Using Matching with Preferences over Colleagues to Solve Classical Matching Problems

Scott Duke Kominers

Harvard University

Boston Undergraduate Research Symposium
April 11, 2009

The Problem

The Problem

College Admissions

The Problem

College Admissions

The Problem

College Admissions

- Students

The Problem

College Admissions

- Students, with preferences over colleges

The Problem

College Admissions

- Students, with preferences over colleges
- Colleges

The Problem

College Admissions

- Students, with preferences over colleges
- Colleges, with preferences over students

The Problem

College Admissions

- Students, with strict preferences over colleges
- Colleges, with strict preferences over students

The Problem

College Admissions

- Students, with strict preferences over colleges
- Colleges, with strict preferences over students

The Problem

College Admissions

- Students, with strict preferences over colleges
- Colleges, with strict preferences over students

Question

How do we match students to colleges?

The Problem

College Admissions

- Students, with strict preferences over colleges
- Colleges, with strict preferences over students

Question

How do we match students to colleges in a stable way?

The Problem

College Admissions

- Students, with strict preferences over colleges
- Colleges, with strict preferences over students

Question

How do we match students to colleges in a stable way?

The Problem

Question

How do we match students to colleges in a stable way?

The Problem

Question

How do we match students to colleges in a stable way?

What is “stability”?

The Problem

Question

How do we match students to colleges in a stable way?

What is "instability"?

The Problem

Question

How do we match students to colleges in a stable way?

What is “instability”?

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is "unstable" if...

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is “unstable” if...

- student s_1 matched to college Y

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is "unstable" if...

- student s_1 matched to college Y ($s_1 \rightarrow Y$)

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is “unstable” if...

- student s_1 matched to college Y ($s_1 \to Y$)
- student s_2 matched to college Z

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is “unstable” if...

- student s_1 matched to college Y ($s_1 \rightarrow Y$)
- student s_2 matched to college Z ($s_2 \rightarrow Z$)

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is "unstable" if...

- student s_1 matched to college Y ($s_1 \to Y$)
- student s_2 matched to college Z ($s_2 \to Z$)
- student s_1 prefers college Z to college Y

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is "unstable" if...

- student s_1 matched to college Y ($s_1 \to Y$)
- student s_2 matched to college Z ($s_2 \to Z$)
- student s_1 prefers college Z to college Y ($Z \succ_{s_1} Y$)

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is "unstable" if...

- student s_1 matched to college Y ($s_1 \rightarrow Y$)
- student s_2 matched to college Z ($s_2 \rightarrow Z$)
- student s_1 prefers college Z to college Y ($Z \succ_{s_1} Y$)
- college Z prefers student s_1 to student s_2

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is "unstable" if...

- student s_1 matched to college Y ($s_1 \rightarrow Y$)
- student s_2 matched to college Z ($s_2 \rightarrow Z$)
- student s_1 prefers college Z to college Y ($Z \succ_{s_1} Y$)
- college Z prefers student s_1 to student s_2 ($s_1 \succ_Z s_2$)

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is “unstable” if...

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is “unstable” if there exist students s_1, s_2 and colleges Y, Z such that

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is "unstable" if there exist students s_1, s_2 and colleges Y, Z such that

$$s_1 \longrightarrow Y, \quad s_2 \longrightarrow Z, \quad Z \succ_{s_1} Y, \quad s_1 \succ_Z s_2.$$

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is “unstable” if there exist students s_1, s_2 and colleges Y, Z such that

$$s_1 \to Y, \quad s_2 \to Z, \quad Z \succ_{s_1} Y, \quad s_1 \succ_Z s_2.$$

Why is instability bad?

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is “unstable” if there exist students s_1, s_2 and colleges Y, Z such that

$$s_1 \longrightarrow Y, \quad s_2 \longrightarrow Z, \quad Z \succ_{s_1} Y, \quad s_1 \succ_Z s_2.$$

Why is instability bad?

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is “unstable” if there exist students s_1, s_2 and colleges Y, Z such that

$$s_1 \rightarrow Y, \quad s_2 \rightarrow Z, \quad Z \succ_{s_1} Y, \quad s_1 \succ_Z s_2.$$

Why is instability bad?

Good news:

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is “unstable” if there exist students s_1, s_2 and colleges Y, Z such that

$$s_1 \longrightarrow Y, \quad s_2 \longrightarrow Z, \quad Z \succ_{s_1} Y, \quad s_1 \succ_Z s_2.$$

Why is instability bad?

