Computational Social Choice

Kidney Exchange

Piotr Skowron
University of Warsaw
An example

patient  donor
An example

patient

group A

donor

group B

\[\text{group A } \rightarrow \text{patient } \rightarrow \text{group B} \]
An example

- **Group A:**
  - Patient
  - Donor (impossible due to blood type
- **Group B:**
  - Patient
  - Donor (impossible due to blood type
An example
An example

- Group A
- Group B

Patient

Donor

Group A

Group B

X

Y
An example
An example
An example
An example
The model

Input:

A set of pairs (donor, recipient) \( C = \{p_1, p_2, \ldots, p_m\} \).
Each pair \( p \) comes with a preference relation, \( >_p \).

Output:

A decomposition into cycles.
The top-trading cycle

In a loop:
1. Each pair points to her most preferred pair.
2. If there is a cycle, we add it to the solution, and remove the pairs from the cycle.
3. If there is no cycle, we stop.
The top-trading cycle
The top-trading cycle

\[ B > E > G \]

\[ E > F > C \]

\[ G > C > A \]

\[ A > E > H \]

\[ G > F > I \]

\[ F > I > C \]

\[ H > C > B \]

\[ D > F \]

\[ E > F > D \]

\[ I \]
The top-trading cycle

1. $B > E > G$
2. $E > F > C$
3. $G > C > A$
4. $A > E > H$
5. $G > F > I$
6. $D > F$
7. $F > I > C$
8. $H > C > B$
9. $E > F > D$
10. $I$

Arrow directions indicate the sequence of trades or preferences.
The top-trading cycle
The top-trading cycle
The top-trading cycle

- $D > F$
- $H > C > B$
- $G > F > I$
- $E > F > D$
- $F > I > C$
- $H > C > B$
- $D$
- $I$
The top-trading cycle

\[ D > F \]
\[ C \]
\[ F > I > C \]
\[ D \]
\[ E > F > D \]
\[ I \]
The top-trading cycle

D > F
C

F > I > C
D

E > F > D
I
The top-trading cycle

Diagram showing the top-trading cycle with the order of preferences:

1. $D > F$
2. $F > I > C$
3. $E > F > D$
4. $D > F$

The cycle is shown with arrows indicating the sequence of trades, where each participant prefers the next in the cycle over themselves.
The top-trading cycle

\[ D \succ F \]

C
The top-trading cycle
The top-trading cycle
The top-trading cycle

In a loop:
1. Each pair points to her most preferred pair.
2. If there is a cycle, we add it to the solution, and remove the pairs from the cycle.
3. If there is no cycle, we stop.

Theorem:
An optimal strategy for each pair is to reveal their true preference relation.
The top-trading cycle

In a loop:
1. Each pair points to her most preferred pair.
2. If there is a cycle, we add it to the solution, and remove the pairs from the cycle.
3. If there is no cycle, we stop.

Theorem:
An optimal strategy for each pair is to reveal their true preference relation.

Proof:
Consider a pair $p$ that is matched with $p'$. Assume that $p$ prefers $p''$ to $p'$. We will show that $p$ cannot be matched with $p''$. 
The top-trading cycle

In a loop:
1. Each pair points to her most preferred pair.
2. If there is a cycle, we add it to the solution, and remove the pairs from the cycle.
3. If there is no cycle, we stop.

Theorem:
An optimal strategy for each pair is to reveal their true preference relation.

Proof:
Consider a pair $p$ that is matched with $p'$. Assume that $p$ prefers $p''$ to $p'$.
We will show that $p$ cannot be matched with $p''$.
1. Consider all time moments until $p''$ was removed; $p$ could not have been removed before, since it was pointing to $p''$ or someone better, and got $p'$. 
The top-trading cycle

In a loop:
1. Each pair points to her most preferred pair.
2. If there is a cycle, we add it to the solution, and remove the pairs from the cycle.
3. If there is no cycle, we stop.

Theorem:
An optimal strategy for each pair is to reveal their true preference relation.

Proof:
Consider a pair $p$ that is matched with $p'$. Assume that $p$ prefers $p''$ to $p'$. We will show that $p$ cannot be matched with $p''$.

