House Markets and Single-Peaked Preferences: From Centralized to Decentralized Procedures

Aurélie Beynier, Nicolas Maudet, Simon Rey and Parham Shams

July 9, 2019

Allocation of indivisible resources

The problem is to allocate a set of goods to a set of agents. Two main difficulties:

- Agents have preference over the goods that should be taken into account to find a "fair allocation"
- Goods are assumed to be indivisible, they are allocated to an unique agent

It is a very general setting that can be applied to different topics:

- Affecting students to courses
- Inheritance sharing

Allocation procedures

Research in fair division focuses on designing procedures to allocate the resources the "fairest way".

Two perspectives can be adopted:

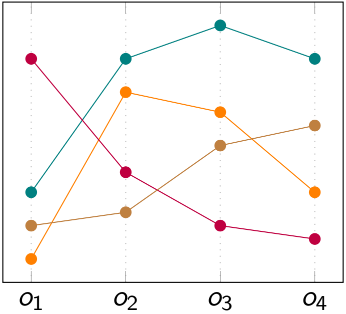
- **Centralized procedures:** a benevolent entity collects the preferences of the agents and computes the final allocation.
- **Distributed procedures:** from an initial allocation agents are allowed to negotiate, exchange their objects using a protocol previously defined.

We focus here on *house markets*: there is exactly one resource per agent.

Preferences

Definition: Single-peaked preferences

Let $L = \{\succ_1, \ldots, \succ_n\}$ be a profile. Let $\lhd$ be a linear order on $\mathcal{O}$, agent a_i' preferences $\succ_i$ are single-peaked with respect to $\lhd$ if they are decreasing on each side of a_i's top resource.
The profile L is single-peaked with respect to $\lhd$ if all the preferences are single-peaked with respect to $\lhd$.

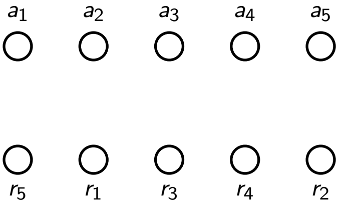


1. Centralized procedures

Consider the following preferences:

$$
\begin{aligned}
a_1 &: r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5} \\
a_2 &: r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \underline{r_1} \\
a_3 &: \underline{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 &: \underline{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 &: r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \underline{r_2} \succ_5 r_1
\end{aligned}
$$

TTC builds the following graph:

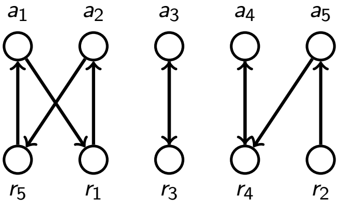


Top trading cycle

Consider the following preferences:

$$\begin{aligned}
a_1 &: r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5} \\
a_2 &: r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \underline{r_1} \\
a_3 &: \underline{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 &: \underline{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 &: r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \underline{r_2} \succ_5 r_1
\end{aligned}$$

TTC builds the following graph:

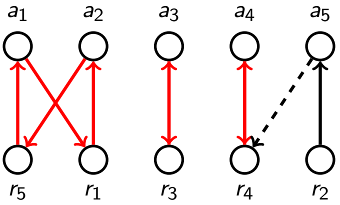


Top trading cycle

Consider the following preferences:

$$a_1 : r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5}$$
$$a_2 : r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \underline{r_1}$$
$$a_3 : \underline{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5$$
$$a_4 : \underline{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5$$
$$a_5 : r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \underline{r_2} \succ_5 r_1$$

TTC builds the following graph:

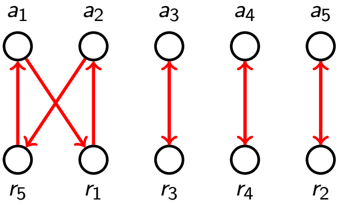


Top trading cycle

Consider the following preferences:

$$\begin{aligned}
a_1 &: r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5} \\
a_2 &: r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \underline{r_1} \\
a_3 &: \underline{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 &: \underline{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 &: r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \underline{r_2} \succ_5 r_1
\end{aligned}$$

TTC builds the following graph:

