CSCI 355: Algorithm Design and Analysis
1. Stable Matching

- *stable matching problem*
- *Gale–Shapley algorithm*
- *hospital optimality*
- *context*

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.

one student per hospital (for now)

favorite → 1st; least favorite → 3rd

	1st	2nd	3rd
Atlanta	Xavier	Yolanda	Zeus
Boston	Yolanda	Xavier	Zeus
Chicago	Xavier	Yolanda	Zeus

hospitals' preference lists

	1st	2nd	3rd
Xavier	Boston	Atlanta	Chicago
Yolanda	Atlanta	Boston	Chicago
Zeus	Atlanta	Boston	Chicago

students' preference lists

Perfect matching

Def. A matching M is a set of ordered pairs h–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 $|M| = |H| = |S| = n$.

	1st	2nd	3rd
Atlanta	Xavier	Yolanda	Zeus
Boston	Yolanda	Xavier	Zeus
Chicago	Xavier	Yolanda	Zeus

	1st	2nd	3rd
Xavier	Boston	Atlanta	Chicago
Yolanda	Atlanta	Boston	Chicago
Zeus	Atlanta	Boston	Chicago

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.

	1st	2nd	3rd
Atlanta	Xavier	Yolanda	Zeus
Boston	Yolanda	Xavier	Zeus
Chicago	Xavier	Yolanda	Zeus

	1st	2nd	3rd
Xavier	Boston	Atlanta	Chicago
Yolanda	Atlanta	Boston	Chicago
Zeus	Atlanta	Boston	Chicago

A–Y is an unstable pair for matching M = { A–Z, B–Y, C–X }

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).

	1st	2nd	3rd
Atlanta	Xavier	Yolanda	Zeus
Boston	Yolanda	Xavier	Zeus
Chicago	Xavier	Yolanda	Zeus

	1st	2nd	3rd
Xavier	Boston	Atlanta	Chicago
Yolanda	Atlanta	Boston	Chicago
Zeus	Atlanta	Boston	Chicago

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.

	1st	2nd	3rd
A	B	C	D
B	C	A	D
C	A	B	D
D	A	B	C

no perfect matching is stable

A–B, *C–D* ⇒ *B–C* unstable
A–C, *B–D* ⇒ *A–B* unstable
A–D, *B–C* ⇒ *A–C* unstable

Observation. Stable matchings need not exist.

CSCI 355: Algorithm Design and Analysis
1. Stable Matching

Gale–Shapley deferred acceptance algorithm

see 1A

An intuitive method that guarantees to find a stable matching.