1. Consider all time moments until $p''$ was removed; $p$ could not have been removed before, since it was pointing to $p''$ or someone better, and got $p'$.
2. Each voter that was removed could not point to $p$ independently of her preferences (since, otherwise, it still would point to $p$ when $p$ was selected).
The top-trading cycle

In a loop:
1. Each pair points to her most preferred pair.
2. If there is a cycle, we add it to the solution, and remove the pairs from the cycle.
3. If there is no cycle, we stop.

Theorem:
An optimal strategy for each pair is to reveal their true preference relation.

Proof:
Consider a pair $p$ that is matched with $p'$. Assume that $p$ prefers $p''$ to $p'$.
We will show that $p$ cannot be matched with $p''$.
1. Consider all time moments until $p''$ was removed; $p$ could not have been removed before, since it was pointing to $p''$ or someone better, and got $p'$.
2. Each voter that was removed could not point to $p$ independently of her preferences (since, otherwise, it still would point to $p$ when $p$ was selected).
3. In such time moments, no voter would point to $p$, and thus, they would be removed (including $p''$) independently of what $p$ reports.
The top-trading cycle

In a loop:
1. Each pair points to her most preferred pair.
2. If there is a cycle, we add it to the solution, and remove the pairs from the cycle.
3. If there is no cycle, we stop.

**Theorem:**
An optimal strategy for each pair is to reveal their true preference relation.

**Definition:**
A solution is in the core, if there exists no group of pairs $S$ that could perform trading on their own in a way that each member of $S$ would get at least as good matched partner as in the solution, and at least one pair from $S$ would get a strictly better one.
The top-trading cycle

In a loop:
1. Each pair points to her most preferred pair.
2. If there is a cycle, we add it to the solution, and remove the pairs from the cycle.
3. If there is no cycle, we stop.

Theorem:
An optimal strategy for each pair is to reveal their true preference relation.

Definition:
A solution is in the core, if there exists no group of pairs $S$ that could perform trading on their own in a way that each member of $S$ would get at least as good matched partner as in the solution, and at least one pair from $S$ would get a strictly better one.

Theorem:
Solutions returned by the top-trading cycle are in the core.
The top-trading cycle

Theorem:
Solutions returned by the top-trading cycle are in the core.

Proof sketch:
Consider a minimal set $S$ that witnesses the violation of the core.
The top-trading cycle

**Theorem:**
Solutions returned by the top-trading cycle are in the core.

**Proof sketch:**
Consider a minimal set $S$ that witnesses the violation of the core.

Take the voter who is in the first cycle.
The top-trading cycle

Theorem:
Solutions returned by the top-trading cycle are in the core.

Proof sketch:
Consider a minimal set $S$ that witnesses the violation of the core.

Take the voter who is in the first cycle. Since she prefers $B$ to $A$, $B$ must have been eliminated before $A$ (otherwise she would point to $B$).
**Theorem:**
Solutions returned by the top-trading cycle are in the core.

**Proof sketch:**
Consider a minimal set $S$ that witnesses the violation of the core.

Take the voter who is in the first cycle. Since she prefers $B$ to $A$, $B$ must have been eliminated before $A$ (otherwise she would point to $B$). This is a contradiction since $A$ was selected at the same time as the voter that we consider.
Practice vs Theory

Problem:
Clearing a cycle of size \( \ell \) requires \( 2\ell \) operating theatres and \( 2\ell \) surgical teams available at the same time.
Problem:
Clearing a cycle of size $\ell$ requires $2\ell$ operating theatres and $2\ell$ surgical teams available at the same time.

Solution:
Use only small cycles.
Problem:
Clearing a cycle of size $\ell$ requires $2\ell$ operating theatres and $2\ell$ surgical teams available at the same time.

Solution:
Use only small cycles.

Matching Theory!
Back to the example
Back to the example
Back to the example
Back to the example
Back to the example
Problem:
Clearing a cycle of size $\ell$ requires $2\ell$ operating theatres and $2\ell$ surgical teams available at the same time.

Solution:
Use only small cycles.