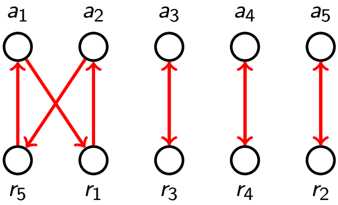


Top trading cycle

Consider the following preferences:

$$\begin{aligned}
a_1 &: \boxed{r_1} \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5} \\
a_2 &: \boxed{r_5} \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \underline{r_1} \\
a_3 &: \boxed{\underline{r_3}} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 &: \boxed{\underline{r_4}} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 &: r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \boxed{\underline{r_2}} \succ_5 r_1
\end{aligned}$$

TTC builds the following graph:



Properties of TTC

The Top Trading Cycle Algorithm satisfies all the following [1, 2] :

- Individual rationality: every agent gets something at least as good as her endowment
- Pareto optimality: there is no other allocation weakly better for everyone and strictly better for one agent
- Strategy proofness: it is optimal for every agent to reveal her true preferences

[1] Shapley and Scarf "On cores and indivisibility" (1974)
[2] Ma "Strategy-proofness and the strict core in a market with indivisibilities" (1994)

Properties of TTC

The Top Trading Cycle Algorithm satisfies all the following [1, 2] :

- Individual rationality: every agent gets something at least as good as her endowment
- Pareto optimality: there is no other allocation weakly better for everyone and strictly better for one agent
- Strategy proofness: it is optimal for every agent to reveal her true preferences

It is even the unique procedure doing so on house market with strict preferences. But not on the single peaked domain...

[1] Shapley and Scarf "On cores and indivisibility" (1974)
[2] Ma "Strategy-proofness and the strict core in a market with indivisibilities" (1994)

The Crawler Procedure

Consider the following preferences:

$$
\begin{array}{l}
a_1 : r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5} \\
a_2 : r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \underline{r_1} \\
a_3 : \underline{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 : \underline{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 : r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \underline{r_2} \succ_5 r_1
\end{array}
$$

The Crawler proceeds as follows:

$$
\begin{array}{ccccc}
r_1 & r_2 & r_3 & r_4 & r_5 \\
\bullet & \bullet & \bullet & \bullet & \bullet \\
a_2 & a_5 & a_3 & a_4 & a_1
\end{array}
$$

The Crawler Procedure

Consider the following preferences:

$$
\begin{aligned}
a_1 &: r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5}\\
a_2 &: r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \underline{r_1}\\
a_3 &: \underline{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5\\
a_4 &: \underline{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5\\
a_5 &: r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \underline{r_2} \succ_5 r_1
\end{aligned}
$$

The Crawler proceeds as follows:

$$
\begin{array}{ccccc}
r_1 & r_2 & \boxed{\begin{array}{c} r_3 \\ \bullet \\ a_3 \end{array}} & r_4 & r_5 \\
\bullet & \bullet & & \bullet & \bullet \\
a_2 & a_5 & & a_4 & a_1
\end{array}
$$

The Crawler Procedure

Consider the following preferences:

$$
\begin{array}{l}
a_1 : r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5} \\
a_2 : r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \underline{r_1} \\
a_3 : \underline{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 : \underline{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 : r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \underline{r_2} \succ_5 r_1
\end{array}
$$

The Crawler proceeds as follows:

$$
\begin{array}{ccccc}
r_1 & r_2 & r_3 & r_4 & r_5 \\
\bullet & \bullet & \bullet & \bullet & \bullet \\
a_2 & a_5 & a_3 & a_4 & a_1
\end{array}
$$

Consider the following preferences:

$$
\begin{aligned}
a_1 &: r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5} \\
a_2 &: r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \underline{r_1} \\
a_3 &: \underline{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 &: \underline{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 &: r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \underline{r_2} \succ_5 r_1
\end{aligned}
$$

The Crawler proceeds as follows:

$$
\boxed{\begin{array}{c} r_1 \\ \bullet \\ a_1 \end{array}} \quad
\begin{array}{c} r_2 \\ \bullet \\ a_2 \end{array} \quad
\boxed{\begin{array}{c} r_3 \\ \bullet \\ a_3 \end{array}} \quad
\boxed{\begin{array}{c} r_4 \\ \bullet \\ a_4 \end{array}} \quad
\begin{array}{c} r_5 \\ \bullet \\ a_5 \end{array}
$$

