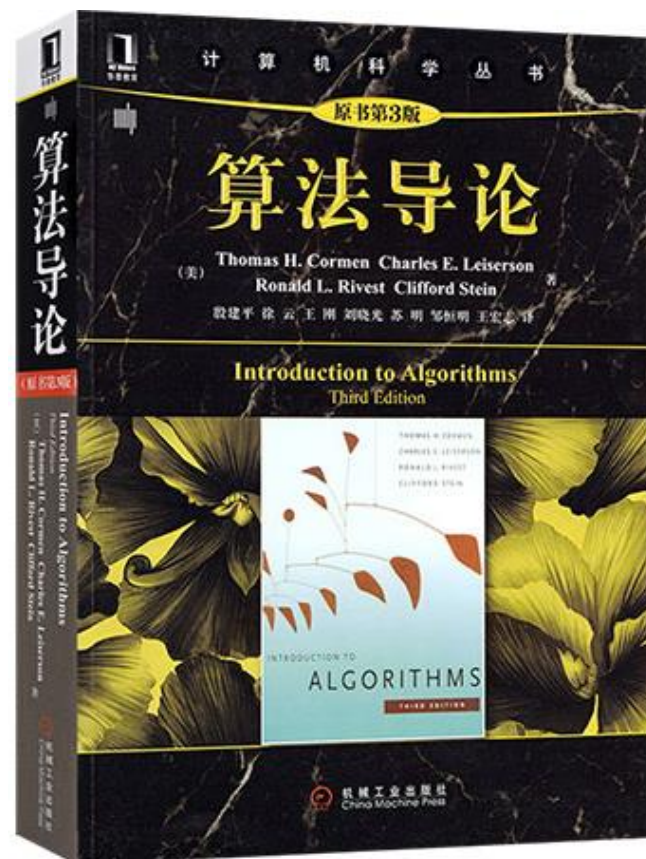
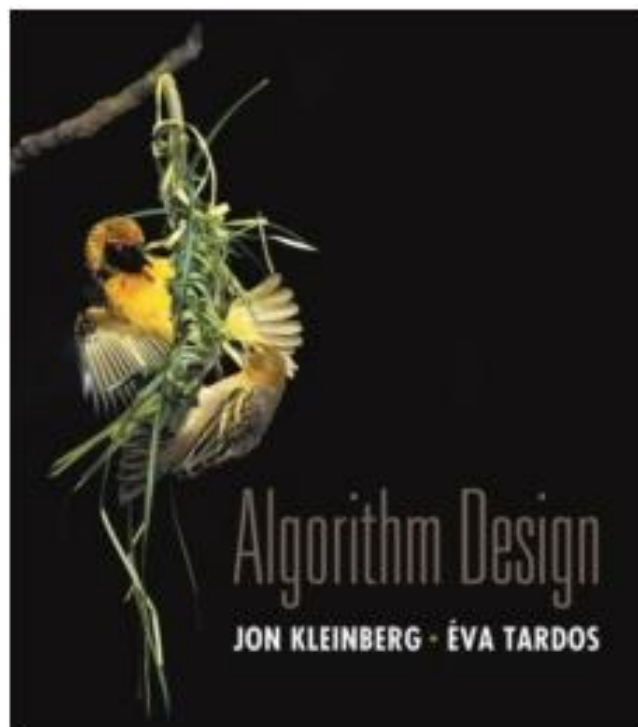


Algorithm Analysis & Design

Stable Matching

Prof. Xin Wang
xinwang@swpu.edu.cn



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 m_1, \dots, m_n
 and n hetero women w_1, \dots, w_n
 find a “stable matching”.

