CSE 421: Introduction to Algorithms

Stable Matching

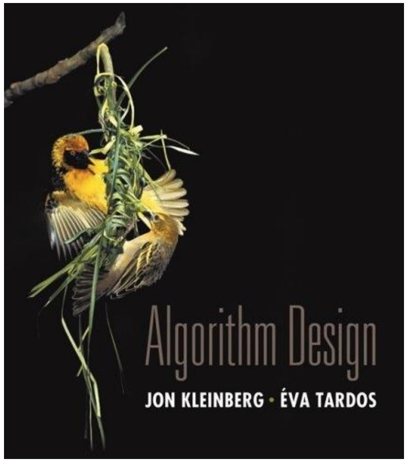
Shayan Oveis Gharan

Administrativia Stuffs

Lectures: M/W/F 2:30-3:20
Office hours: M/W/F 3:30-4:20

TAs:
- Tianchi Cao M 11:30-12:20
- Alireza Rezaei Tu 12:00-12:50
- Ben Jones Tu 2:00-2:50
- Eddie Huang W 9:00-9:50
- Robbie Weber Th 10:30-11:20

Grading Scheme
- Homework ~ 50%
- Midterm ~ 15-20%
- Final ~ 30-35%



Course textbook

CSE 421: Introduction to Algorithms

Winter, 2018

Shayan Oveis Gharan

MWF 2:30-3:20, MGH 389
Office hours in CSE 636
M/W/F 3:30-4:20

Textbook:

Algorithm Design by Jon Kleinberg and Eva Tardos, Addison-Wesley, 2006. We will cover almost all of chapters 1-8 of the Kleinberg/Tardos text plus some additional material from later chapters. In addition, I recommend reading chapter 5 of Introduction to Algorithms: A Creative Approach, by Udi Manber, Addison-Wesley 1989. This book has a unique point of view on algorithm design.

Another handy reference is Steven Skiena's Stonybrook Algorithm Repository

Grading Scheme (Roughly):

Homework 50%
Midterm 15-20%
Final Exam 30-35%

Website: cs.washington.edu/421

Matching Residents to Hospitals

Goal: Given a set of preferences among hospitals and medical school residents (graduating medical students), design a self-reinforcing admissions process.

Unstable pair: applicant **A** and hospital **Y** are unstable if:

A prefers **Y** to its assigned hospital.

Y prefers **A** to one of its admitted applicants.

Stable assignment. Assignment with no unstable pairs.

- Natural and desirable condition.
- Individual self-interest will prevent any applicant/hospital side deal from being made.

Simpler: Stable Matching Problem

Given **n** hetero men and **n** hetero women, find a "stable matching".

- Participants rate members of 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.

	favorite		least favorite
	1st	2nd	3rd
Xavier	Amy	Brenda	Claire
Yuri	Brenda	Amy	Claire
Zoran	Amy	Brenda	Claire

Men's Preference Profile

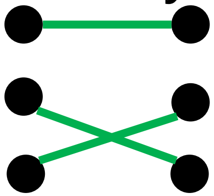
	favorite		least favorite
	1st	2nd	3rd
Amy	Yuri	Xavier	Zoran
Brenda	Xavier	Yuri	Zoran
Claire	Xavier	Yuri	Zoran

Women's Preference Profile

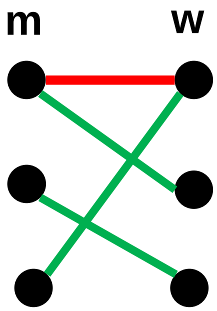
Stable Matching

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 a matching **M**, an unmatched pair **m**-**w** is unstable if man **m** and woman **w** prefer each other to current partners.

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.

Example

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

favorite ↓ … least favorite ↓

	1st	2nd	3rd
Xavier	Amy	Brenda	Claire
Yuri	Brenda	Amy	Claire
Zoran	Amy	Brenda	Claire

Men's Preference Profile

favorite ↓ … least favorite ↓

	1st	2nd	3rd
Amy	Yuri	Xavier	Zoran
Brenda	Xavier	Yuri	Zoran
Claire	Xavier	Yuri	Zoran

Women's Preference Profile

Example

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

Answer. No. Brenda and Xavier will hook up.

	1st (favorite)	2nd	3rd (least favorite)
Xavier	Amy	Brenda	Claire
Yuri	Brenda	Amy	Claire
Zoran	Amy	Brenda	Claire

Men's Preference Profile

	1st (favorite)	2nd	3rd (least favorite)
Amy	Yuri	Xavier	Zoran
Brenda	Xavier	Yuri	Zoran
Claire	Xavier	Yuri	Zoran

Women's Preference Profile

Example (cont'd)

Question: Is assignment X-A, Y-B, Z-C stable?

