CSC2556

Lecture 5

Matching

- Stable Matching
- Kidney Exchange

[Slides: Ariel Procaccia]
Announcements

• Project proposal
  - Due: Mar 03 by 11:59PM
  - I have put up a few sample project ideas on Piazza.
  - If you have trouble finding a project idea, meet me.

• Structure
  - Problem space introduction
  - High-level research question
  - Prior work
  - Detailed goals

• Length: Ideally 1 page (2 pages max)
Stable Matching

• Recap Graph Theory:

• In graph $G = (V, E)$, a matching $M \subseteq E$ is a set of edges with no common vertices
  ➢ That is, each vertex should have at most one incident edge
  ➢ A matching is perfect if no vertex is left unmatched.

• $G$ is a bipartite graph if there exist $V_1, V_2$ such that $V = V_1 \cup V_2$ and $E \subseteq V_1 \times V_2$
Stable Marriage Problem

• Bipartite graph, two sides with equal vertices
  ➢ $n$ men and $n$ women (old school terminology 😞)

• Each man has a ranking over women & vice versa
  ➢ E.g., Eden might prefer Alice $\succ$ Tina $\succ$ Maya
  ➢ And Tina might prefer Tony $\succ$ Alan $\succ$ Eden

• Want: a perfect, stable matching
  ➢ Match each man to a unique woman such that no pair of man $m$ and woman $w$ prefer each other to their current matches (such a pair is called a “blocking pair”)

### Example: Preferences

<table>
<thead>
<tr>
<th></th>
<th>Albert</th>
<th>Diane</th>
<th>Emily</th>
<th>Fergie</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bradley</td>
<td>Emily</td>
<td>Diane</td>
<td>Fergie</td>
<td></td>
</tr>
<tr>
<td>Charles</td>
<td>Diane</td>
<td>Emily</td>
<td>Fergie</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Diane</th>
<th>Bradley</th>
<th>Albert</th>
<th>Charles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emily</td>
<td>Albert</td>
<td>Bradley</td>
<td>Charles</td>
<td></td>
</tr>
<tr>
<td>Fergie</td>
<td>Albert</td>
<td>Bradley</td>
<td>Charles</td>
<td></td>
</tr>
</tbody>
</table>
Example: Matching 1

<table>
<thead>
<tr>
<th></th>
<th>Albert</th>
<th>Diane</th>
<th>Emily</th>
<th>Fergie</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diane</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Emily</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fergie</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Diane</th>
<th>Bradley</th>
<th>Albert</th>
<th>Charles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bradley</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Albert</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Charles</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Diane</th>
<th>Emily</th>
<th>Fergie</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emily</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fergie</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Question: Is this a stable matching?
Example: Matching 1

<table>
<thead>
<tr>
<th>Albert</th>
<th>Diane</th>
<th>Emily</th>
<th>Fergie</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bradley</td>
<td>Emily</td>
<td>Diane</td>
<td>Fergie</td>
</tr>
<tr>
<td>Charles</td>
<td>Diane</td>
<td>Emily</td>
<td>Fergie</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Diane</th>
<th>Bradley</th>
<th>Albert</th>
<th>Charles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emily</td>
<td>Albert</td>
<td>Bradley</td>
<td>Charles</td>
</tr>
<tr>
<td>Fergie</td>
<td>Albert</td>
<td>Bradley</td>
<td>Charles</td>
</tr>
</tbody>
</table>

No, Albert and Emily form a **blocking pair**.
Example: Matching 2

<table>
<thead>
<tr>
<th></th>
<th>Albert</th>
<th>Diane</th>
<th>Emily</th>
<th>Fergie</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bradley</td>
<td>Emily</td>
<td>Diane</td>
<td>Fergie</td>
<td></td>
</tr>
<tr>
<td>Charles</td>
<td>Diane</td>
<td>Emily</td>
<td>Fergie</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Diane</th>
<th>Bradley</th>
<th>Albert</th>
<th>Charles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emily</td>
<td>Albert</td>
<td>Bradley</td>
<td>Charles</td>
<td></td>
</tr>
<tr>
<td>Fergie</td>
<td>Albert</td>
<td>Bradley</td>
<td>Charles</td>
<td></td>
</tr>
</tbody>
</table>

