Introduction and Representative Problems

CS 3AC3

Ryszard Janicki

Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada

Acknowledgments: Material based on *Algorithm Design* by Jon Kleinberg and Éva Tardos (Chapter 1)

Basic Information

Instructor: Dr. Ryszard Janicki, ITB 217, e-mail: janicki@mcmaster.ca, tel: 529-7070 ext: 23919
Teaching Assistants:

- Holly Koponen, e-mail: holly.koponen@gmail.com
- Atiyeh Sayadi, e-mail: atiye.sayadi@yahoo.com
- Yang Xiao, e-mail: xiaoy@mcmaster.ca
- Dennis Fong, e-mail: fongd1@mcmaster.ca

Course website: `http://www.cas.mcmaster.ca/~cs3ac3`,
Lectures: Monday: 12:30-1:20, Tuesday: 1:30-2:20 Thursday: 12:30-1:20, in TSH B128
Tutorial: Monday: 10:30-11:20, in KTH 104, Wednesday 12:30-1:20 in PGCLL M24, Wednesday 1:30-2:20 in T13 106, Friday 12:30-1:20 in T13 107; start January 15, 2024
Office hours: Monday 1:30-2:30.

Course Details

Calendar Description:
Basic computability models; the Church-Turing thesis, complexity classes; P versus NP; NP-completeness, reduction techniques; algorithmic design strategies; flows, distributed algorithms, advanced techniques such as randomization.

Prerequisites:
COMP SCI 2C03, COMP SCI 2FA3, or permission of instructor

Mission:
This course is designed to provide students with an understanding of the principles and techniques in the design and analysis of efficient data structures and algorithms. We shall discuss and analyze a variety of data structures and algorithms chosen for their importance and their illustration of fundamental concepts. We shall emphasize analyzing the worst-case running time of an algorithm as a function of input size. We shall also spend some time exploring the boundary between feasible (polynomial time) computations and infeasible computations. This will include discussion of the notorious P vs. NP question.

Outline of Course Topics and Texts

1. Introduction and Representative Problems
2. Greedy Algorithms
3. Divide-and-conquer
4. Dynamic programming
5. Network flow
6. Intractability and coping with intractability
7. Approximation algorithms
8. Randomized algorithms
9. Online algorithms

Texts:

- J. Kleinberg and E. Tardos, Algorithm Design, Addison-Wesley 2005 (main)
- M. Soltys, An Introduction to the Analysis of Algorithms, World Scientific 2012 (auxiliary)

The course may not always follow any text-book closely.

Evaluation

- **Evaluation:** There will be a 2.5 hours (one double sided cheat sheet will be allowed) final examination (50%), 60 minutes (plus 20 minutes for technology) midterm test (20%, take home, virtual on Avenue) and three assignments ($3 \times 10 = 30\%$).
- **Detailed grading scheme:** $Grade = 0.50 \times exam + 0.2 \times midterm + 0.1 \times (assg1 + assg2 + assg3)$
- *Late assignments will not be accepted.*
- Although you may discuss the general concept of the course material with your classmates, your assignment must be your individual effort.

Learning Objectives: Preconditions

Preconditions:

Students should have basic knowledge of discrete mathematics (especially of logic, sets and relations), basic knowledge of data structures and algorithms.

Learning Objectives: Postconditions

Postconditions:

A. Students should know and understand:

1. Greedy Algorithms
2. Divide-and-conquer
3. Dynamic programming
4. Network flow
5. Intractability and coping with intractability
6. Approximation algorithms
7. Randomized algorithms
8. Online algorithms

B. Students should be able to use and/or implement, dependently of the need, each of the below:

1. Greedy Algorithms
2. Divide-and-conquer
3. Dynamic programming
4. Network flow
5. Approximation algorithms
6. Randomized algorithms
7. Online algorithms

Stable Matching

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

Definition (Unstable pair)

Student x and hospital y are unstable if:

- x prefers y to its assigned hospital, and
- y prefers x to one of its admitted students.

Definition (Stable assignment)

It is an assignment with no unstable pairs. It is:

- Natural and desirable condition.
- Individual self-interest prevents any hospital-student side deal.

Stable Matching Problem

GOAL. Given a set of n men and a set of n women, find a "suitable" matching.