Answer: Yes.

favorite ↓ … least favorite ↓

	1st	2nd	3rd
Xavier	Amy	Brenda	Claire
Yuri	Brenda	Amy	Claire
Zoran	Amy	Brenda	Claire

Men's Preference Profile

favorite ↓ … least favorite ↓

	1st	2nd	3rd
Amy	Yuri	Xavier	Zoran
Brenda	Xavier	Yuri	Zoran
Claire	Xavier	Yuri	Zoran

Women's Preference Profile

Existence of Stable Matchings

Question. Do stable matchings always exist?
Answer. Yes, but 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.

	1st	2nd	3rd
Adam	B	C	D
Bob	C	A	D
Chris	A	B	D
David	A	B	C

A-B, C-D ⇒ B-C unstable
A-C, B-D ⇒ A-B unstable
A-D, B-C ⇒ A-C unstable

So, Stable matchings do not always exist for stable roommate problem.

Propose-And-Reject Algorithm [Gale-Shapley'62]

```
Initialize each person to be free.
while (some man is free and hasn't proposed to every woman) {
    Choose such a man m
    w = 1st woman on m's list to whom m has not yet proposed
    if (w is free)
        assign m and w to be engaged
    else if (w prefers m to her fiancé m')
        assign m and w to be engaged, and m' to be free
    else
        w rejects m
}
```

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 $\leq n^2$ iterations of while loop.

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

	1st	2nd	3rd	4th	5th
Victor	A	B	C	D	E
Walter	B	C	D	A	E
Xavier	C	D	A	B	E
Yuri	D	A	B	C	E
Zoran	A	B	C	D	E

	1st	2nd	3rd	4th	5th
Amy	W	X	Y	Z	V
Brenda	X	Y	Z	V	W
Claire	Y	Z	V	W	X
Diane	Z	V	W	X	Y
Erika	V	W	X	Y	Z

n(n-1) + 1 proposals required

Proof of Correctness: Perfection

Claim. All men and women get matched.

Proof. (by contradiction)

Suppose, for sake of contradiction, that Zoran is not matched upon termination of algorithm.

Then some woman, say Amy, is not matched upon termination.

By Observation 2 (only trading up, never becoming unmatched), Amy was never proposed to.

But, Zoran proposes to everyone, since he ends up unmatched.

■

Proof of Correctness: Stability

Claim. No unstable pairs.

Proof. (by contradiction)

Suppose **A**-**Z** is an unstable pair: each prefers each other to the 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.

men propose in decreasing order of preference

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.

women only trade up

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?
- Q: How many stable matchings are there?

Implementation of GS Algorithm

Problem size

- $N=2n^2$ words
  - $2n$ people each with a preference list of length n
- $2n^2 \log n$ bits
  - specifying an ordering for each preference list takes $n \log n$ bits

Brute force algorithm

- Try all $n!$ possible matchings
- Do any of them work?

Gale-Shapley Algorithm

- n^2 iterations, each costing constant time as follows:

Efficient Implementation

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

Representing men and women:

Assume men are named **1**, …, **n**.

Assume women are named **n+1**, …, 2**n**.

Engagements.

Maintain a list of free men, e.g., in a queue.

Maintain two arrays **wife**[**m**], and **husband**[**w**].

- set entry to **0** if unmatched
- if **m** matched to **w** then **wife**[**m**]=**w** and **husband**[**w**]=**m**

Men proposing:

For each man, maintain a list of women, ordered by preference.

Maintain an array **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 per woman.

O(n^2) total reprocessing cost.

Amy	1st	2nd	3rd	4th	5th	6th	7th	8th
Pref	8	3	7	1	4	5	6	2

Amy	1	2	3	4	5	6	7	8
Inverse	4th	8th	2nd	5th	6th	7th	3rd	1st

```
for i = 1 to n
   inverse[pref[i]] = i
```

Amy prefers man **3** to **6**

since `inverse[3]=2 < 7=inverse[6]`

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.

- GS algorithm finds a stable matching in $O(n^2)$ time.

- Q: If there are multiple stable matchings, which one does GS find?

- Q: How many stable matchings are there?

Understanding the Solution

Q. 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?

An instance with two stable matchings:

- A-X, B-Y.
- A-Y, B-X.

