CHAPTER 1
INTRODUCTION

Iris Hui-Ru Jiang
Fall 2017

Department of Electrical Engineering
National Taiwan University

Outline

- Content:
  - Opening: stable matching
  - Five representative problems
- Reading:
  - Chapter 1

Stable Matching

Opening topic

Opening

- We start from an algorithmic problem that illustrates many of the themes we will be emphasizing.
  1. It is motivated by some very natural and practical concerns.
  2. From these, we formulate a clean and simple statement of a problem.
  3. The algorithm to solve the problem is very clean as well.
  4. Most of our work will be spent in proving that it is correct and giving an acceptable bound on the amount of time it takes to terminate with an answer.

If You Were a Matchmaker…

- Imagine you are a matchmaker
  - One hundred female clients, and one hundred male clients
  - Each woman has given you a complete list of the hundred men, ordered by her preference: first choice, second choice, and so on
  - Each man has given you a list of the women, ranked analogously
  - Your job is to arrange one hundred happy marriages
    - Neither singlehood nor polygamy
    - Happy and stable (You don't want them to divorce or to develop outside relationships)

The Stable Matching Problem

- Given n men and n women, find a suitable matching
  - Participants rank members of opposite gender
  - 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 1st	2nd	least favorite 3rd
Xavier	Amy	Bertha	Clare
Yancey	Bertha	Amy	Clare
Zeus	Amy	Bertha	Clare

Men's Preference Profile

	favorite 1st	2nd	least favorite 3rd
Amy	Yancey	Xavier	Zeus
Bertha	Xavier	Yancey	Zeus
Clare	Xavier	Yancey	Zeus

Women's Preference Profile

Example (1/2)

- Try this way…
  - Z-A, Y-B, X-C
- Q: Is assignment X-C, Y-B, Z-A stable?
- A:

	favorite ↓ 1st	2nd	least favorite ↓ 3rd
Xavier	Amy	Bertha	Clare
Yancey	Bertha	Amy	Clare
Zeus	Amy	Bertha	Clare

Men's Preference Profile

	favorite ↓ 1st	2nd	least favorite ↓ 3rd
Amy	Yancey	Xavier	Zeus
Bertha	Xavier	Yancey	Zeus
Clare	Xavier	Yancey	Zeus

Women's Preference Profile

Example (2/2)

- Q: Is assignment X-C, Y-B, Z-A stable?
- A: No. Bertha and Xavier will hook up.
- Q: What has gone wrong?
- A: The process is not self-enforcing.

	favorite ↓ 1st	2nd	least favorite ↓ 3rd
Xavier	Amy	Bertha	Clare
Yancey	Bertha	Amy	Clare
Zeus	Amy	Bertha	Clare

Men's Preference Profile

	favorite ↓ 1st	2nd	least favorite ↓ 3rd
Amy	Yancey	Xavier	Zeus
Bertha	Xavier	Yancey	Zeus
Clare	Xavier	Yancey	Zeus

Women's Preference Profile

The 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 without unstable pairs.

- The stable matching problem: Given the preference lists of *n* men and *n* women, find a stable matching if one exists.
  - The preference lists are complete and have no ties

Simple but Invalid

- Random matching and fixing up

Simple-But-Invalid
1. Start with some matching
2. **while** (there is an unstable pair) **do**
3. Swap mates to make the pair stable

- Q: Valid?
- A: This will NOT work since a loop can occur. Swaps can be made that might continually result in unstable pairs.
  - Terminate?

Propose and Reject

2012 Nobel Prize in Economics

- Status: free (unmarried) ~ engaged ~ married
  - Deferred decision making: You may hesitate for a while before you are pretty sure you have got a reasonably good partner
- Goal: self-enforcing
  - [Gale-Shapley 1962] Intuitive method: act in one’s self-interest

```
Gale-Shapley
initialize each person to be free
while (some man m is free and hasn't proposed to every woman) do
    w = highest ranked woman in m's list to whom m has not yet proposed
    if (w is free) then
        (m, w) become engaged
    else if (w prefers m to her fiancé m') then
        (m, w) become engaged
        m' become free
return the set S of engaged pairs
```

- Q: Correct? (Stable? or even perfect?) Terminate?

