1. **Representative Problems**

- stable matching
- five representative problems

Lecture slides by Kevin Wayne
Copyright © 2005 Pearson Addison Wesley
http://www.cs.princeton.edu/~wayne/kleinberg-tardos
1. **Representative Problems**

- stable matching
- five representative problems
## Gale-Shapley demo

### Hospitals' preference lists

<table>
<thead>
<tr>
<th></th>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
<th>4th</th>
<th>5th</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atlanta</td>
<td>Wayne</td>
<td>Val</td>
<td>Yolanda</td>
<td>Zeus</td>
<td>Xavier</td>
</tr>
<tr>
<td>Boston</td>
<td>Yolanda</td>
<td>Wayne</td>
<td>Val</td>
<td>Xavier</td>
<td>Zeus</td>
</tr>
<tr>
<td>Chicago</td>
<td>Wayne</td>
<td>Zeus</td>
<td>Xavier</td>
<td>Yolanda</td>
<td>Val</td>
</tr>
<tr>
<td>Dallas</td>
<td>Val</td>
<td>Yolanda</td>
<td>Xavier</td>
<td>Wayne</td>
<td>Zeus</td>
</tr>
<tr>
<td>Eugene</td>
<td>Wayne</td>
<td>Yolanda</td>
<td>Val</td>
<td>Zeus</td>
<td>Xavier</td>
</tr>
</tbody>
</table>

### Students' preference lists

<table>
<thead>
<tr>
<th></th>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
<th>4th</th>
<th>5th</th>
</tr>
</thead>
<tbody>
<tr>
<td>Val</td>
<td>Eugene</td>
<td>Atlanta</td>
<td>Boston</td>
<td>Dallas</td>
<td>Chicago</td>
</tr>
<tr>
<td>Wayne</td>
<td>Chicago</td>
<td>Boston</td>
<td>Dallas</td>
<td>Atlanta</td>
<td>Eugene</td>
</tr>
<tr>
<td>Xavier</td>
<td>Boston</td>
<td>Chicago</td>
<td>Dallas</td>
<td>Eugene</td>
<td>Atlanta</td>
</tr>
<tr>
<td>Yolanda</td>
<td>Atlanta</td>
<td>Eugene</td>
<td>Dallas</td>
<td>Chicago</td>
<td>Boston</td>
</tr>
<tr>
<td>Zeus</td>
<td>Dallas</td>
<td>Boston</td>
<td>Eugene</td>
<td>Chicago</td>
<td>Atlanta</td>
</tr>
</tbody>
</table>
Matching med-school students to hospitals

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

Unstable pair. Hospital $h$ and student $s$ form an unstable pair if both:
- $h$ prefers $s$ to one of its admitted students.
- $s$ prefers $h$ to assigned hospital.

Stable assignment. Assignment with no unstable pairs.
- Natural and desirable condition.
- Individual self-interest prevents any hospital–student side deal.
Stable matching problem: input

**Input.** A set of $n$ hospitals $H$ and a set of $n$ students $S$.
- Each hospital $h \in H$ ranks students.
- Each student $s \in S$ ranks hospitals.

<table>
<thead>
<tr>
<th>hospitals' preference lists</th>
<th>students' preference lists</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>favorite</strong></td>
<td><strong>least favorite</strong></td>
</tr>
<tr>
<td>1st</td>
<td>2nd</td>
</tr>
<tr>
<td>Atlanta</td>
<td>Xavier</td>
</tr>
<tr>
<td>Boston</td>
<td>Yolanda</td>
</tr>
<tr>
<td>Chicago</td>
<td>Xavier</td>
</tr>
</tbody>
</table>
Perfect matching

Def. A **matching** $M$ is a set of ordered pairs $h \rightarrow s$ with $h \in H$ and $s \in S$ s.t.

- Each hospital $h \in H$ appears in at most one pair of $M$.
- Each student $s \in S$ appears in at most one pair of $M$.

Def. A matching $M$ is **perfect** if $|M| = |H| = |S| = n$.