- Participants rank 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	Bertha	Clare
Yancey	Bertha	Amy	Clare
Zeus	Amy	Bertha	Clare

men's preference list

favorite ↓ … least favorite ↓

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

women's preference list

Perfect matching

Definition

A matching S is a set of ordered pairs m–w with $m \in M$ and $w \in W$ s.t.

- Each man $m \in M$ appears in at most one pair of S.
- Each woman $w \in W$ appears in at most one pair of S.

Definition

A matching S is perfect if $|S| = |M| = |W| = n$.

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

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

a perfect matching S = { X–C, Y–B, Z–A }

Unstable pair

Definition

Given a perfect matching S, man m and woman w are unstable if:

- m prefers w to his current partner.
- w prefers m to her current partner.

KEY POINT *An unstable pair m–w could each improve partner by joint action.*

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

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

Bertha and Xavier are an unstable pair

Stable matching problem

Definition

A stable matching is a 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).

- Natural, desirable, and self-reinforcing condition.
- Individual self-interest prevents any man–woman pair from eloping.

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

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

a perfect matching S = { X–A, Y–B, Z–C }

Stable roommate problem

Q. Do stable matchings always exist?
A. Not obvious a priori.

Definition (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
Doofus	A	B	C

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 for stable roommate problem.

Gale-Shapley deferred acceptance algorithm

- An intuitive method that guarantees to find a stable matching.

GALE–SHAPLEY (*preference lists for men and women*)

INITIALIZE S to empty matching.

WHILE (some man m is unmatched and hasn't proposed to every woman)

  $w \leftarrow$ first woman on m's list to whom m has not yet proposed.

  IF (w is unmatched)

    Add pair m–w to matching S.

  ELSE IF (w prefers m to her current partner m')

    Remove pair m'–w from matching S.

    Add pair m–w to matching S.

  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 at most 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
Wyatt	B	C	D	A	E
Xavier	C	D	A	B	E
Yancey	D	A	B	C	E
Zeus	A	B	C	D	E

	1st	2nd	3rd	4th	5th
Amy	W	X	Y	Z	V
Bertha	X	Y	Z	V	W
Clare	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

In Gale-Shapley matching, all men and women get matched.

Proof.

(by contradiction)

- Suppose, for sake of contradiction, that Zeus is not matched upon termination of GS algorithm.
- Then some woman, say Amy, is not matched upon termination.
- By Observation 2, Amy was never proposed to.
- But, Zeus proposes to everyone, since he ends up unmatched.

□

Proof of correctness: stability

Claim

In Gale-Shapley matching, there are no unstable pairs.

Proof.

Suppose the GS matching S^* does not contain the pair A–Z.

- Case 1: Z never proposed to A.
  - $\Rightarrow$ Z prefers his GS partner B to A. ← men propose in decreasing order of preference
  - $\Rightarrow$ A–Z is stable.
- Case 2: Z proposed to A.
  - $\Rightarrow$ A rejected Z (right away or later)
  - $\Rightarrow$ A prefers her GS partner Y to Z. ← women only trade up
  - $\Rightarrow$ A–Z is stable.
- In either case, the pair A–Z is stable. ▪

$A - Y$

$B - Z$

$\vdots$

Gale–Shapley matching S*

□

Stable matching problem. Given n men and n women, and their preferences, find a stable matching if one exists.

Theorem (Gale-Shapley 1962)

The 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?

Efficient implementation

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

Representing men and women.

- Assume men are named $1, \ldots, n$.
- Assume women are named $1', \ldots, n'$.

Representing the matching.

- Maintain a list of free men (in a stack or queue).
- Maintain two arrays $wife[m]$ and $husband[w]$.
  - if m matched to w, then $wife[m] = w$ and $husband[w] = m$
    set entry to 0 if unmatched

Men proposing.

- For each man, maintain a list of women, ordered by preference.
- For each man, maintain a pointer to woman in list for next proposal.

Efficient implementation (continued)

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.

pref[]

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

inverse[]

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

woman prefers man 3 to 6
since `inverse[3] < inverse[6]`

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

Understanding the solution

For a given problem instance, there may be several stable matchings.

- Do all executions of GS algorithm yield the same stable matching?
- If so, which one?