- Participants rate members of opposite sex.
- Each man lists women in order of preference.
- Each woman lists men in order of preference.

	favorite	least favorite		
	1st	2nd	3rd	
m_1	w_1	w_2	w_3	
m_2	w_2	w_1	w_3	
m_3	w_1	w_2	w_3	

	favorite	least favorite		
	1st	2nd	3rd	
w_1	m_2	m_1	m_3	
w_2	m_1	m_2	m_3	
w_3	m_1	m_2	m_3	

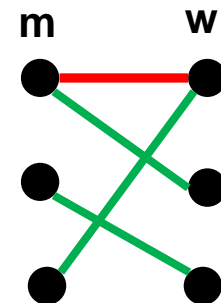
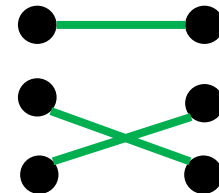
Stable Matching

Perfect matching:

- 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 $(m_1, w_3), (m_2, w_2), (m_3, w_1)$ stable?

	favorite ↓		least favorite ↓
	1st	2nd	3rd
m_1	w_1	w_2	w_3
m_2	w_2	w_1	w_3
m_3	w_1	w_2	w_3

	favorite ↓		least favorite ↓
	1st	2nd	3rd
w_1	m_2	m_1	m_3
w_2	m_1	m_2	m_3
w_3	m_1	m_2	m_3

Example

Question. Is assignment $(m_1, w_3), (m_2, w_2), (m_3, w_1)$ stable?

Answer. No. w_2, m_1 will hook up.

	favorite ↓		least favorite ↓
	1st	2nd	3rd
m_1	w_1	w_2	w_3
m_2	w_2	w_1	w_3
m_3	w_1	w_2	w_3

	favorite ↓		least favorite ↓
	1st	2nd	3rd
w_1	m_2	m_1	m_3
w_2	m_1	m_2	m_3
w_3	m_1	m_2	m_3

Example

Question: Is assignment $(m_1, w_1), (m_2, w_2), (m_3, w_3)$ stable?

Answer: Yes.

	favorite ↓		least favorite ↓
	1st	2nd	3rd
m_1	w_1	w_2	w_3
m_2	w_2	w_1	w_3
m_3	w_1	w_2	w_3

	favorite ↓		least favorite ↓
	1st	2nd	3rd
w_1	m_2	m_1	m_3
w_2	m_1	m_2	m_3
w_3	m_1	m_2	m_3

Existence of Stable Matchings

Question. Do stable matchings always exist?

Answer. ?

Stable roommate problem:

2n people; each person ranks others from **1** to **2n-1**.

Assign roommate pairs so that no unstable pairs.

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

A-B, C-D \Rightarrow B-C unstable
A-C, B-D \Rightarrow A-B unstable
A-D, B-C \Rightarrow 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  $m$   
}
```

First step: Properties of Algorithm

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

Observation 2: Each man proposes to each woman at most once

Observation 3: Once a woman is matched, she never becomes unmatched; she only "trades up."

What do we need to prove?

- 1) The algorithm ends
 - How many steps does it take?
- 2) The algorithm is correct [usually the harder part]
 - It outputs a perfect matching
 - The output matching is stable

1) Termination

Claim. Algorithm terminates after $\leq n^2$ iterations of while loop.

Proof. Observation 2: Each man proposes to each woman at most once.

Each man makes at most n proposals

So, 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

2) Correctness: Output is Perfect matching

Claim. All men and women get matched.

Proof. (by contradiction)

Suppose, for sake of contradiction, that m_1 is not matched upon termination of algorithm.

Then some woman, say w_1 , is not matched upon termination.

By Observation 3 (only trading up, never becoming unmatched), w_1 was never proposed to.

But, m_1 proposes to everyone, since he ends up unmatched.



2) Correctness: Stability

Claim. No unstable pairs.

Proof. (by contradiction)

Suppose m, w is an unstable pair: each prefers each other to the partner in Gale-Shapley matching S^* .

Obs1: men propose in

Case 1: m never proposed to w .

$\Rightarrow m$ prefers his S^* partner to w .

$\Rightarrow m, w$ is stable.

decreasing order of preference

Case 2: m proposed to w .

$\Rightarrow w$ rejected m (right away or later)

$\Rightarrow w$ prefers her S^* partner to m .

$\Rightarrow m, w$ is stable.

Obs2: women only trade up

In either case m, w 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?