D. Gale and L. S. Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, vol. 69, pp. 9-14, 1962.

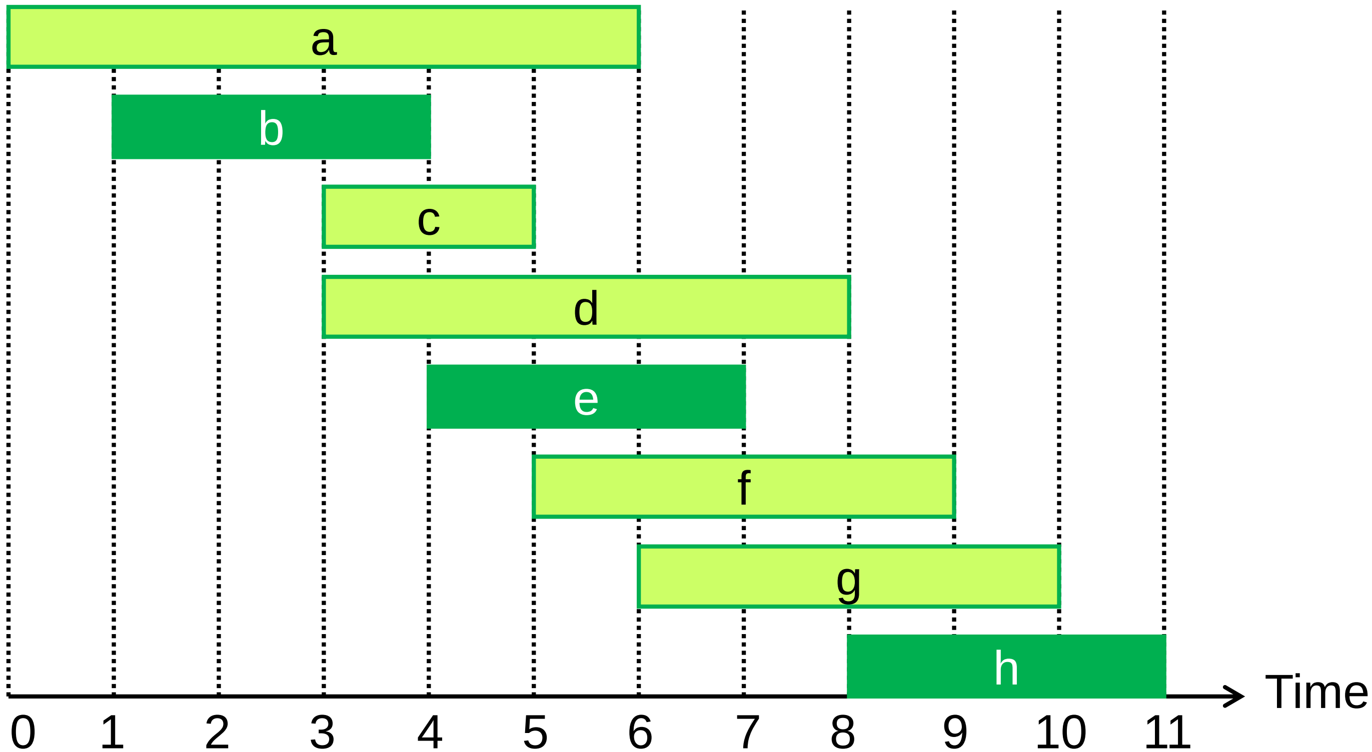
Example of Gale-Shapley Algorithm

Five Representative Problems

Interval Scheduling

- Given: Set of jobs with start times and finish times
- Goal: Find maximum cardinality subset of mutually compatible jobs