GALE–SHAPLEY (*preference lists for hospitals and students*)

```
INITIALIZE M to empty matching.
WHILE (some hospital h is unmatched and hasn't proposed to every student)
   s ← first student on h's list to whom h has not yet proposed.
   IF (s is unmatched)
      Add h–s to matching M.
   ELSE IF (s prefers h to current partner h′)
      Replace h′–s with h–s in matching M.
   ELSE
      s rejects h.
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. Thus, there are at most n^2 possible proposals. ▪

	1st	2nd	3rd	4th	5th
A	V	W	X	Y	Z
B	W	X	Y	V	Z
C	X	Y	V	W	Z
D	Y	V	W	X	Z
E	V	W	X	Y	Z

	1st	2nd	3rd	4th	5th
V	B	C	D	E	A
W	C	D	E	A	B
X	D	E	A	B	C
Y	E	A	B	C	D
Z	A	B	C	D	E

n(n–1) + 1 proposals

Proof of correctness: perfect matching

Claim. Gale–Shapley outputs a matching.

Pf.

- Hospital proposes only if unmatched. ⇒ matched to ≤ 1 student
- Student keeps only best hospital. ⇒ matched to ≤ 1 hospital ▪

Claim. In Gale–Shapley matching, all hospitals get matched.

Pf. [by contradiction]

- Suppose, for sake of contradiction, that some hospital $h \in H$ is unmatched upon termination of Gale–Shapley algorithm.
- Then some student, say $s \in S$, is unmatched 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 counting]

- By previous claim, all n hospitals get matched.
- Thus, all n students get matched. ▪

Proof of correctness: stability

Claim. In Gale–Shapley matching M^*, there are no unstable pairs.

Pf. Consider any pair h–s that is not in M^*.

- Case 1: h never proposed to s.
  - ⇒ h prefers its Gale–Shapley partner s' to s. ← hospitals propose in decreasing order of preference
  - ⇒ h–s is not unstable.
- Case 2: h proposed to s.
  - ⇒ s rejected h (either right away or later)
  - ⇒ s prefers Gale–Shapley partner h' to h. ← students only trade up
  - ⇒ h–s is not unstable.
- In either case, the pair h–s is not unstable. ▪

$h - s'$

$h' - s$

⋮

Gale–Shapley matching M*

Summary

Stable matching problem. Given n hospitals and n students, and their preference lists, 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.

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.

CSCI 355: ALGORITHM DESIGN AND ANALYSIS
1. STABLE MATCHING

- stable matching problem
- Gale–Shapley algorithm
- *hospital optimality*
- context

Understanding the solution

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

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

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

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 X and Y are valid partners for A.
- Both X and Y are valid partners for B.
- Z is the only valid partner for C.

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

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

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 a perfect matching?
- Is it stable?

Claim. All executions of Gale–Shapley yield hospital-optimal assignment.
Corollary. Hospital-optimal assignment is a stable matching!

Hospital optimality

Claim. Gale–Shapley matching M^* 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.
  ⇒ some hospital is rejected by a valid partner during Gale–Shapley
- Let h be first such hospital, and let s be the first valid partner 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'.
  ⇒ s prefers h' to h. ← students only trade up
- 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. ← because this is the first rejection by a valid partner
- Thus, h' had not yet proposed to s' when h' proposed to s.
  ⇒ h' prefers s to s'. ← hospitals propose in decreasing order of preference
- Thus, h'–s is unstable in M, a contradiction. ▪

$h - s$
$h' - s'$
⋮

stable matching M

Student pessimality

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

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

Claim. Gale–Shapley finds student-pessimal stable matching M^*.
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.
  ⇒ 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.
  ⇒ h prefers s to s'.
- Thus, h–s is an unstable pair in M, a contradiction. ▪

$h' - s$
$h - s'$
⋮

stable matching M

Extensions

Extension 1. Some agents declare others as unacceptable. ← med-school student unwilling to work in Cleveland