Question: How about this matching?
Example: Matching 2

<table>
<thead>
<tr>
<th></th>
<th>Albert</th>
<th>Diane</th>
<th>Emily</th>
<th>Fergie</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bradley</td>
<td>Emily</td>
<td>Diane</td>
<td>Fergie</td>
<td></td>
</tr>
<tr>
<td>Charles</td>
<td>Diane</td>
<td>Emily</td>
<td>Fergie</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Diane</th>
<th>Bradley</th>
<th>Albert</th>
<th>Charles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emily</td>
<td>Albert</td>
<td>Bradley</td>
<td>Charles</td>
<td></td>
</tr>
<tr>
<td>Fergie</td>
<td>Albert</td>
<td>Bradley</td>
<td>Charles</td>
<td></td>
</tr>
</tbody>
</table>

Yes! (Charles and Fergie are unhappy, but helpless.)
Does a stable matching always exist in the marriage problem?

Can we compute it in a strategyproof way?

Can we compute it efficiently?
Gale-Shapley 1962

- Men-Proposing Deferred Acceptance (MPDA):

1. Initially, no proposals, engagements, or matches are made.
2. While some man $m$ is unengaged:
   - $w \leftarrow m$’s most preferred woman to whom $m$ has not proposed yet
   - $m$ proposes to $w$
   - If $w$ is unengaged:
     - $m$ and $w$ are engaged
   - Else if $w$ prefers $m$ to her current partner $m'$
     - $m$ and $w$ are engaged, $m'$ becomes unengaged
   - Else: $w$ rejects $m$
3. Match all engaged pairs.
### Example: MPDA

<table>
<thead>
<tr>
<th></th>
<th>Diane</th>
<th>Emily</th>
<th>Fergie</th>
</tr>
</thead>
<tbody>
<tr>
<td>Albert</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bradley</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Charles</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Diane</th>
<th>Emily</th>
<th>Fergie</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diane</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Emily</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fergie</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **= proposed**
- **= engaged**
- **= rejected**
Running Time

• **Theorem:** DA terminates in polynomial time (at most $n^2$ iterations of the outer loop)

• **Proof:**
  - In each iteration, a man proposes to someone to whom he has never proposed before.
  - $n$ men, $n$ women $\rightarrow n \times n$ possible proposals
  - Can actually tighten a bit to $n(n - 1) + 1$ iterations

• At termination, it must return a perfect matching.
Stable Matching

• **Theorem:** DA always returns a stable matching.

• **Proof by contradiction:**
  - Assume \((m, w)\) is a blocking pair.
  
  - **Case 1:** \(m\) never proposed to \(w\)
    - \(m\) cannot be unmatched o/w algorithm would not terminate.
    - Men propose in the order of preference.
    - Hence, \(m\) must be matched with a woman he prefers to \(w\)
    - \((m, w)\) is not a blocking pair
Stable Matching

• **Theorem:** DA always returns a stable matching.

• **Proof by contradiction:**
  
  ➢ Assume \((m, w)\) is a blocking pair.

  ➢ **Case 2:** \(m\) proposed to \(w\)
    
    o \(w\) must have rejected \(m\) at some point
    o Women only reject to get better partners
    o \(w\) must be matched at the end, with a partner she prefers to \(m\)
    o \((m, w)\) is not a blocking pair
Men-Optimal Stable Matching

• The stable matching found by MPDA is special.

• **Valid partner:** For a man $m$, call a woman $w$ a valid partner if $(m, w)$ is in *some* stable matching.

• **Best valid partner:** For a man $m$, a woman $w$ is the best valid partner if she is a valid partner, and $m$ prefers her to every other valid partner.
  - Denote the best valid partner of $m$ by $\text{best}(m)$. 
Men-Optimal Stable Matching

• **Theorem:** Every execution of MPDA returns the “men-optimal” stable matching: every man is matched to his best valid partner.