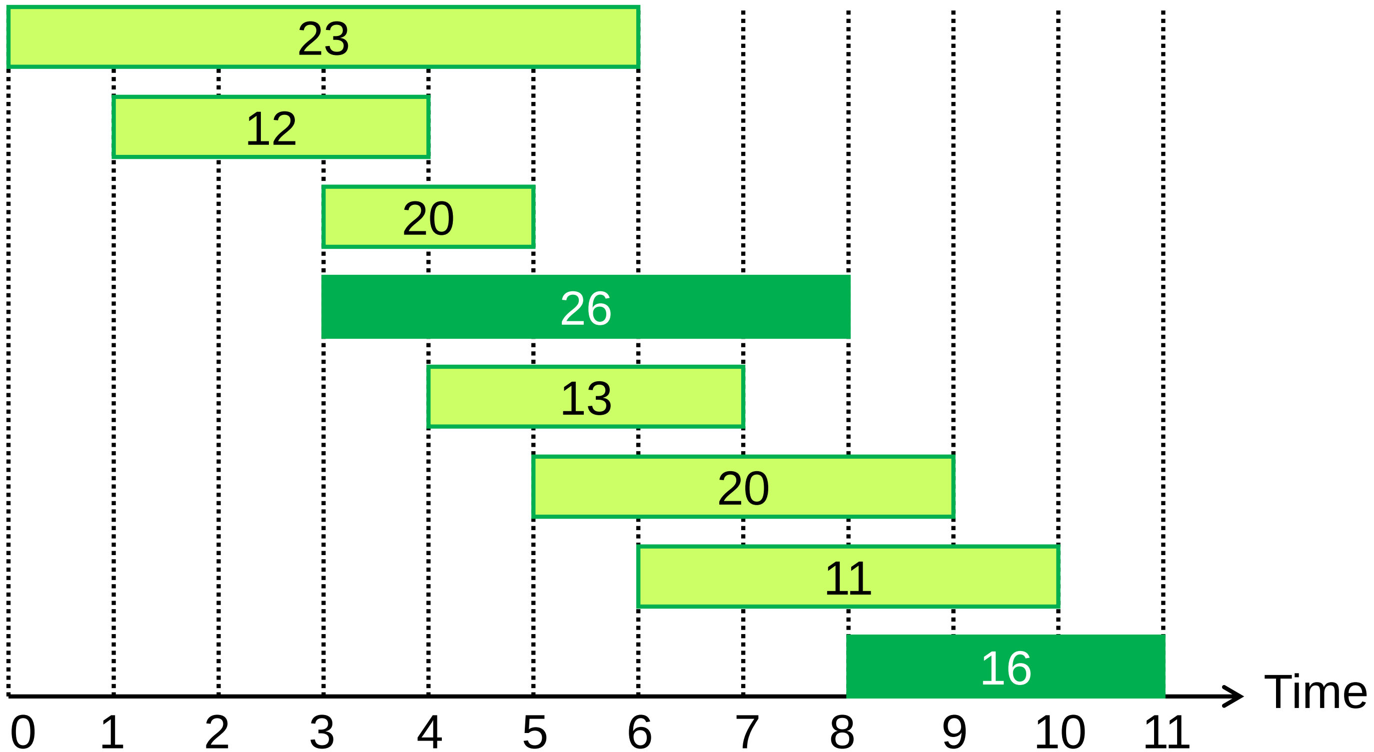
jobs don't overlap



Maximum compatible set {b, e, h}

Weighted Interval Scheduling

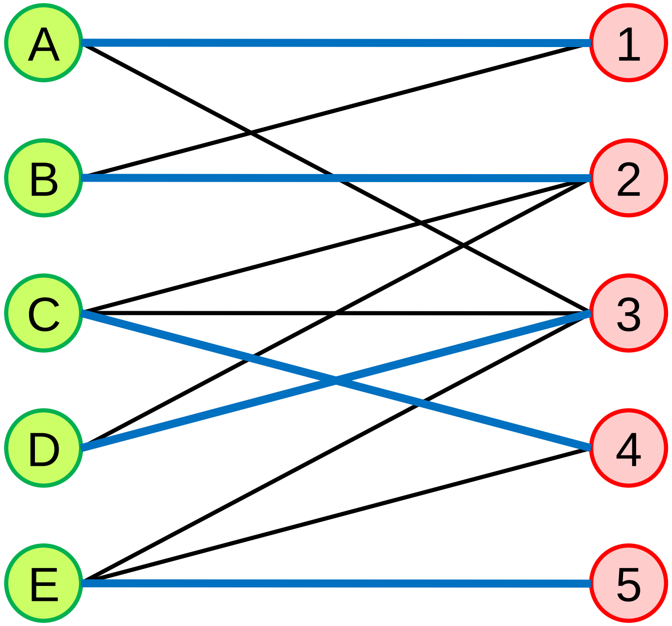
- Given: Set of jobs with start times, finish times, weights
- Goal: Find maximum weight subset of mutually compatible jobs



Maximum weighted compatible set {26, 16}

Bipartite Matching

- Given: Bipartite graph
- Goal: Find maximum cardinality matching

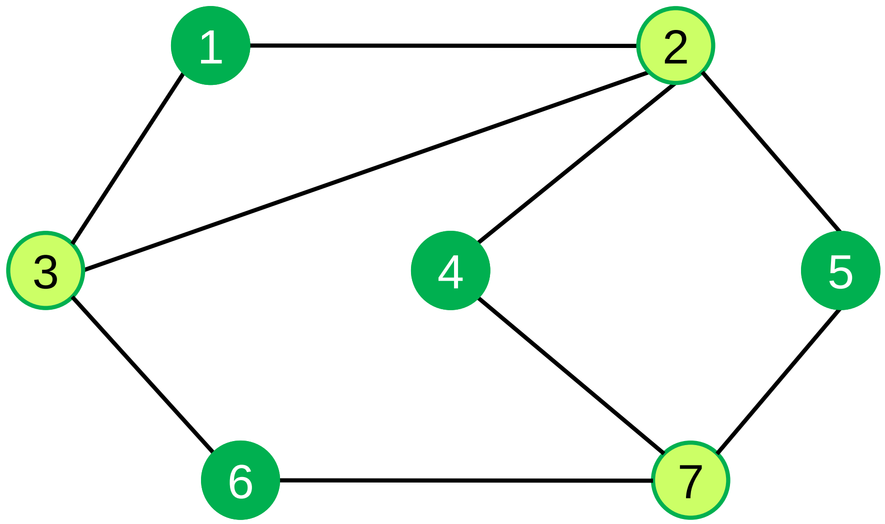


Maximum matching {(A, 1), (B, 2), (C, 4), (D, 3), (E, 5)}

Independent Set

- Given: Graph
- Goal: Find maximum cardinality independent set

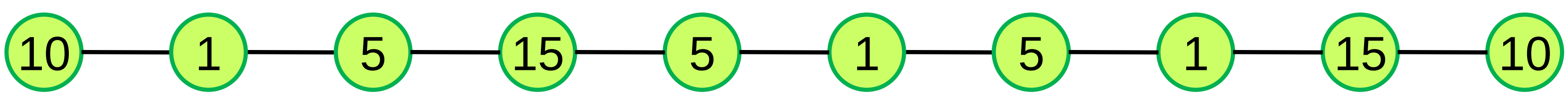
subset of nodes s.t. no two joined by an edge



Maximum independent set {1, 4, 5, 6}

Competitive Facility Location

- Given: Graph with weight on each node
- Game:
  - Two competing players alternate in selecting nodes
  - Do not allow to select a node if any of its neighbors have been selected
- Goal: Select a maximum weight subset of nodes



Second player can guarantee 20, but not 25

Five Representative Problems

- Efficiently solvable
  - Interval scheduling: $n \log n$ greedy algorithm
  - Weighted interval scheduling: $n \log n$ dynamic programming algorithm
  - Bipartite matching: n^k max-flow based algorithm
- Hard
  - Independent set: NP-complete
  - Competitive facility location: PSPACE-complete (even harder!)