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

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

an instance with two stable matching: M = { A–X, B–Y, C–Z } and M' = { A–Y, B–X, C–Z }

Definition

Woman w is a valid partner of man m if there exists some stable matching in which m and w are matched.

Example

- Both Amy and Bertha are valid partners for Xavier.
- Both Amy and Bertha are valid partners for Yancey.
- Clare is the only valid partner for Zeus.

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

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

an instance with two stable matching: M = { A–X, B–Y, C–Z } and M' = { A–Y, B–X, C–Z }

Understanding the solution

Definition

Woman w is a valid partner of man m if there exists some stable matching in which m and w are matched.

Definition

Man-optimal assignment. Each man receives best valid partner.

- Is it perfect?
- Is it stable?

Claim

All executions of GS yield man-optimal assignment.

Corollary

Man-optimal assignment is a stable matching!

Man optimality

Claim

GS matching S^* is man-optimal.

Proof. (by contradiction).

- Suppose a man is matched with someone other than best valid partner.
- Men propose in decreasing order of preference
  $\Rightarrow$ some man is rejected by valid partner during GS.
- Let Y be first such man, and let A be the first valid woman that rejects him.
- Let S be a stable matching where A and Y are matched.
- When Y is rejected by A in GS, A forms (or reaffirms) engagement with a man, say Z.
  $\Rightarrow$ $\boxed{A \text{ prefers } Z \text{ to } Y.}$
- Let B be partner of Z in S.
- Z has not been rejected by any valid partner (including B) at the point when Y is rejected by A. ⟵ because this is the first rejection by a valid partner
- Thus, Z has not yet proposed to B when he proposes to A.
  $\Rightarrow$ $\boxed{Z \text{ prefers } A \text{ to } B.}$
- Thus A–Z is unstable in S, a contradiction. ▪

$A - Y$

$B - Z$

$\vdots$

stable matching S

Woman pessimality

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

A. Yes.

Definition

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

Claim

GS finds woman-pessimal stable matching S.*

Proof. (by contradiction).

- 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.

  $\Rightarrow$ $\boxed{A \text{ prefers } Z \text{ to } Y.}$
- Let B be the partner of Z in S. By man-optimality, A is the best valid partner for Z.

  $\Rightarrow$ $\boxed{Z \text{ prefers } A \text{ to } B.}$
- Thus, A–Z is an unstable pair in S, a contradiction. ▪

$$\begin{array}{c} A - Y \\ B - Z \\ \vdots \end{array}$$

stable matching S

Deceit: Machiavelli meets Gale-Shapley

Q. Can there be an incentive to misrepresent (i.e. cheat and lie) your preference list?

- Assume you know men's propose-and-reject algorithm will be run.
- Assume preference lists of all other participants are known.

Fact. No, for any man; yes, for some women.

men's preference list

	1st	2nd	3rd
X	A	B	C
Y	B	A	C
Z	A	B	C

women's preference list

	1st	2nd	3rd
A	Y	X	Z
B	X	Y	Z
C	X	Y	Z

Amy lies

	1st	2nd	3rd
A	Y	Z	X
B	X	Y	Z
C	X	Y	Z

Extensions: matching residents to hospitals

Ex: Men ≈ hospitals, Women ≈ med school residents.

Variant 1. Some participants declare others as unacceptable.

resident A unwilling to work in Cleveland

Variant 2. Unequal number of men and women.

Variant 3. Limited polygamy. ⟵ hospital X wants to hire 3 residents

Definition

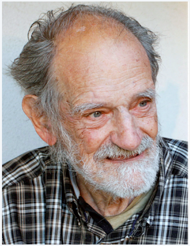
Matching S is unstable if there is a hospital h and resident r such that:

- 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.

2012 Nobel Prize in Economics

Lloyd Shapley. Stable matching theory and Gale-Shapley algorithm.

Alvin Roth. Applied Gale-Shapley to matching new doctors with hospitals, students with schools, and organ donors with patients.