---

**How many perfect matchings?**

---

<table>
<thead>
<tr>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atlanta</td>
<td>Xavier</td>
<td>Yolanda</td>
</tr>
<tr>
<td>Boston</td>
<td>Yolanda</td>
<td>Xavier</td>
</tr>
<tr>
<td>Chicago</td>
<td>Xavier</td>
<td>Yolanda</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xavier</td>
<td>Boston</td>
<td>Atlanta</td>
</tr>
<tr>
<td>Yolanda</td>
<td>Atlanta</td>
<td>Boston</td>
</tr>
<tr>
<td>Zeus</td>
<td>Atlanta</td>
<td>Boston</td>
</tr>
</tbody>
</table>

---

**a perfect matching $M = \{ A-Z, B-Y, C-X \}$**
Unstable pair

Def. Given a perfect matching $M$, hospital $h$ and student $s$ form an unstable pair if both:

- $h$ prefers $s$ to matched student.
- $s$ prefers $h$ to matched hospital.

Key point. An unstable pair $h-s$ could each improve by joint action.
Stable matching problem

**Def.** A **stable matching** is a perfect matching with no unstable pairs.

**Stable matching problem.** Given the preference lists of \( n \) hospitals and \( n \) students, find a stable matching (if one exists).

- Natural, desirable, and self-reinforcing condition.
- Individual self-interest prevents any hospital-student pair from breaking commitment.

\[
\frac{n^2 \text{ pairs}}{n^2 - n} = n(n-1) \quad O(n^2) \quad \frac{n! \times n^2}{n!} \quad O(2^n 2^{n^2})
\]

<table>
<thead>
<tr>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atlanta</td>
<td>Xavier</td>
<td>Yolanda</td>
</tr>
<tr>
<td>Boston</td>
<td>Yolanda</td>
<td>Xavier</td>
</tr>
<tr>
<td>Chicago</td>
<td>Xavier</td>
<td>Yolanda</td>
</tr>
</tbody>
</table>

\[\text{a stable matching } M = \{ A-X, B-Y, C-Z \}\]
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.

<table>
<thead>
<tr>
<th></th>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
</tr>
<tr>
<td>B</td>
<td>C</td>
<td>A</td>
<td>D</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>B</td>
<td>D</td>
</tr>
<tr>
<td>D</td>
<td>A</td>
<td>B</td>
<td>C</td>
</tr>
</tbody>
</table>

no perfect matching is stable

$A-B, C-D \Rightarrow B-C$ unstable

$A-C, B-D \Rightarrow A-B$ unstable

$A-D, B-C \Rightarrow A-C$ unstable

Observation. Stable matchings need not exist.
Gale–Shapley deferred acceptance algorithm

An intuitive method that guarantees to find a stable matching.

\[ \text{GALE–SHAPLEY (preference lists for hospitals and students)} \]

**INITIALIZE** \( M \) to empty matching.

\( M = \{ \} \)

**WHILE** (some hospital \( h \) is unmatched and hasn’t proposed to every student)

\( s \leftarrow \) first student on \( h \)'s list to whom \( h \) has not yet proposed.

**IF** (\( s \) is unmatched)

Add \( h - s \) to matching \( M \).

\( (h, s) \rightarrow M \quad M = M \cup \{(h, s)\} \)

**ELSE IF** (\( s \) prefers \( h \) to current partner \( h' \))

Replace \( h' - s \) with \( h - s \) in matching \( M \).

\[ \{M - \{(h', s)\}\} \cup \{(h, s)\} \]

**ELSE**

\( s \) rejects \( h \).

\( s \) is paired with a higher program hospital.

**RETURN** stable matching \( M \).
Proof of correctness: termination

Observation 1. Hospitals propose to students in decreasing order of preference.

Observation 2. Once a student is matched, the student never becomes unmatched; only "trades up."

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

Pf. Each time through the while loop a hospital proposes to a new student. There are only $n^2$ possible proposals. ■

<table>
<thead>
<tr>
<th></th>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
<th>4th</th>
<th>5th</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atlanta</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
</tr>
<tr>
<td>Boston</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>A</td>
<td>E</td>
</tr>
<tr>
<td>Chicago</td>
<td>C</td>
<td>D</td>
<td>A</td>
<td>B</td>
<td>E</td>
</tr>
<tr>
<td>Dallas</td>
<td>D</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>E</td>
</tr>
<tr>
<td>Eugene</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
<th>4th</th>
<th>5th</th>
</tr>
</thead>
<tbody>
<tr>
<td>Val</td>
<td>W</td>
<td>X</td>
<td>Y</td>
<td>Z</td>
<td>V</td>
</tr>
<tr>
<td>Wayne</td>
<td>X</td>
<td>Y</td>
<td>Z</td>
<td>V</td>
<td>W</td>
</tr>
<tr>
<td>Xavier</td>
<td>Y</td>
<td>Z</td>
<td>V</td>
<td>W</td>
<td>X</td>
</tr>
<tr>
<td>Yolanda</td>
<td>Z</td>
<td>V</td>
<td>W</td>
<td>X</td>
<td>Y</td>
</tr>
<tr>
<td>Zeus</td>
<td>V</td>
<td>W</td>
<td>X</td>
<td>Y</td>
<td>Z</td>
</tr>
</tbody>
</table>