Good news: a stable matching always exists.

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is "unstable" if there exist students s_1, s_2 and colleges Y, Z such that

$$s_1 \longrightarrow Y, \quad s_2 \longrightarrow Z, \quad Z \succ_{s_1} Y, \quad s_1 \succ_Z s_2.$$

Why is instability bad?

Good news: a stable matching always exists.

The Problem

Question

How do we match students to colleges in a stable way?

An matching of students to colleges is "unstable" if there exist students s_1, s_2 and colleges Y, Z such that

$$s_1 \longrightarrow Y, \quad s_2 \longrightarrow Z, \quad Z \succ_{s_1} Y, \quad s_1 \succ_Z s_2.$$

Why is instability bad?

Good news: a stable matching always exists.[1]

[1] Gale–Shapley (1962)

An Example

An Example

- One college: Z

An Example

- One college: Z
- Three students: s_1, s_2, s_3

An Example

- One college: Z
- Three students: s_1, s_2, s_3 — want to go to college

An Example

- One college: Z
- Three students: s_1, s_2, s_3 — want to go to college

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \rightarrow Z$, $s_2, s_3 \rightarrow \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \rightarrow Z$, $s_2, s_3 \rightarrow \emptyset$ — unstable

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \to Z$, $s_2, s_3 \to \emptyset$ — unstable

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \to Z,\ s_2, s_3 \to \emptyset$ — unstable
  - $s_1 \to Z,\ s_2, s_3 \to \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \rightarrow Z,\ s_2, s_3 \rightarrow \emptyset$ — unstable
  - $s_1 \rightarrow Z,\ s_2, s_3 \rightarrow \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \to Z,\ s_2, s_3 \to \emptyset$ — unstable
  - $s_1, s_2 \to Z,\ s_3 \to \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \rightarrow Z$, $s_2, s_3 \rightarrow \emptyset$ — unstable
  - $s_1, s_2 \rightarrow Z$, $s_3 \rightarrow \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \rightarrow Z$, $s_2, s_3 \rightarrow \emptyset$ — unstable
  - $s_1, s_2 \rightarrow Z$, $s_3 \rightarrow \emptyset$ — stable

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \rightarrow Z,\ s_2, s_3 \rightarrow \emptyset$ — unstable
  - $s_1, s_2 \rightarrow Z,\ s_3 \rightarrow \emptyset$ — stable

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \longrightarrow Z,\ s_2, s_3 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \longrightarrow Z,\ s_3 \longrightarrow \emptyset$ — stable
- $s_2, s_3 \longrightarrow Z,\ s_1 \longrightarrow \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \to Z$, $s_2, s_3 \to \emptyset$ — unstable
  - $s_1, s_2 \to Z$, $s_3 \to \emptyset$ — stable
- $s_2, s_3 \to Z$, $s_1 \to \emptyset$ — unstable

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \rightarrow Z,\ s_2, s_3 \rightarrow \emptyset$ — unstable
    - $s_1, s_2 \rightarrow Z,\ s_3 \rightarrow \emptyset$ — stable
- $s_2, s_3 \rightarrow Z,\ s_1 \rightarrow \emptyset$ — unstable

