is stable.  
  • Case 2: Z proposed to A.  
    $\Rightarrow$ A rejected Z (right away or later)  
    $\Rightarrow$ A prefers her GS partner to Z.  
    $\Rightarrow$ A-Z is stable.  
  $\Rightarrow$ In either case A-Z is stable, a contradiction.  

Summary

• Stable matching problem.  
  • Given n men and n women, and their preferences, find a stable matching if one exists.  
  • Gale-Shapley algorithm. Guarantees to find a stable matching for any problem instance.  
  • Q. How to implement GS algorithm efficiently?  
  • Q. If there are multiple stable matchings, which one does GS find?  

Efficient Implementation

• Efficient implementation. We describe $O(n^2)$ time implementation.  
  • Representing men and women.  
    • Assume men are named 1, ..., n.  
    • Assume women are named 1', ..., n'.  
  • Engagements.  
    • Maintain a list of free men, e.g., in a queue.  
    • Maintain two arrays $\text{wife}[m]$, and $\text{husband}[w]$.  
      • set entry to 0 if unmatched  
      • if $m$ matched to $w$ then $\text{wife}[m]=w$ and $\text{husband}[w]=m$  
  • Men proposing.  
    • For each man, maintain a list of women, ordered by preference.  
    • Maintain an array $\text{count}[m]$ that counts the number of proposals made by man $m$.  

Efficient Implementation

• Women rejecting/accepting.  
  • Does woman $w$ prefer man $m$ to man $m'$?  
  • For each woman, create inverse of preference list of men.  
  • Constant time access for each query after $O(n)$ preprocessing.  

<table>
<thead>
<tr>
<th>Amy</th>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
<th>4th</th>
<th>5th</th>
<th>6th</th>
<th>7th</th>
<th>8th</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pref</td>
<td>8</td>
<td>3</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Amy</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inverse</td>
<td>4th</td>
<td>8th</td>
<td>2nd</td>
<td>7th</td>
<td>5th</td>
<td>6th</td>
<td>1st</td>
<td>3rd</td>
</tr>
</tbody>
</table>

Amy prefers man 3 to 6  
since $\text{inverse}[3] \prec \text{inverse}[6]$
Stable Matching Summary

- Stable matching problem. Given preference profiles of n men and n women, find a stable matching.
  - No man and women prefer to be with each other than assigned partner.

- Gale-Shapley algorithm. Finds a stable matching in $O(n^2)$ time.

- Interesting facts in the version of GS where men propose,
  - Man-optimality. Each man receives best valid partner.
  - Woman-pessimal assignment. Each woman receives worst valid partner.