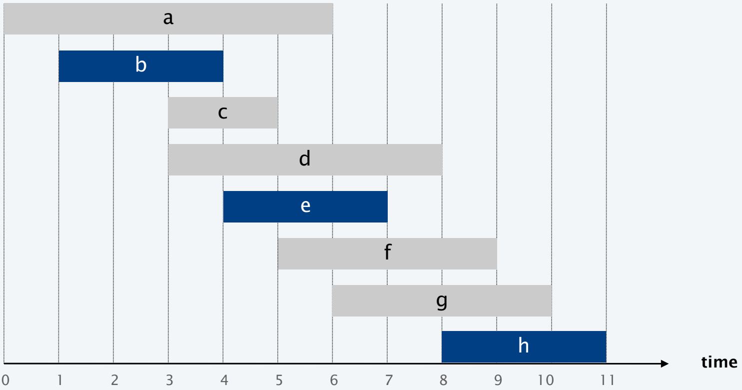
Lloyd Shapley

Alvin Roth

Input. Set of jobs with start times and finish times.

Goal. Find maximum cardinality subset of mutually compatible jobs.

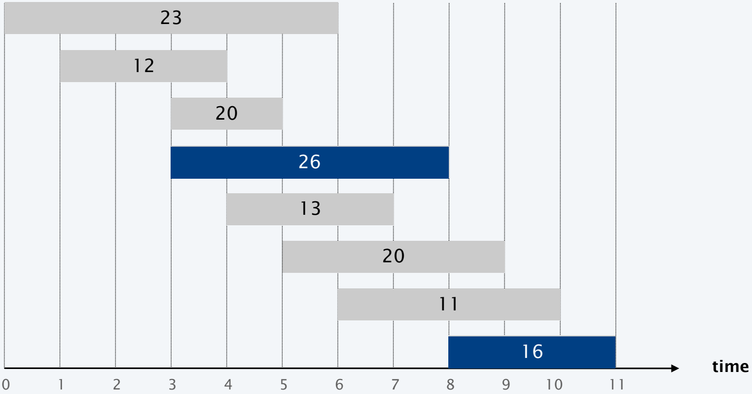
jobs don't overlap



Problem 2: Weighted interval scheduling

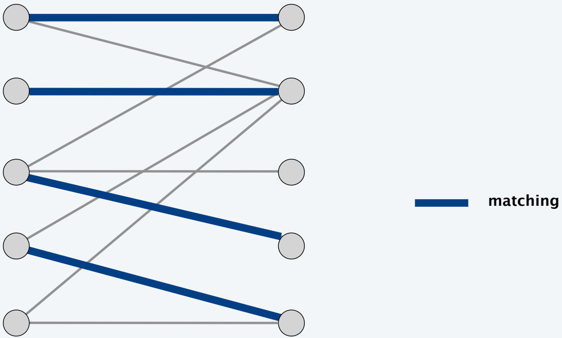
Input. Set of jobs with start times, finish times, and weights.

Goal. Find maximum weight subset of mutually compatible jobs.



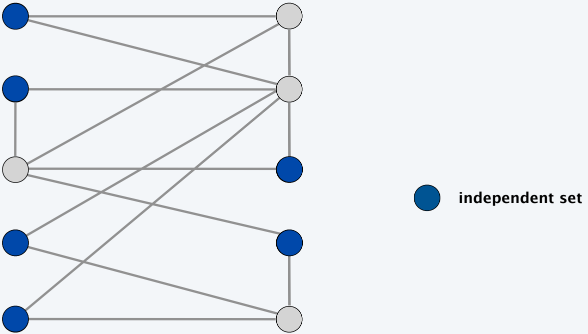
Problem. Given a bipartite graph $G = (L \cup R, E)$, find a max cardinality matching.

Def. A subset of edges $M \subseteq E$ is a matching if each node appears in exactly one edge in M.



Problem. Given a graph $G = (V, E)$, find a max cardinality independent set.

Def. A subset $S \subseteq V$ is independent if for every $(u, v) \in E$, either $u \notin S$ or $v \notin S$ (or both).

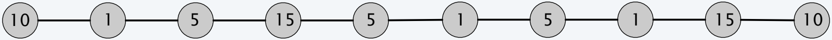


Problem 5: Competitive facility location

Input. Graph with weight on each node.

Game. Two competing players alternate in selecting nodes.
Not allowed 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

- Interval scheduling: $O(n \, log \, n)$ greedy algorithm.
- Weighted interval scheduling: $O(n \, log \, n)$ dynamic programming algorithm.
- Bipartite matching: $O(n^k)$ max-flow based algorithm.
- Independent set: **NP**-complete.
- Competitive facility location: **PSPACE**-complete.