$n(n-1) + 1$ proposals required
Proof of correctness: perfection

Claim. Gale–Shapley produces a matching.
Pf. Hospital proposes only if unmatched; student

Claim. In Gale–Shapley matching, all hospitals get matched.
Pf. [by contradiction]
   • Suppose, for sake of contradiction, that some hospital \( h \in H \) is not matched upon termination of Gale–Shapley algorithm.
   • Then some student, say \( s \in S \), is not matched upon termination.
   • By Observation 2, \( s \) was never proposed to.
   • But, \( h \) proposes to every student, since \( h \) ends up unmatched.

Claim. In Gale–Shapley matching, all students get matched.
Pf.
   • By previous claim, all \( n \) hospitals get matched.
   • Thus, all \( n \) students get matched. •
Claim. In Gale-Shapley matching $M^*$, there are no unstable pairs.

pf. Suppose that $M^*$ does not contain the pair $h - s$.

- Case 1: $h$ never proposed to $s$.
  \[ h \text{ prefers its Gale-Shapley partner } s' \text{ to } s. \]
  \[ h - s \text{ is not unstable.} \]

- Case 2: $s$ rejected $h$ (right away or later).
  \[ s \text{ prefers Gale-Shapley partner } h' \text{ to } h. \]
  \[ h - s \text{ is not unstable.} \]

In either case, the pair $h - s$ is not unstable.
Summary

Stable matching problem. Given $n$ hospitals and $n$ students, and their preferences, find a stable matching if one exists.


Q. How to implement Gale–Shapley algorithm efficiently?
Q. If multiple stable matchings, which one does Gale–Shapley find?

COLLEGE ADMISSIONS AND THE STABILITY OF MARRIAGE

D. GALE* and L. S. SHAPLEY, Brown University and the RAND Corporation

1. Introduction. The problem with which we shall be concerned relates to the following typical situation: A college is considering a set of $n$ applicants of which it can admit a quota of only $q$. Having evaluated their qualifications, the admissions office must decide which ones to admit. The procedure of offering admission only to the $q$ best-qualified applicants will not generally be satisfactory, for it cannot be assumed that all who are offered admission will accept. Accordingly, in order for a college to receive $q$ acceptances, it will generally have to offer to admit more than $q$ applicants. The problem of determining how many and which ones to admit requires some rather involved guesswork. It may not be known (a) whether a given applicant has also applied elsewhere; if this is known it may not be known (b) how he ranks the colleges to which he has applied; even if this is known it will not be known (c) which of the other colleges will offer to admit him. A result of all this uncertainty is that colleges can expect only that the entering class will come reasonably close in numbers to the desired quota, and be reasonably close to the attainable optimum in quality.
Efficient implementation

Efficient implementation. We describe an $O(n^2)$ time implementation.

Representing hospitals and students. Index hospitals and students $1, \ldots, n$.

Representing the matching.
- Maintain a list of free hospitals (in a stack or queue).
- Maintain two arrays $student[h]$ and $hospital[s]$.
  - if $h$ matched to $s$, then $student[h] = s$ and $hospital[s] = h$
  - use value 0 to designate that hospital or student is unmatched

Hospitals proposing.
- For each hospital, maintain a list of students, ordered by preference.
- For each hospital, maintain a pointer to students in list for next proposal.
Students rejecting/accepting.