The Crawler Procedure

Consider the following preferences:

$$
\begin{array}{lll}
a_1 : & r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5} \\
a_2 : & r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \underline{r_1} \\
a_3 : & \underline{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 : & \underline{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 : & r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \underline{r_2} \succ_5 r_1
\end{array}
$$

The Crawler proceeds as follows:

r_1	r_2	r_3	r_4	r_5
●	●	●	●	●
a_1	a_2	a_3	a_4	a_5

The Crawler Procedure

Consider the following preferences:

$$
\begin{aligned}
a_1 &: \boxed{r_1} \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5} \\
a_2 &: r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 \boxed{r_2} \succ_2 \underline{r_1} \\
a_3 &: \boxed{\underline{r_3}} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 &: \boxed{\underline{r_4}} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 &: r_4 \succ_5 \boxed{r_5} \succ_5 r_3 \succ_5 \underline{r_2} \succ_5 r_1
\end{aligned}
$$

The Crawler proceeds as follows:

r_1	r_2	r_3	r_4	r_5
●	●	●	●	●
a_1	a_2	a_3	a_4	a_5

The Crawler satisfies the following properties [3, 4] :

- Individual rationality
- Pareto Optimality
- Strategy-proofness
- Can be implemented using obviously dominant strategies when TTC cannot

[3] Bade "Matching with single-peaked preferences" (2019)
[4] Li "Obviously strategy-proof mechanisms" (2017)

The Crawler satisfies the following properties [3, 4] :

- Individual rationality
- Pareto Optimality
- Strategy-proofness
- Can be implemented using obviously dominant strategies when TTC cannot

However TTC and the Crawler possibly implement long cycles which is not suitable in practice.

➥ Need for decentralized procedures

[3] Bade "Matching with single-peaked preferences" (2019)
[4] Li "Obviously strategy-proof mechanisms" (2017)

2. Decentralized procedures

The *swap-deal procedure* proceeds as follow:

1. All the resources are randomly allocated to the agents
2. Agents sequentially perform improving swap-deals until having reached a stable allocation with respect to swap-deals

Example of the swap-deal procedure

Consider the following preferences:

$$a_1 : r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \boxed{\underline{r_5}}$$

$$a_2 : r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \boxed{\underline{r_1}}$$

$$a_3 : \boxed{\underline{r_3}} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5$$

$$a_4 : \boxed{\underline{r_4}} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5$$

$$a_5 : r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \boxed{\underline{r_2}} \succ_5 r_1$$

There are two possible deals: either $\langle a_1, a_2 \rangle$ or $\langle a_1, a_5 \rangle$.

Example of the swap-deal procedure

Applying $\langle a_1, a_2 \rangle$ leads to :

$$
\begin{aligned}
a_1 &: \boxed{r_1} \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5} \\
a_2 &: \boxed{r_5} \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \underline{r_1} \\
a_3 &: \boxed{\underline{r_3}} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 &: \boxed{\underline{r_4}} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 &: r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \boxed{\underline{r_2}} \succ_5 r_1
\end{aligned}
$$

This allocation is stable with respect to swap-deals. It corresponds to the allocation returned by TTC.

Example of the swap-deal procedure

Applying $\langle a_1, a_5 \rangle$ leads to:

$$a_1 : r_1 \succ_1 \boxed{r_2} \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5}$$

$$a_2 : r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \boxed{\underline{r_1}}$$

$$a_3 : \boxed{\underline{r_3}} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5$$

$$a_4 : \boxed{\underline{r_4}} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5$$

$$a_5 : r_4 \succ_5 \boxed{r_5} \succ_5 r_3 \succ_5 \underline{r_2} \succ_5 r_1$$

Example of the swap-deal procedure

Applying $\langle a_1, a_2 \rangle$ leads to:

$$
\begin{aligned}
a_1 &: \boxed{r_1} \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \underline{r_5} \\
a_2 &: r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 \boxed{r_2} \succ_2 \underline{r_1} \\
a_3 &: \boxed{\underline{r_3}} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 &: \boxed{\underline{r_4}} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 &: r_4 \succ_5 \boxed{r_5} \succ_5 r_3 \succ_5 \underline{r_2} \succ_5 r_1
\end{aligned}
$$

This allocation is stable with respect to swap-deals. It corresponds to the allocation returned by the Crawler.