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \longrightarrow Z,\ s_2, s_3 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \longrightarrow Z,\ s_3 \longrightarrow \emptyset$ — stable
- $s_2, s_3 \longrightarrow Z,\ s_1 \longrightarrow \emptyset$ — unstable
  - $s_2, s_3 \longrightarrow Z,\ s_1 \longrightarrow \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \longrightarrow Z,\ s_2, s_3 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \longrightarrow Z,\ s_3 \longrightarrow \emptyset$ — stable
- $s_2, s_3 \longrightarrow Z,\ s_1 \longrightarrow \emptyset$ — unstable
  - $s_2, s_3 \longrightarrow Z,\ s_1 \longrightarrow \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \longrightarrow Z,\ s_2, s_3 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \longrightarrow Z,\ s_3 \longrightarrow \emptyset$ — stable
- $s_2, s_3 \longrightarrow Z,\ s_1 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2, s_3 \longrightarrow Z,\ \longrightarrow \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \longrightarrow Z,\ s_2, s_3 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \longrightarrow Z,\ s_3 \longrightarrow \emptyset$ — stable
- $s_2, s_3 \longrightarrow Z,\ s_1 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2, s_3 \longrightarrow Z, \ \longrightarrow \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \longrightarrow Z,\ s_2, s_3 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \longrightarrow Z,\ s_3 \longrightarrow \emptyset$ — stable
- $s_2, s_3 \longrightarrow Z,\ s_1 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2, s_3 \longrightarrow Z,\ \longrightarrow \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \to Z,\ s_2, s_3 \to \emptyset$ — unstable
    - $s_1, s_2 \to Z,\ s_3 \to \emptyset$ — stable
- $s_2, s_3 \to Z,\ s_1 \to \emptyset$ — unstable
    - $s_1, s_2, s_3 \to Z,\ \to \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \longrightarrow Z,\ s_2, s_3 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \longrightarrow Z,\ s_3 \longrightarrow \emptyset$ — stable
- $s_2, s_3 \longrightarrow Z,\ s_1 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \longrightarrow Z,\ s_3 \longrightarrow \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \longrightarrow Z,\ s_2, s_3 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \longrightarrow Z,\ s_3 \longrightarrow \emptyset$ — stable
- $s_2, s_3 \longrightarrow Z,\ s_1 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \longrightarrow Z,\ s_3 \longrightarrow \emptyset$

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \longrightarrow Z$, $s_2, s_3 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \rightarrow Z$, $s_3 \rightarrow \emptyset$ — stable
- $s_2, s_3 \longrightarrow Z$, $s_1 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \rightarrow Z$, $s_3 \rightarrow \emptyset$ — stable

An Example

- One college: Z — can only admit two students
- Three students: s_1, s_2, s_3 — want to go to college
- $s_1 \succ_Z s_2 \succ_Z s_3$

Possible Matchings

- $s_1 \longrightarrow Z,\ s_2, s_3 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \longrightarrow Z,\ s_3 \longrightarrow \emptyset$ — stable
- $s_2, s_3 \longrightarrow Z,\ s_1 \longrightarrow \emptyset$ — unstable
  - $s_1, s_2 \longrightarrow Z,\ s_3 \longrightarrow \emptyset$ — stable

Real-world Applications

Real-world Applications

Matching of...

Real-world Applications

Matching of...

- students to schools

Real-world Applications

Matching of...

- students to schools (in Boston and New York)

Real-world Applications

Matching of...

- students to schools (in Boston and New York)
- (medical) students to residencies

Real-world Applications

Matching of...

- students to schools (in Boston and New York)
- (medical) students to residencies
- students to sororities

Real-world Applications

Matching of...

- students to schools (in Boston and New York)
- (medical) students to residencies
- students to sororities

However...

Real-world Applications

Matching of...

- students to schools (in Boston and New York)
- (medical) students to residencies
- students to sororities

However...

- no direct application to college admissions

Real-world Applications

Matching of...

- students to schools (in Boston and New York)
- (medical) students to residencies
- students to sororities

However...

- no direct application to college admissions (yet)

Pause

Pause

- We just described “classical matching”.

Pause

- We just described "classical matching".
- Recall the title slide....

Pause

Using Matching with Preferences over Colleagues to Solve Classical Matching Problems

Scott Duke Kominers

Harvard University

Boston Undergraduate Research Symposium
April 11, 2009

Pause

- We just described "classical matching".
- Recall the title slide....

Pause

- We just described “classical matching”.
- Recall the title slide....

Natural Question

Pause

- We just described "classical matching".
- Recall the title slide....

Natural Question

What is "matching with preferences over colleagues"?

The Problem

College Admissions

- Students, with strict preferences over colleges
- Colleges, with strict preferences over students

Question

How do we match students to colleges in a stable way?

The New Problem

College Admissions

- Students, with strict preferences over colleges
- Colleges, with strict preferences over students

Question

How do we match students to colleges in a stable way?

The New Problem

College Admissions

- Students, with strict preferences over colleges and over their possible sets of classmates
- Colleges, with strict preferences over students

Question

How do we match students to colleges in a stable way?

The New Problem

College Admissions

- Students, with strict preferences over colleges and over their possible sets of classmates
- Colleges, with strict preferences over students

Question

How do we match students to colleges in a stable way?

The New Problem

College Admissions

- Students, with strict preferences over colleges and over their possible sets of classmates
- Colleges, with strict preferences over students

Question — Solved

How do we match students to colleges in a stable way?