  ➢ Surprising that this is a matching. E.g., it means two men cannot have the same best valid partner!

• **Theorem:** Every execution of MPDA produces the “women-pessimal” stable matching: every woman is matched to her worst valid partner.
Men-Optimal Stable Matching

• **Theorem:** Every execution of MPDA returns the men-optimal stable matching.

• **Proof by contradiction:**
  - Let $S$ = matching returned by MPDA.
  - $m \leftarrow$ first man rejected by $\text{best}(m) = w$
  - $m' \leftarrow$ the more preferred man due to which $w$ rejected $m$
  - $w$ is valid for $m$, so $(m, w)$ part of stable matching $S'$
  - $w' \leftarrow$ woman $m'$ is matched to in $S'$
  - We show that $S'$ cannot be stable because $(m', w)$ is a blocking pair.
**Men-Optimal Stable Matching**

- **Theorem:** Every execution of MPDA returns the men-optimal stable matching.
- **Proof by contradiction:**

\[ S \]

- Not yet rejected by a valid partner \( \Rightarrow \) hasn’t proposed to \( w' \) \( \Rightarrow \) prefers \( w \) to \( w' \)
- First to be rejected by best valid partner (\( w \))

\[ S' \]

- Rejects \( m \) because prefers \( m' \) to \( m \)
- Blocking pair

\[ m' \]
\[ m \]
\[ w \]
\[ m' \]
\[ m \]
\[ w \]
Strategyproofness

• **Theorem**: MPDA is strategyproof for men.
  ➢ We’ll skip the proof of this.
  ➢ Actually, it is group-strategyproof.

• But the women might gain by misreporting.

• **Theorem**: No algorithm for the stable matching problem is strategyproof for both men and women.
Women-Proposing Version

• Women-Proposing Deferred Acceptance (WPDA)
  ➢ Just flip the roles of men and women
  ➢ Strategyproof for women, not strategyproof for men
  ➢ Returns the women-optimal and men-pessimal stable matching
Extensions

• Unacceptable matches
  ➢ Allow every agent to report a partial ranking
  ➢ If woman $w$ does not include man $m$ in her preference list, it means she would rather be unmatched than matched with $m$. And vice versa.
  ➢ $(m, w)$ is blocking if each prefers the other over their current state (matched with another partner or unmatched)
  ➢ Just $m$ (or just $w$) can also be blocking if they prefer being unmatched than be matched to their current partner

• Magically, DA still produces a stable matching.
Extensions

• **Resident Matching (or College Admission)**
  - Men $\rightarrow$ residents (or students)
  - Women $\rightarrow$ hospitals (or colleges)
  - Each side has a ranked preference over the other side
  - But each hospital (or college) $q$ can accept $c_q > 1$ residents (or students)
  - Many-to-one matching

• An extension of Deferred Acceptance works
  - Resident-proposing (resp. hospital-proposing) results in resident-optimal (resp. hospital-optimal) stable matching
Extensions

• For ~20 years, most people thought that these problems are very similar to the stable marriage problem

• Roth [1985] shows:
  ➢ No stable matching algorithm is strategyproof for hospitals (or colleges).
Extensions

• Roommate Matching
  ➢ Still one-to-one matching
  ➢ But no partition into men and women
    o “Generalizing from bipartite graphs to general graphs”
  ➢ Each of $n$ agents submits a ranking over the other $n - 1$ agents

• Unfortunately, there are instances where no stable matching exist.
  ➢ A variant of DA can still find a stable matching if it exists.
  ➢ Due to Irving [1985]
NRMP: Matching in Practice

• 1940s: Decentralized resident-hospital matching
  ➢ Markets “unraveled”, offers came earlier and earlier, quality of matches decreased

• 1950s: NRMP introduces centralized “clearinghouse”

• 1960s: Gale-Shapley introduce DA

• 1984: Al Roth studies NRMP algorithm, finds it is really a version of DA!