Properties of the swap-deal procedure

The swap-deal procedure satisfies the following [5, 6] :

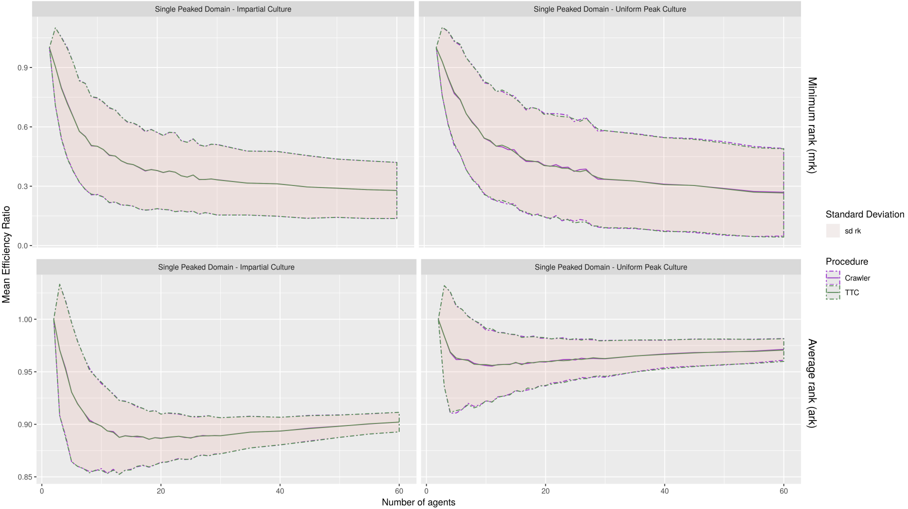
- Individual Rationality
- Pareto-Optimality
- *not* Strategy-proof

Moreover, the single-peaked domain is a maximal domain for the Pareto-optimality of the procedure: every domain strictly larger than the single-peaked one induces at least one instance on which the swap-deal procedure is not Pareto-optimal.

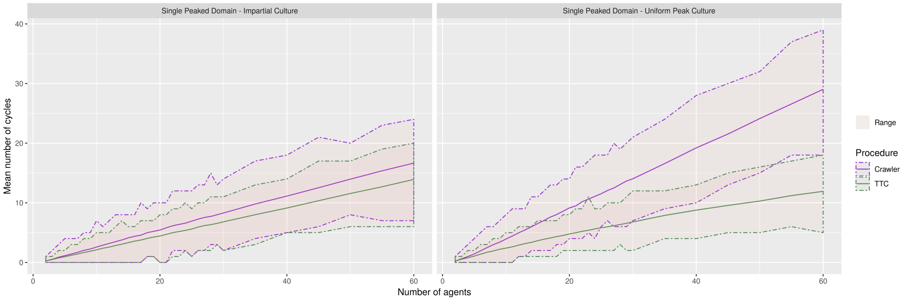
[5] Damamme, Beynier, Chevaleyre, and Maudet "The power of swap deals in distributed resource allocation" (2015)
[6] Beynier, Bouveret, Lemaitre, Maudet, Rey, and Shams "Efficiency, Sequenceability and Deal-Optimality in Fair Division of Indivisible Goods" (2019)