The New Problem

College Admissions

- Students, with strict preferences over colleges and over their possible sets of classmates
- Colleges, with strict preferences over students

Question — Solved, with an Algorithm

How do we match students to colleges in a stable way?

The New Problem

College Admissions

- Students, with strict preferences over colleges and over their possible sets of classmates
- Colleges, with strict preferences over students

Question — Solved, with an Algorithm

How do we match students to colleges in a stable way?[a]

[a] Echenique–Yenmez (2007)

The New Problem

College Admissions

- Students, with strict preferences over colleges and over their possible sets of classmates
- Colleges, with strict preferences over students

Question — Solved, with an Algorithm

How do we match students to colleges in a stable way?

The New Problem

College Admissions

- Students, with strict preferences over colleges and over their possible sets of classmates
- Colleges, with strict preferences over students

Question — Solved, with an Algorithm

How do we match students to colleges in a stable way?

Question

Can we use this algorithm to solve classical matching?

The New Problem

College Admissions

- Students, with strict preferences over colleges and over their possible sets of classmates
- Colleges, with strict preferences over students

Question — Solved, with an Algorithm

How do we match students to colleges in a stable way?

Nontrivial Question

Can we use this algorithm to solve classical matching?

The New Problem

College Admissions

- Students, with strict preferences over colleges and over their possible sets of classmates
- Colleges, with strict preferences over students

Question — Solved, with an Algorithm

How do we match students to colleges in a stable way?

Nontrivial Question

Can we use this algorithm to solve classical matching?

The New Problem

College Admissions

- Students, with strict preferences over colleges and over their possible sets of classmates
- Colleges, with strict preferences over students

Question — Solved, with an Algorithm

How do we match students to colleges in a stable way?

Nontrivial Question

Can we use this algorithm to solve classical matching?

The Solution

Nontrivial Question

Can we use this algorithm to solve classical matching?

The Solution

Nontrivial Question

Can we use this algorithm to solve classical matching?

- Yes, we can

The Solution

Nontrivial Question

Can we use this algorithm to solve classical matching?

- Yes, we can...

The Solution

Nontrivial Question

Can we use this algorithm to solve classical matching?

- Yes, we can...
  - with an elementary construction...

The Solution

Nontrivial Question

Can we use this algorithm to solve classical matching?

- Yes, we can...
  - with an elementary construction...
  - but at a complexity cost.

The Solution

Nontrivial Question

Can we use this algorithm to solve classical matching?

- Yes, we can!

The Solution

Nontrivial Question

Can we use this algorithm to solve classical matching?

- Yes, we can!

Theorem

The Solution

Nontrivial Question

Can we use this algorithm to solve classical matching?

- Yes, we can!

Theorem

For any "classical matching" problem

The Solution

Nontrivial Question

Can we use this algorithm to solve classical matching?

- Yes, we can!

Theorem

For any "classical matching" problem, there is an associated "matching with preferences over colleagues" problem

The Solution

Nontrivial Question

Can we use this algorithm to solve classical matching?

- Yes, we can!

Theorem

For any "classical matching" problem, there is an associated "matching with preferences over colleagues" problem with stable matchings directly corresponding to the stable matchings of the original classical problem.

The Solution

Nontrivial Question

Can we use this algorithm to solve classical matching?

- Yes, we can!

Corollary

Given any classical matching problem, we can find **all** *stable matchings.*

The Solution

Nontrivial Question

Can we use this algorithm to solve classical matching?

- Yes, we can!

Key Idea

Align student and college preferences!

Acknowledgments

Acknowledgments

- Professors Alvin E. Roth and Peter A. Coles

Acknowledgments

- Professors Alvin E. Roth and Peter A. Coles
- Zachary Abel, John Hatfield, Noam D. Elkies, Drew Fudenberg, Bettina Klaus, and Fuhito Kojima

Acknowledgments

- Professors Alvin E. Roth and Peter A. Coles
- Zachary Abel, John Hatfield, Noam D. Elkies, Drew Fudenberg, Bettina Klaus, and Fuhito Kojima
- Harvard College PRISE and Harvard Department of Economics

Acknowledgments

- Professors Alvin E. Roth and Peter A. Coles
- Zachary Abel, John Hatfield, Noam D. Elkies, Drew Fudenberg, Bettina Klaus, and Fuhito Kojima
- Harvard College PRISE and Harvard Department of Economics
- BURS