	1st	2nd
Xavier	A	B
Yuri	B	A

	1st	2nd
Amy	Y	X
Brenda	X	Y

Man Optimal Assignments

Definition: Man **m** is a valid partner of woman **w** if there exists some stable matching in which they are matched.

Man-optimal matching: Each man receives the best valid partner (according to his preferences).

- Simultaneously best for each and every man.

Claim: All executions of GS yield a man-optimal matching, which is a stable matching!

No reason a priori to believe that man-optimal matching is perfect, let alone stable.

Man Optimality

S
Amy-Yuri
Brenda-Zoran
. . .

Claim: GS matching **S*** is man-optimal.

Proof: (by contradiction)

Suppose some man is paired with someone other than his best partner. Men propose in decreasing order of preference $\Rightarrow$ some man is rejected by a valid partner.

Let **Y** be the man who is the first such rejection, and let **A** be the women who is first valid partner that rejects him.

Let **S** be a stable matching where **A** and **Y** are matched.

In building **S***, when **Y** is rejected, **A** forms (or reaffirms) engagement with a man, say **Z**, whom she prefers to **Y**.

Let **B** be **Z**'s partner in **S**.

In building **S***, **Z** is not rejected by any valid partner at the point when **Y** is rejected by **A**. Thus, **Z** prefers **A** to **B**.

since this is the **first** rejection by a valid partner

But **A** prefers **Z** to **Y**.

Thus **A-Z** is unstable in **S**. ▪

■

Man Optimality Summary

Man-optimality: In version of GS where men propose, each man receives the best valid partner.

> **w** is a valid partner of **m** if there exist some stable matching where **m** and **w** are paired

Q: Does man-optimality come at the expense of the women?

Woman Pessimality

Woman-pessimal assignment: Each woman receives the worst valid partner.

Claim. GS finds woman-pessimal stable matching **S***.

Proof.

- Suppose **A**-**Z** matched in **S***, but **Z** is not worst valid partner for **A**.
- There exists stable matching **S** in which **A** is paired with a man, say **Y**, whom she likes less than **Z**.
- Let **B** be **Z**'s partner in **S**.
- **Z** prefers **A** to **B**. ← man-optimality of **S***
- Thus, **A**-**Z** is an unstable in **S**.

■

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.
- GS algorithm finds a stable matching in $O(n^2)$ time. ✓
- GS algorithm finds man-optimal woman pessimal matching ✓
- Q: How many stable matching are there?

How many stable Matchings?

We already show every instance has at least 1 stable matchings.

There are instances with about c^n stable matchings for $c > 2$

[Research-Question]:

Is there an “efficient” algorithm that chooses a uniformly random stable matching of a given instance.

Extensions: Matching Residents to Hospitals

Men ≈ hospitals, Women ≈ med school residents.

- Variant 1: Some participants declare others as unacceptable.
- Variant 2: Unequal number of men and women.
- Variant 3: Limited polygamy.

e.g. A resident not interested in Cleveland

e.g. A hospital wants to hire **3** residents

Def: Matching **S** is unstable if there is hospital **h** and resident **r** s.t.

- **h** and **r** are acceptable to each other; and
- either **r** is unmatched, or **r** prefers **h** to her assigned hospital; and
- either **h** does not have all its places filled, or **h** prefers **r** to at least one of its assigned residents.

Lessons Learned

- Powerful ideas learned in course.
  - Isolate underlying structure of problem.
  - Create useful and efficient algorithms.

- Potentially deep social ramifications. [legal disclaimer]
  - Historically, men propose to women. Why not vice versa?
  - Men: propose early and often.
  - Men: be more honest.
  - Women: ask out the guys.
  - Theory can be socially enriching and fun!

"The Match": Doctors and Medical Residences

- Each medical school graduate submits a ranked list of hospital where he wants to do a residency

- Each hospital submits a ranked list of newly minted doctors

- A computer runs stable matching algorithm (extended to handle polygamy)

- Until recently, it was hospital-optimal.

History

1900

- Idea of hospital having residents (then called “interns”)

1900-1940s

- Intense competition among hospitals
  - Each hospital makes offers independently
  - Process degenerates into a race; hospitals advancing date at which they finalize binding contracts

1944

- Medical schools stop releasing info about students before a fixed date

1945-1949

- Hospitals started putting time limits on offers
  - Time limits down to 12 hours; lots of unhappy people

“The Match”

1950

- NICI run a centralized algorithm for a trial run
- The pairing was not stable, Oops!!

1952

- The algorithm was modified and adopted. It was called the Match.
- The first matching produced in April 1952