3. Experimental analysis of the procedures

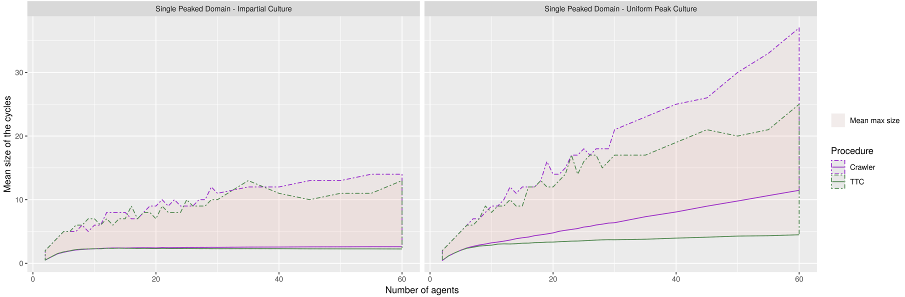
Comparison between the Crawler and TTC



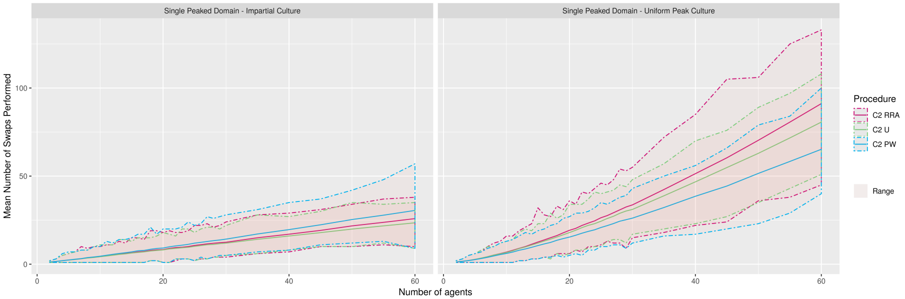
Number of cycle-deals



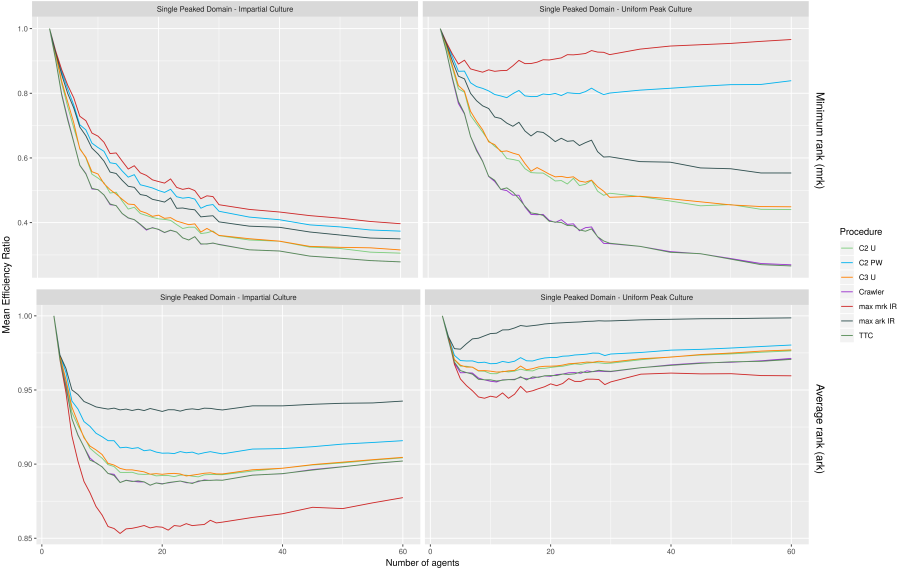
Size of the cycle-deals



Number of swap-deals



Efficiency of the procedures



4. Checking Pareto-optimality efficiently

The problem

IS-PO	
Instance:	A set of agents $\mathcal{N}$, a preference profile $(\succ_i)_{i \in \mathcal{N}}$ and an allocation π.
Question:	Does π satisfy Pareto-optimality?

Time complexity

Since TTC and the Crawler are individually rational and Pareto-optimal, an allocation π is Pareto-optimal if and only if π is returned by TTC or the Crawler when it is given as the input of these procedures.

- TTC would stop at the first if it does not find any cycle, hence time-complexity in $\mathcal{O}(n^2)$.
- The Crawler has the same worst-case scenario: consider the instance where every agent has their next resource as her peak.

Time complexity

Since TTC and the Crawler are individually rational and Pareto-optimal, an allocation π is Pareto-optimal if and only if π is returned by TTC or the Crawler when it is given as the input of these procedures.

- TTC would stop at the first if it does not find any cycle, hence time-complexity in $\mathcal{O}(n^2)$.
- The Crawler has the same worst-case scenario: consider the instance where every agent has their next resource as her peak.