Back to Stable Matching

Proofs

Formulating the Problem

- The stable matching problem

Clean!

- Given:
  - *M*: *n* men
  - *W: n* women
  - *M* x *W*: Each person has ranked all members of the opposite sex with a unique number between 1 and *n* in order of preference
- Goal:
  - Marry the men and women off such that
  - There are no two people of opposite sex who would both rather have each other than their current partners.
    - If there are no such people, all the marriages are stable.

Designing the Algorithm

- Correctness:
  - Termination: G-S terminates after at most n^2 iterations.
  - Perfection: Everyone gets married.
  - Stability: The marriages are stable.
- Male-optimal and female-pessimal
- All executions yield the same matching

Clean!

```
Gale-Shapley
initialize each person to be free
while (some man m is free and hasn't proposed to every woman) do
    w = highest ranked woman in m's list to whom m has not yet proposed
    if (w is free) then
        (m, w) become engaged
    else if (w prefers m to her fiancé m') then
        (m, w) become engaged
        m' become free
return the set S of engaged pairs
```

Correctness: Termination

- G-S terminates after at most n^2 iterations of while loop.
- Pf:
  - At each iteration, a man proposes to a new woman.
  - There are only n^2 possible proposals.

Don't count free people

- Observation 1: The sequence of women to whom *m* proposes gets worse and worse.
- Observation 2: Once a woman *w* receives her first proposal, *w* remains engaged; she only trades up.

	1st	2nd	3rd
Xavier	Amy	Bertha	Clare
Yancey	Bertha	Amy	Clare
Zeus	Amy	Bertha	Clare

Men's Preference Profile

	1st	2nd	3rd
Amy	Yancey	Xavier	Zeus
Bertha	Xavier	Yancey	Zeus
Clare	Xavier	Yancey	Zeus

Women's Preference Profile

Opening

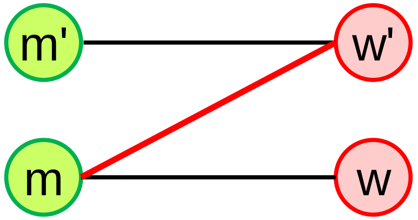
Correctness: Perfection

- The set *S* returned at termination is perfect matching.
  - i.e., all men get married
  - Prove by contradiction!
- Pf:
  - Suppose that G-S algorithm terminates with a free man *m.*
  - At termination, *m* had already proposed to every woman. Otherwise, the while loop would not have exited.
  - Since there is a free man *m*, there must be at least a free woman, say *w*. *w* was never proposed to.
  - This contradicts that *m* had already proposed to every woman.

```
Gale-Shapley
initialize each person to be free
while (some man m is free and hasn't proposed to every woman) do
    w = highest ranked woman in m's list to whom m has not yet proposed
    if (w is free) then
        (m, w) become engaged
    else if (w prefers m to her fiancé m') then
        (m, w) become engaged
        m' become free
return the set S of engaged pairs
```

Correctness: Stability

- Consider an execution returns S. Set S is a stable matching.
  - Assume an instability w.r.t. S and obtain a contradiction.
- Pf.
  - This instability involve two pairs, (m, w) and (m', w'), in S with
    - m prefers w' to w
    - w' prefers m to m'
  - In the execution, m's last proposal was w.
  - Did m propose to w' at some earlier point?
    - Case 1: No.
      - m must prefers w to w'. $\rightarrow\leftarrow$
    - Case 2: Yes.
      - m was rejected by w' because of m'', w' prefers m'' to m
      - $m'' = m'$ (final partner) $\rightarrow\leftarrow$
      - w' prefers her final partner m' to m'' $\rightarrow\leftarrow$



Uniqueness?