Matching Theory!

Properties:
Maximum-cardinality matchings can be found in polynomial time. How to make it incentive compatible? (Using \textit{max-weight-matching}.)

44
Next step: cycles of size at most 3
Next step: cycles of size at most 3
Next step: cycles of size at most 3
Next step: cycles of size at most 3
Cycles of size at most 3: algorithm
Cycles of size at most 3: algorithm

**Theorem:**
The problem of finding a cover with cycles of size at most 3 that covers the maximal number of pairs is NP-hard.
Cycles of size at most 3: algorithm

**Theorem:**
The problem of finding a cover with cycles of size at most 3 that covers the maximal number of pairs is NP-hard.

\( \mathcal{C} \): the set of all cycles of size at most 3.
for \( c \in \mathcal{C} \) we have binary variable \( x_c \)

maximize: \( \sum_{c \in \mathcal{C}} w_c \cdot x_c \)

subject to: \( \sum_{c: e \in c} x_c \leq 1 \) for each edge \( e \).
Cyclics of size at most 3: algorithm

**Theorem:**
The problem of finding a cover with cycles of size at most 3 that covers the maximal number of pairs is NP-hard.

\[\mathcal{C}: \text{the set of all cycles of size at most 3.}\]

for \( c \in \mathcal{C} \) we have binary variable \( x_c \)

maximize:\[\sum_{c \in \mathcal{C}} w_c \cdot x_c\]

subject to:\[\sum_{c: e \in c} x_c \leq 1 \text{ for each edge } e.\]

This is not sufficiently efficient.
Cycles of size at most 3: algorithm

**Theorem:**
The problem of finding a cover with cycles of size at most 3 that covers the maximal number of pairs is NP-hard.

\[ \mathcal{C} : \text{the set of all cycles of size at most 3.} \]
for \( c \in \mathcal{C} \) we have binary variable \( x_c \)

maximize: \[ \sum_{c \in \mathcal{C}} w_c \cdot x_c \]
subject to: \[ \sum_{c : e \in c} x_c \leq 1 \text{ for each edge } e. \]

This is not sufficiently efficient.

**Heuristic algorithms using LP:**
Further complication: altruistic donors
Further complication: altruistic donors
Further complication: altruistic donors
Further complication: altruistic donors

Source: New York Times

Problem:
Hospitals typically report only the pairs that are hard to match.
Problem:
Hospitals typically report only the pairs that are hard to match.
Problem:
Hospitals typically report only the pairs that are hard to match.
Problem:
Hospitals typically report only the pairs that are hard to match.
Strategic hospitals: non-optimality

Problem:
Hospitals typically report only the pairs that are hard to match.
Strategic hospitals: non-optimality

Problem:
Hospitals typically report only the pairs that are hard to match.
Problem:
Hospitals typically report only the pairs that are hard to match.
Strategic hospitals: non-optimality

Problem:
Hospitals typically report only the pairs that are hard to match.
Problem: Hospitals typically report only the pairs that are hard to match.
Problem:
Hospitals typically report only the pairs that are hard to match.
Strategic hospitals: incentives

Problem:
Hospitals typically report only the pairs that are hard to match.

Hospital 1

- A
- B
- C
- G

3/4

Hospital 2

- D
- E
- F

3/3
Problem:
Hospitals typically report only the pairs that are hard to match.
Problem: Hospitals typically report only the pairs that are hard to match.
Problem: Hospitals typically report only the pairs that are hard to match.
Strategic hospitals: incentives

Problem:
Hospitals typically report only the pairs that are hard to match.
Problem:
Hospitals typically report only the pairs that are hard to match.

This is a big challenge to design mechanisms, where the hospitals will have incentive to report all pairs.
Problem:
Hospitals typically report only the pairs that are hard to match.

This is a big challenge to design mechanisms, where the hospitals will have incentive to report all pairs.

Mechanisms based on credits:
Summary

1. Top-trading cycle. (Econ Theory!)
2. Covering with matchings. (Graph Theory and CS!)
3. Covering with cycles of size at most 3. (AI!)
4. Strategic hospitals. (Mechanism Design!)