Acknowledgments

- Professors Alvin E. Roth and Peter A. Coles
- Zachary Abel, John Hatfield, Noam D. Elkies, Drew Fudenberg, Bettina Klaus, and Fuhito Kojima
- Harvard College PRISE and Harvard Department of Economics
- BURS organizers

Acknowledgments

- Professors Alvin E. Roth and Peter A. Coles
- Zachary Abel, John Hatfield, Noam D. Elkies, Drew Fudenberg, Bettina Klaus, and Fuhito Kojima
- Harvard College PRISE and Harvard Department of Economics
- BURS organizers and audience

Acknowledgments

- Professors Alvin E. Roth and Peter A. Coles
- Zachary Abel, John Hatfield, Noam D. Elkies, Drew Fudenberg, Bettina Klaus, and Fuhito Kojima
- Harvard College PRISE and Harvard Department of Economics
- BURS organizers and audience ($\mathbb{QED}$)

Questions?

Extra Slides

The Construction

The Construction

- One College: Z

The Construction

- One College: Z
- Two students: s_1, s_2

The Construction

- One College: Z
- Two students: s_1, s_2
- Classical preference profiles $\succ$

The Construction

- One College: Z
- Two students: s_1, s_2
- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$

The Construction

- One College: Z
- Two students: s_1, s_2
- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_1} \emptyset$

The Construction

- One College: Z
- Two students: s_1, s_2
- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_1} \emptyset$
  - $Z \succ_{s_2} \emptyset$

The Construction

- One College: Z
- Two students: s_1, s_2

- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_i} \emptyset$

The Construction

- One College: Z
- Two students: s_1, s_2

- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_i} \emptyset \; (i = 1, 2)$

The Construction

- One College: Z
- Two students: s_1, s_2
- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_i} \emptyset \ (i = 1, 2)$
- Nonclassical preference profiles $\triangleright$

The Construction

- One College: Z
- Two students: s_1, s_2
- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_i} \emptyset$ $(i = 1, 2)$
- Nonclassical preference profiles $\rhd$
  - $\{s_1, s_2\} \rhd_Z \{s_1\} \rhd_Z \emptyset$

The Construction

- One College: Z
- Two students: s_1, s_2
- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_i} \emptyset \ (i = 1, 2)$
- Nonclassical preference profiles $\triangleright$
  - $\{s_1, s_2\} \triangleright_Z \{s_1\} \triangleright_Z \emptyset$
  - $Z \triangleright_{s_1} \emptyset$

The Construction

- One College: Z
- Two students: s_1, s_2

- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_i} \emptyset \ (i = 1, 2)$

- Nonclassical preference profiles $\triangleright$
  - $\{s_1, s_2\} \triangleright_Z \{s_1\} \triangleright_Z \emptyset$
  - $(Z,) \triangleright_{s_1} (\emptyset,)$

The Construction

- One College: Z
- Two students: s_1, s_2
- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_i} \emptyset \; (i = 1, 2)$
- Nonclassical preference profiles $\triangleright$
  - $\{s_1, s_2\} \triangleright_Z \{s_1\} \triangleright_Z \emptyset$
  - $(Z,) \triangleright_{s_1} (Z,) \triangleright_{s_1} (\emptyset,)$

The Construction

- One College: Z
- Two students: s_1, s_2
- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_i} \emptyset$ $(i = 1, 2)$
- Nonclassical preference profiles $\triangleright$
  - $\{s_1, s_2\} \triangleright_Z \{s_1\} \triangleright_Z \emptyset$
  - $(Z, \{s_1, s_2\}) \triangleright_{s_1} (Z, \{s_1\}) \triangleright_{s_1} (\emptyset, \emptyset)$

The Construction

- One College: Z
- Two students: s_1, s_2

- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_i} \emptyset \; (i = 1, 2)$

- Nonclassical preference profiles $\triangleright$
  - $\{s_1, s_2\} \triangleright_Z \{s_1\} \triangleright_Z \emptyset$
  - $(Z, \{s_1, s_2\}) \triangleright_{s_1} (Z, \{s_1\}) \triangleright_{s_1} (\emptyset, \emptyset)$
  - $Z \triangleright_{s_2} \emptyset$

The Construction

- One College: Z
- Two students: s_1, s_2

- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_i} \emptyset \ (i = 1, 2)$