Extension 2. Some hospitals have more than one position.
Extension 3. Unequal number of positions and students. ← ≥ 43K med-school students; only 31K positions

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.

Theorem. There exists a stable matching.
Pf. Straightforward generalization of Gale–Shapley algorithm.

CSCI 355: Algorithm Design and Analysis
1. Stable Matching

- stable matching problem
- Gale–Shapley algorithm
- hospital optimality
- context

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)

hospitals began making offers earlier and earlier, up to 2 years in advance

stable matching no longer guaranteed to exist

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)

THE MATCH
NATIONAL RESIDENT MATCHING PROGRAM®

2012 Nobel Prize in Economics

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

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.

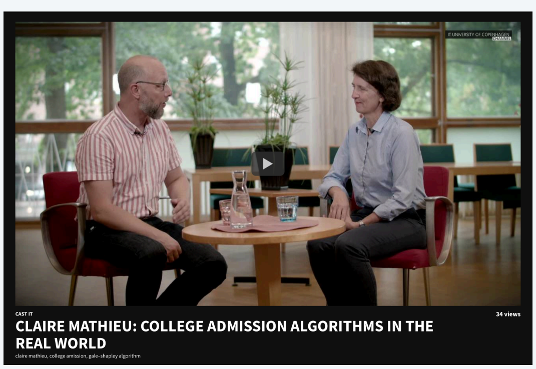
original applications: college admissions and opposite-sex marriage

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

Lloyd Shapley Alvin Roth

College admissions in France

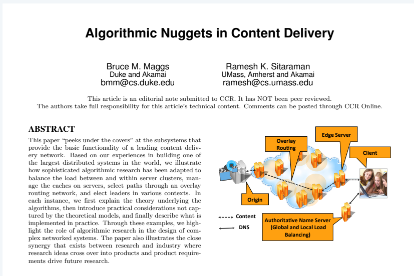
French student. Applies to 10 college programs.
French college. Ranks applicants; starts sending out offers on May 21.
Goal. Match 1M students to 10K college programs.



A modern application

Content delivery networks. Distribute much of world's content on web.

User. Preferences based on latency and packet loss.
Web server. Preferences based on costs of bandwidth and co-location.
Goal. Assign billions of users to servers, every 10 seconds.



Algorithmic Nuggets in Content Delivery

Bruce M. Maggs — Duke and Akamai — bmm@cs.duke.edu
Ramesh K. Sitaraman — UMass, Amherst and Akamai — ramesh@cs.umass.edu

This article is an editorial note submitted to CCR. It has NOT been peer reviewed.
The authors take full responsibility for this article's technical content. Comments can be posted through CCR Online.

ABSTRACT

This paper "peeks under the covers" at the subsystems that provide the basic functionality of a leading content delivery network. Based on our experiences in building one of the largest distributed systems in the world, we illustrate how sophisticated algorithmic research has been adapted to balance the load between and within server clusters, manage the caches on servers, select paths through an overlay routing network, and elect leaders in various contexts. In each instance, we first explain the theory underlying the algorithms, then introduce practical considerations not captured by the theoretical models, and finally describe what is implemented in practice. Through these examples, we highlight the role of algorithmic research in the design of complex networked systems. The paper also illustrates the close synergy that exists between research and industry where research ideas cross over into products and product requirements drive future research.

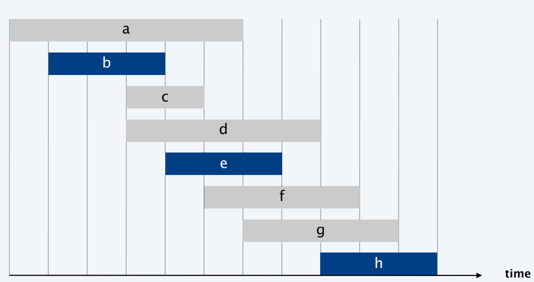
CSCI 355: Algorithm Design and Analysis
1. Representative Problems

- *five representative problems*

Interval scheduling

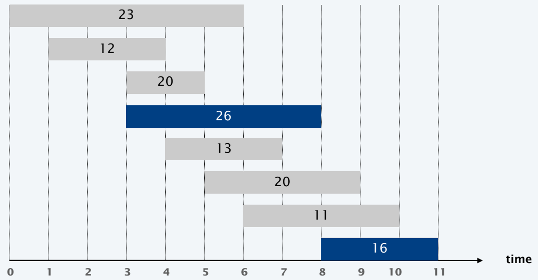
Input. Set of jobs with start times and finish times.
Goal. Find maximum cardinality subset of mutually compatible jobs.

jobs don't overlap



Weighted interval scheduling

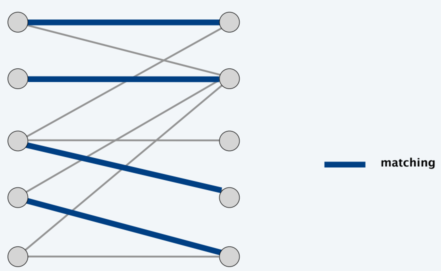
Input. Set of jobs with start times, finish times, and weights.
Goal. Find maximum weight subset of mutually compatible jobs.



Bipartite matching

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

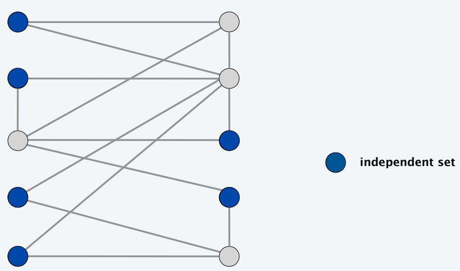
Def. A subset of edges $M \subseteq E$ is a matching if each node appears in exactly one edge in M.



Independent set

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

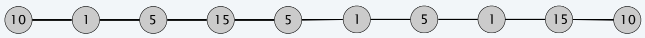
Def. A subset $S \subseteq V$ is independent if for every $(u, v) \in E$, either $u \notin S$ or $v \notin S$ (or both).



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

Variations on a theme: independent set.

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.