- Q: For a given instance, is stable matching is unique?
- A:

	1st	2nd	3rd
Xavier	Amy	Bertha	Clare
Yancey	Bertha	Amy	Clare
Zeus	Amy	Bertha	Clare

Men's Preference Profile

	1st	2nd	3rd
Amy	Yancey	Xavier	Zeus
Bertha	Xavier	Yancey	Zeus
Clare	Xavier	Yancey	Zeus

Women's Preference Profile

	1st	2nd	3rd
Xavier	Amy	Bertha	Clare
Yancey	Bertha	Amy	Clare
Zeus	Amy	Bertha	Clare

Men's Preference Profile

	1st	2nd	3rd
Amy	Yancey	Xavier	Zeus
Bertha	Xavier	Yancey	Zeus
Clare	Xavier	Yancey	Zeus

Women's Preference Profile

Male-Optimality

- $S^*=\{(m, \text{best}(m)): m \in M\}$: best possible outcome for all men
  - $w \in W$ is a valid partner of a man m if there is a stable matching that contains the pair (m, w)
  - $w = \text{best}(m)$: best valid partner of m if w is a valid partner of m and no woman whom m ranks higher than w is m's valid partner
- Every execution of G-S algorithm results in the set S^*
  - Male-optimal (and female-pessimal)
  - The same matching for all executions

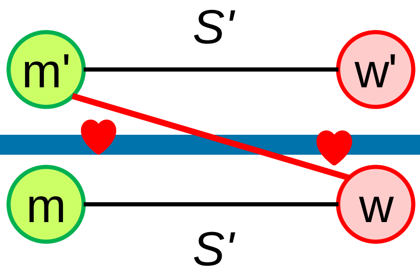
	1st	2nd	3rd
Xavier	Amy	Bertha	Clare
Yancey	Bertha	Amy	Clare
Zeus	Clare	Bertha	Amy

Men's Preference Profile

	1st	2nd	3rd
Amy	Yancey	Zeus	Xavier
Bertha	Xavier	Zeus	Yancey
Clare	Xavier	Yancey	Zeus

Women's Preference Profile

Male-Optimality



- G-S algorithm results in the set S^*
  - Again, by contradiction!
- Pf.
  - Suppose some execution E of G-S yields S where some man is paired with someone other than his best valid partner.
  - Since men propose in decreasing order of preference $\Rightarrow$ some man is rejected by a valid partner during E.
    - Let m be first such man & w be first valid woman who rejects m
    - w = best(m)
    - By definition, there is a stable matching S' containing (m, w).
  - In E, m was rejected by w because of $m' \Rightarrow w$ prefers m' to m
  - Assume $(m', w') \in S'$, $w' \neq w$. (w' is a valid partner of m')
  - In E, (m, w) is the first rejection of a valid pair
  - m' hadn't been rejected by any valid partner when m' is engaged to w in E.
  - m' proposed in decreasing order of preference (those women ranked before w are invalid).
  - And w' is a valid partner of $m' \Rightarrow m'$ must prefer w to w'
  - Since $(m', w) \notin S'$, it follows that (m', w) is an instability in S'. $\rightarrow\leftarrow$

What Did We Learn?

- Powerful ideas learned from stable matching
  - Isolate underlying structure of problem
  - Create effective and efficient algorithms
    - Effectiveness: Logical proofs
      - Useful technique: proof by contradiction
    - Efficiency will be detailed in the next chapter

For me, great algorithms are the poetry of computation.
Just like verse, they can be terse, allusive, dense, and even mysterious.
But once unlocked, they cast a brilliant new light on some aspect of computing.
-- Francis Sullivan

An algorithm must be seen to be believed.
The best way to learn what an algorithm is all about is to try it.
The reader should always take pencil and paper and work through an example...
-- Donald Knuth