- Nonclassical preference profiles $\triangleright$
  - $\{s_1, s_2\} \triangleright_Z \{s_1\} \triangleright_Z \emptyset$
  - $(Z, \{s_1, s_2\}) \triangleright_{s_1} (Z, \{s_1\}) \triangleright_{s_1} (\emptyset, \emptyset)$
  - $(Z,) \triangleright_{s_2} (\emptyset,)$

The Construction

- One College: Z
- Two students: s_1, s_2

- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_i} \emptyset \ (i = 1, 2)$

- Nonclassical preference profiles $\rhd$
  - $\{s_1, s_2\} \rhd_Z \{s_1\} \rhd_Z \emptyset$
  - $(Z, \{s_1, s_2\}) \rhd_{s_1} (Z, \{s_1\}) \rhd_{s_1} (\emptyset, \emptyset)$
  - $(Z, \{s_1, s_2\}) \rhd_{s_2} (\emptyset, \emptyset)$

The Construction

- One College: Z
- Two students: s_1, s_2

- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_i} \emptyset \ (i = 1, 2)$

- Nonclassical preference profiles $\triangleright$
  - $\{s_1, s_2\} \triangleright_Z \{s_1\} \triangleright_Z \emptyset$
  - $(Z, \{s_1, s_2\}) \triangleright_{s_1} (Z, \{s_1\}) \triangleright_{s_1} (\emptyset, \emptyset)$
  - $(Z, \{s_1, s_2\}) \triangleright_{s_2} (\emptyset, \emptyset)$

The Construction

- One College: Z
- Two students: s_1, s_2

- Classical preference profiles $\succ$
  - $\{s_1, s_2\} \succ_Z \{s_1\} \succ_Z \emptyset$
  - $Z \succ_{s_i} \emptyset \ (i = 1, 2)$

- Nonclassical preference profiles $\triangleright$
  - $\{s_1, s_2\} \triangleright_Z \{s_1\} \triangleright_Z \emptyset$
  - $(Z, \{s_1, s_2\}) \triangleright_{s_1} (Z, \{s_1\}) \triangleright_{s_1} (\emptyset, \emptyset)$
  - $(Z, \{s_1, s_2\}) \triangleright_{s_2} (\emptyset, \emptyset)$

Complexity Analysis

Complexity Analysis

- $|\mathcal{P}| :=$ size of largest preference relation in $\mathcal{P}$

Complexity Analysis

- $|\mathcal{P}| :=$ size of largest preference relation in $\mathcal{P}$
- $|\mathcal{P}_{GS}| \sim$ baseline

Complexity Analysis

- $|\mathcal{P}| :=$ size of largest preference relation in $\mathcal{P}$
- $|\mathcal{P}_{GS}| \sim$ baseline
- $|\mathcal{P}_{EY}(\mathcal{P}_{GS})| = O(|\mathcal{P}_{GS}|^2)$

Complexity Analysis

- $|\mathcal{P}| :=$ size of largest preference relation in $\mathcal{P}$
- $|\mathcal{P}_{GS}| \sim$ baseline
- $|\mathcal{P}_{EY}(\mathcal{P}_{GS})| = O(|\mathcal{P}_{GS}|^2)$
    - $\sim$ *input size* of our algorithm

Complexity Analysis

- $|\mathcal{P}| :=$ size of largest preference relation in $\mathcal{P}$
- $|\mathcal{P}_{GS}| \sim$ baseline
- $|\mathcal{P}_{EY}(\mathcal{P}_{GS})| = O(|\mathcal{P}_{GS}|^2)$
  - $\sim$ *input size* of our algorithm
  - $\sim$ *running time* of the deferred acceptance algorithm

Complexity Analysis

- $|\mathcal{P}| :=$ size of largest preference relation in $\mathcal{P}$

- $|\mathcal{P}_{GS}| \sim$ baseline
- $|\mathcal{P}_{EY}(\mathcal{P}_{GS})| = O(|\mathcal{P}_{GS}|^2)$
  - $\sim$ *input size* of our algorithm
  - $\sim$ *running time* of the deferred acceptance algorithm

Question

Can we do better?

Complexity Analysis

- $|\mathcal{P}| :=$ size of largest preference relation in $\mathcal{P}$

- $|\mathcal{P}_{GS}| \sim$ baseline
- $|\mathcal{P}_{EY}(\mathcal{P}_{GS})| = O(|\mathcal{P}_{GS}|^2)$
  - $\sim$ *input size* of our algorithm
  - $\sim$ *running time* of the deferred acceptance algorithm

Open Question

Can we do better?