➥ The Crawler cannot be used to get a better time-complexity bound for IS-PO.

The Diver

Consider the following preferences:

$$
\begin{aligned}
a_1 &: r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \boxed{r_5} \\
a_2 &: r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \boxed{r_1} \\
a_3 &: \boxed{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 &: \boxed{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 &: r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \boxed{r_2} \succ_5 r_1
\end{aligned}
$$

Sorting the agents according to $\lhd$:

r_1	r_2	r_3	r_4	r_5
$\bullet$	$\bullet$	$\bullet$	$\bullet$	$\bullet$
a_2	a_5	a_3	a_4	a_1

The Diver asks every agent whether she wishes to pick her resource, pass or pick a smaller resource.

Consider the following preferences:

$$a_1 : r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \boxed{r_5}$$

$$a_2 : r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \boxed{r_1}$$

$$a_3 : \boxed{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5$$

$$a_4 : \boxed{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5$$

$$a_5 : r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \boxed{r_2} \succ_5 r_1$$

Sorting the agents according to $\lhd$:

r_1	r_2	r_3	r_4	r_5
●	●	●	●	●
a_2	a_5	a_3	a_4	a_1

a_2 passes

The Diver

Consider the following preferences:

$$a_1 : r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \boxed{r_5}$$

$$a_2 : r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \boxed{r_1}$$

$$a_3 : \boxed{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5$$

$$a_4 : \boxed{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5$$

$$a_5 : r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \boxed{r_2} \succ_5 r_1$$

Sorting the agents according to $\lhd$:

r_1	r_2	r_3	r_4	r_5
●	●	●	●	●
a_2	a_5	a_3	a_4	a_1

a_5 passes

The Diver

Consider the following preferences:

$$a_1: \; r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \boxed{r_5}$$
$$a_2: \; r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \boxed{r_1}$$
$$a_3: \; \boxed{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5$$
$$a_4: \; \boxed{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5$$
$$a_5: \; r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \boxed{r_2} \succ_5 r_1$$

Sorting the agents according to $\lhd$:

r_1	r_2	r_3	r_4	r_5
●	●	●	●	●
a_2	a_5	a_3	a_4	a_1

a_3 picks her resource, a_5 still agrees to pass

The Diver

Consider the following preferences:

$$a_1: r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \boxed{r_5}$$

$$a_2: r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \boxed{r_1}$$

$$a_3: \boxed{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5$$

$$a_4: \boxed{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5$$

$$a_5: r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \boxed{r_2} \succ_5 r_1$$

Sorting the agents according to $\lhd$:

r_1	r_2	r_3	r_4	r_5
●	●	●	●	●
a_2	a_5	a_3	a_4	a_1

a_4 picks her resource, a_5 still agrees to pass

The Diver

Consider the following preferences:

$$a_1 : r_1 \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 \boxed{r_5}$$
$$a_2 : r_5 \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 \boxed{r_1}$$
$$a_3 : \boxed{r_3} \succ_3 r_2 \succ_3 r_1 \succ_3 r_4 \succ_3 r_5$$
$$a_4 : \boxed{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5$$
$$a_5 : r_4 \succ_5 r_5 \succ_5 r_3 \succ_5 \boxed{r_2} \succ_5 r_1$$

Sorting the agents according to $\lhd$:

r_1	r_2	r_3	r_4	r_5
●	●	●	●	●
a_2	a_5	a_3	a_4	a_1

a_1 wants a smaller resource, *the allocation is not Pareto-optimal*.

The Diver

Consider the following preferences:

$$
\begin{aligned}
a_1 &: \boxed{r_1} \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 r_5 \\
a_2 &: \boxed{r_5} \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 r_1 \\
a_3 &: r_3 \succ_3 \boxed{r_2} \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 &: \boxed{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 &: r_4 \succ_5 r_5 \succ_5 \boxed{r_3} \succ_5 r_2 \succ_5 r_1
\end{aligned}
$$

Sorting the agents according to $\lhd$:

$$
\begin{array}{ccccc}
r_1 & r_2 & r_3 & r_4 & r_5 \\
\bullet & \bullet & \bullet & \bullet & \bullet \\
a_1 & a_3 & a_5 & a_4 & a_2
\end{array}
$$

The Diver asks every agent whether she wishes to pick her resource, pass or pick a smaller resource.