- Does student $s$ prefer hospital $h$ to hospital $h'$?
- For each student, create inverse of preference list of hospitals.
- Constant time access for each query after $O(n)$ preprocessing.

\[
\begin{array}{cccccccc}
\text{pref[]} & 1^{\text{st}} & 2^{\text{nd}} & 3^{\text{rd}} & 4^{\text{th}} & 5^{\text{th}} & 6^{\text{th}} & 7^{\text{th}} & 8^{\text{th}} \\
8 & 3 & 7 & 1 & 4 & 5 & 6 & 2 \\
\end{array}
\]

\[
\begin{array}{cccccccc}
\text{inverse[]} & 4^{\text{th}} & 8^{\text{th}} & 2^{\text{nd}} & 5^{\text{th}} & 6^{\text{th}} & 7^{\text{th}} & 3^{\text{rd}} & 1^{\text{st}} \\
1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
\end{array}
\]

student prefers hospital 3 to 6 since $\text{inverse}[3] < \text{inverse}[6]$

\[
\text{for } i = 1 \text{ to } n \\
\text{inverse}[\text{pref}[i]] = i
\]
Understanding the solution

For a given problem instance, there may be several stable matchings.
- Do all executions of Gale–Shapley yield the same stable matching?
- If so, which one?

<table>
<thead>
<tr>
<th></th>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atlanta</td>
<td>Xavier</td>
<td>Yolanda</td>
<td>Zeus</td>
</tr>
<tr>
<td>Boston</td>
<td>Yolanda</td>
<td>Xavier</td>
<td>Zeus</td>
</tr>
<tr>
<td>Chicago</td>
<td>Xavier</td>
<td>Yolanda</td>
<td>Zeus</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xavier</td>
<td>Boston</td>
<td>Atlanta</td>
<td>Chicago</td>
</tr>
<tr>
<td>Yolanda</td>
<td>Atlanta</td>
<td>Boston</td>
<td>Chicago</td>
</tr>
<tr>
<td>Zeus</td>
<td>Atlanta</td>
<td>Boston</td>
<td>Chicago</td>
</tr>
</tbody>
</table>

an instance with two stable matchings: $S = \{ A-X, B-Y, C-Z \}$ and $S' = \{ A-Y, B-X, C-Z \}$
Understanding the solution

**Def.** Student $s$ is a **valid partner** for hospital $h$ if there exists any stable matching in which $h$ and $s$ are matched.

**Ex.**
- Both Xavier and Yolanda are valid partners for Atlanta.
- Both Xavier and Yolanda are valid partners for Boston.
- Zeus is the only valid partner for Chicago.

<table>
<thead>
<tr>
<th></th>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atlanta</td>
<td>Xavier</td>
<td>Yolanda</td>
<td>Zeus</td>
</tr>
<tr>
<td>Boston</td>
<td>Yolanda</td>
<td>Xavier</td>
<td>Zeus</td>
</tr>
<tr>
<td>Chicago</td>
<td>Xavier</td>
<td>Yolanda</td>
<td>Zeus</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xavier</td>
<td>Boston</td>
<td>Atlanta</td>
<td>Chicago</td>
</tr>
<tr>
<td>Yolanda</td>
<td>Atlanta</td>
<td>Boston</td>
<td>Chicago</td>
</tr>
<tr>
<td>Zeus</td>
<td>Atlanta</td>
<td>Boston</td>
<td>Chicago</td>
</tr>
</tbody>
</table>

an instance with two stable matchings: $S = \{ A-X, B-Y, C-Z \}$ and $S' = \{ A-Y, B-X, C-Z \}$
Understanding the solution

**Def.** Student $s$ is a **valid partner** for hospital $h$ if there exists any stable matching in which $h$ and $s$ are matched.

**Hospital-optimal assignment.** Each hospital receives best valid partner.
- Is it perfect?
- Is it stable?

**Claim.** All executions of Gale–Shapley yield **hospital-optimal** assignment.

**Corollary.** Hospital-optimal assignment is a stable matching!
Claim. Gale–Shapley matching $S^*$ is hospital-optimal.

Pf. [by contradiction]
- Suppose a hospital is matched with student other than best valid partner.
- Hospitals propose in decreasing order of preference
  $\Rightarrow$ some hospital is rejected by valid partner during Gale–Shapley.
- Let $h$ be first such hospital, and let $s$ be the first valid student that rejects $h$.
- Let $M$ be a stable matching where $h$ and $s$ are matched.
- When $s$ rejects $h$ in Gale–Shapley, $s$ forms (or re-affirms) commitment to a hospital, say $h'$.
  $\Rightarrow$ $s$ prefers $h'$ to $h$.
- Let $s'$ be partner of $h'$ in $M$.
- $h'$ had not been rejected by any valid partner (including $s'$) at the point when $h$ is rejected by $s$.
- Thus, $h'$ had not yet proposed to $s'$ when $h'$ proposed to $s$.
  $\Rightarrow$ $h'$ prefers $s$ to $s'$.
- Thus $h-s'$ is unstable in $S$, a contradiction. $\blacksquare$
Student pessimality

Q. Does hospital-optimality come at the expense of the students?
A. Yes.

Student-pessimal assignment. Each student receives worst valid partner.