• 1970s: Couples increasingly don’t use NRMP

• 1998: NRMP implements matching with couple constraints (stable matchings may not exist anymore…)

• More recently, DA applied to college admissions
Kidney Exchange
Incentives

• A decade ago kidney exchanges were carried out by individual hospitals
• Today there are nationally organized exchanges; participating hospitals have little other interaction
• It was observed that hospitals match easy-to-match pairs internally, and enroll only hard-to-match pairs into larger exchanges
• Goal: incentivize hospitals to enroll all their pairs
The strategic model

• Undirected graph, only pairwise matches
  ➢ Vertex = donor-patient pair
  ➢ Edge = compatibility

• Each agent controls a subset of vertices
  ➢ Possible strategy: hide some vertices (match internally), and only reveal others
  ➢ Utility of agent = # its matched vertices (self-matched + matched by mechanism)
The strategic model

- Mechanism:
  - Input: revealed vertices by agents (edges are public)
  - Output: matching

- Target: # matched vertices

- Strategyproof (SP): If no agent benefits from hiding vertices irrespective of what other agents do.
OPT is manipulable
OPT is manipulable
Approximating SW

• Theorem [Ashlagi et al. 2010]: No deterministic SP mechanism can give a $2 - \epsilon$ approximation

• Proof:

  ➢ No perfect matching exists.
  ➢ Any algorithm must match at most three blue nodes, or at most two gray nodes.
Approximating SW

• Theorem [Ashlagi et al. 2010]: No deterministic SP mechanism can give a $2 - \epsilon$ approximation

• Proof:

➢ Suppose the algorithm matches at most three blue nodes
  o Cannot match both blue nodes in the following graph, otherwise blue agent has an incentive to hide nodes.
  o Must return a matching of size 1 when a matching of size 2 exists.
Approximating SW

- **Theorem [Ashlagi et al. 2010]**: No deterministic SP mechanism can give a $2 - \epsilon$ approximation

- **Proof:**
  
  - Suppose the algorithm matches at most two gray nodes
    - Cannot match the gray node in the following graph, otherwise the gray agent has an incentive to hide nodes.
    - Must return a matching of size 1 when a matching of size 2 exists.
Approximating SW

• Theorem [Kroer and Kurokawa 2013]: No randomized SP mechanism can give a $\frac{6}{5} - \epsilon$ approximation.

• Proof: Homework!
SP mechanism: Take 1

• Assume two agents

• \text{MATCH}_{\{1\},\{2\}} mechanism:
  
  ➢ Consider matchings that maximize the number of “internal edges” for each agent.
  
  ➢ Among these return, a matching with max overall cardinality.
Another example
Guarantees

• \( \text{MATCH}_{{\{1\},\{2\}}} \) gives a 2-approximation
  
  ➢ Cannot add more edges to matching
  
  ➢ For each edge in optimal matching, one of the two vertices is in mechanism’s matching

• **Theorem (special case):** \( \text{MATCH}_{{\{1\},\{2\}}} \) is strategyproof for two agents.
Proof

• $M =$ matching when player 1 is honest, $M' =$ matching when player 1 hides vertices

• $M \Delta M'$ consists of paths and even-length cycles, each consisting of alternating $M, M'$ edges

What’s wrong with the illustration on the right?
Proof

• Consider a path in $MΔM'$, denote its edges in $M$ by $P$ and its edges in $M'$ by $P'$

• Consider sets $P_{11}, P_{22}, P_{12}$ containing edges of $P$ among $V_1$, among $V_2$, and between $V_1 - V_2$
  ➢ Same for $P'_{11}, P'_{22}, P'_{12}$

• Note that $|P_{11}| \geq |P'_{11}|$
  ➢ Property of the algorithm
Proof

• Case 1: $|P_{11}| = |P'_{11}|$

• Agent 2’s vertices don’t change, so $|P_{22}| = |P'_{22}|$

• $M$ is max cardinality $\Rightarrow |P_{12}| \geq |P'_{12}|$

• $U_1(P) = 2|P_{11}| + |P_{12}|$

$$\geq 2|P'_{11}| + |P'_{12}| = U_1(P')$$
Proof

• Case 2: $|P_{11}| > |P'_{11}|$

• $|P_{12}| \geq |P'_{12}| - 2$
  
  ➢ Every sub-path within $V_2$ is of even length
  
  ➢ Pair up edges of $P_{12}$ and $P'_{12}$, except maybe the first and the last

• $U_1(P) = 2|P_{11}| + |P_{12}|$
  
  $\geq 2(|P'_{11}| + 1) + |P'_{12}| - 2$
  
  $= U_1(P')$  ■
The case of 3 players
SP Mechanism: Take 2

• Let $\Pi = (\Pi_1, \Pi_2)$ be a bipartition of the players

• $\text{MATCH}_\Pi$ mechanism:
  - Consider matchings that maximize the number of “internal edges” and do not have any edges between different players on the same side of the partition
  - Among these return a matching with max cardinality (need tie breaking)
Eureka?

• Theorem [Ashlagi et al. 2010]: MATCH\(\Pi\) is strategyproof for any number of agents and any partition \(\Pi\).

• Recall: For \(n = 2\), MATCH\(\{\{1\},\{2\}\}\) is a 2-approximation.

• Question: \(n = 3\), MATCH\(\{\{1\},\{2,3\}\}\) approximation?
  1. 2
  2. 3
  3. 4
  4. More than 4
The Mechanism

• The Mix-and-Match mechanism:
  - Mix: choose a random partition $\Pi$
  - Match: Execute $\mathsf{MATCH}_\Pi$

• Theorem [Ashlagi et al. 2010]: Mix-and-Match is strategyproof and a 2-approximation.

• We only prove the approximation ratio.
Proof

• $M^* =$ optimal matching

• **Claim:** I can create a matching $M'$ such that
  ➢ $M'$ is max cardinality on each $V_i$, and
  ➢ $\Sigma_i |M'_{ii}| + \frac{1}{2} \Sigma_{i \neq j} |M'_{ij}| \geq \Sigma_i |M^*_{ii}| + \frac{1}{2} \Sigma_{i \neq j} |M^*_{ij}|$

  ➢ $M^{**} =$ max cardinality on each $V_i$
  ➢ For each path $P$ in $M^* \Delta M^{**}$, add $P \cap M^{**}$ to $M'$ if $M^{**}$ has more internal edges than $M^*$, otherwise add $P \cap M^*$ to $M'$
  ➢ For every internal edge $M'$ gains relative to $M^*$, it loses at most one edge overall ■
Proof

• Fix \( \Pi \) and let \( M^\Pi \) be the output of MATCH\( \Pi \)

• The mechanism returns max cardinality across \( \Pi \) subject to being max cardinality internally, therefore

\[
\sum_i |M^\Pi_{ii}| + \sum_{i \in \Pi_1, j \in \Pi_2} |M^\Pi_{ij}| \geq \sum_i |M'_i| + \sum_{i \in \Pi_1, j \in \Pi_2} |M'_{ij}|
\]
Proof

\[ \mathbb{E}[|M^\Pi|] = \frac{1}{2^n} \sum_\Pi \left( \sum_i |M^\Pi_{ii}| + \sum_{i \in \Pi_1, j \in \Pi_2} |M^\Pi_{ij}| \right) \]

\[ \geq \frac{1}{2^n} \sum_\Pi \left( \sum_i |M'_\Pi_{ii}| + \sum_{i \in \Pi_1, j \in \Pi_2} |M'_\Pi_{ij}| \right) \]

\[ = \sum_i |M'_\Pi_{ii}| + \frac{1}{2^n} \sum_\Pi \sum_{i \in \Pi_1, j \in \Pi_2} |M'_\Pi_{ij}| \]

\[ = \sum_i |M'_\Pi_{ii}| + \frac{1}{2} \sum_{i \neq j} |M'_\Pi_{ij}| \geq \sum_i |M^*_\Pi_{ii}| + \frac{1}{2} \sum_{i \neq j} |M^*_\Pi_{ij}| \]

\[ \geq \frac{1}{2} \sum_i |M^*_\Pi_{ii}| + \frac{1}{2} \sum_{i \neq j} |M^*_\Pi_{ij}| = \frac{1}{2} |M^*| \quad \blacksquare \]