Pf. [by contradiction]
  - Suppose $h-s$ matched in $M^*$ but $h$ is not the worst valid partner for $s$.
  - There exists stable matching $M$ in which $s$ is paired with a hospital, say $h'$, whom $s$ prefers less than $h$.
    $\Rightarrow$ **s prefers $h$ to $h'$**.
  - Let $s'$ be the partner of $h$ in $M$. By hospital-optimality, $s$ is the best valid partner for $h$.
    $\Rightarrow$ **h prefers $s$ to $s'$**.
  - Thus, $h-s$ is an unstable pair in $M$, a contradiction. $\blacksquare$
Deceit: Machiavelli meets Gale–Shapley

Q. Can there be an incentive to misrepresent your preference list?
- Assume you know hospital’s propose-and-reject algorithm will be run.
- Assume preference lists of all other participants are known.

Fact. No, for any hospital; yes, for some students.

<table>
<thead>
<tr>
<th>hospitals' preference lists</th>
<th>students' preference lists</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st</td>
<td>2nd</td>
</tr>
<tr>
<td>A</td>
<td>X</td>
</tr>
<tr>
<td>B</td>
<td>Y</td>
</tr>
<tr>
<td>C</td>
<td>X</td>
</tr>
</tbody>
</table>

X lies

<table>
<thead>
<tr>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>B</td>
<td>C</td>
</tr>
<tr>
<td>Y</td>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>Z</td>
<td>A</td>
<td>B</td>
</tr>
</tbody>
</table>
Extensions

Extension 1. Some participants declare others as unacceptable.

Extension 2. Some hospitals have more than one position.

Extension 3. Unequal number of positions and students.

Def. Matching $M$ is **unstable** if there is a hospital $h$ and student $s$ such that:

- $h$ and $s$ are acceptable to each other; and
- Either $s$ is unmatched, or $s$ prefers $h$ to assigned hospital; and
- Either $h$ does not have all its places filled, or $h$ prefers $s$ to at least one of its assigned students.
Historical context

National resident matching program (NRMP).
- Centralized clearinghouse to match med-school students to hospitals.
- Began in 1952 to fix unraveling of offer dates.
- Originally used the "Boston Pool" algorithm.
- Algorithm overhauled in 1998.
  - med-school student optimal
  - deals with various side constraints
    (e.g., allow couples to match together)

The Redesign of the Matching Market for American Physicians:
Some Engineering Aspects of Economic Design

By Alvin E. Roth and Elliott Peranson*

We report on the design of the new clearinghouse adopted by the National Resident Matching Program, which annually fills approximately 20,000 jobs for new physicians. Because the market has complementarities between applicants and between positions, the theory of simple matching markets does not apply directly. However, computational experiments show the theory provides good approximations. Furthermore, the set of stable matchings, and the opportunities for strategic manipulation, are surprisingly small. A new kind of "core convergence" result explains this; that each applicant interviews only a small fraction of available positions is important. We also describe engineering aspects of the design process. (JEL C78, B41, J44)
2012 Nobel Prize in Economics


COLLEGE ADMISSIONS AND THE STABILITY OF MARRIAGE
D. GALE* AND L. S. SHAPLEY, Brown University and the RAND Corporation

1. Introduction. The problem with which we shall be concerned relates to the following typical situation: A college is considering a set of \( n \) applicants of which it can admit a quota of only \( q \). Having evaluated their qualifications, the admissions office must decide which ones to admit. The procedure of offering admission only to the \( q \) best-qualified applicants will not generally be satisfactory, for it cannot be assumed that all who are offered admission will accept.

Alvin Roth. Applied Gale–Shapley to matching med-school students with hospitals, students with schools, and organ donors with patients.