The Diver

Consider the following preferences:

$$a_1 : \boxed{r_1} \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 r_5$$

$$a_2 : \boxed{r_5} \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 r_1$$

$$a_3 : r_3 \succ_3 \boxed{r_2} \succ_3 r_1 \succ_3 r_4 \succ_3 r_5$$

$$a_4 : \boxed{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5$$

$$a_5 : r_4 \succ_5 r_5 \succ_5 \boxed{r_3} \succ_5 r_2 \succ_5 r_1$$

Sorting the agents according to $\lhd$:

r_1	r_2	r_3	r_4	r_5
●	●	●	●	●
a_1	a_3	a_5	a_4	a_2

a_1 picks her resource

The Diver

Consider the following preferences:

$$a_1 : \boxed{r_1} \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 r_5$$
$$a_2 : \boxed{r_5} \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 r_1$$
$$a_3 : r_3 \succ_3 \boxed{r_2} \succ_3 r_1 \succ_3 r_4 \succ_3 r_5$$
$$a_4 : \boxed{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5$$
$$a_5 : r_4 \succ_5 r_5 \succ_5 \boxed{r_3} \succ_5 r_2 \succ_5 r_1$$

Sorting the agents according to $\lhd$:

r_1	r_2	r_3	r_4	r_5
●	●	●	●	●
a_1	a_3	a_5	a_4	a_2

a_3 passes

The Diver

Consider the following preferences:

$$\begin{array}{l}
a_1 : \boxed{r_1} \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 r_5 \\
a_2 : \boxed{r_5} \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 r_1 \\
a_3 : r_3 \succ_3 \boxed{r_2} \succ_3 r_1 \succ_3 r_4 \succ_3 r_5 \\
a_4 : \boxed{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5 \\
a_5 : r_4 \succ_5 r_5 \succ_5 \boxed{r_3} \succ_5 r_2 \succ_5 r_1
\end{array}$$

Sorting the agents according to $\lhd$:

r_1	r_2	r_3	r_4	r_5
●	●	●	●	●
a_1	a_3	a_5	a_4	a_2

a_5 passes

The Diver

Consider the following preferences:

$$a_1 : \boxed{r_1} \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 r_5$$

$$a_2 : \boxed{r_5} \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 r_1$$

$$a_3 : r_3 \succ_3 \boxed{r_2} \succ_3 r_1 \succ_3 r_4 \succ_3 r_5$$

$$a_4 : \boxed{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5$$

$$a_5 : r_4 \succ_5 r_5 \succ_5 \boxed{r_3} \succ_5 r_2 \succ_5 r_1$$

Sorting the agents according to $\lhd$:

r_1	r_2	r_3	r_4	r_5
●	●	●	●	●
a_1	a_3	a_5	a_4	a_2

a_4 picks her resource, a_5 still agrees to pass

The Diver

Consider the following preferences:

$$a_1 : \boxed{r_1} \succ_1 r_2 \succ_1 r_3 \succ_1 r_4 \succ_1 r_5$$

$$a_2 : \boxed{r_5} \succ_2 r_4 \succ_2 r_3 \succ_2 r_2 \succ_2 r_1$$

$$a_3 : r_3 \succ_3 \boxed{r_2} \succ_3 r_1 \succ_3 r_4 \succ_3 r_5$$

$$a_4 : \boxed{r_4} \succ_4 r_3 \succ_4 r_2 \succ_4 r_1 \succ_4 r_5$$

$$a_5 : r_4 \succ_5 r_5 \succ_5 \boxed{r_3} \succ_5 r_2 \succ_5 r_1$$

Sorting the agents according to $\lhd$:

r_1	r_2	r_3	r_4	r_5
●	●	●	●	●
a_1	a_3	a_5	a_4	a_2

a_2 picks her resource, so is a_5, so is a_3, *the allocation is Pareto-optimal*.

Optimality of the Diver

THEOREM:

The Diver returns in time $\mathcal{O}(n \log n)$ whether the given allocation is Pareto-optimal or not.

THEOREM:

The Diver requires $4n$ bits of communication.

PROOF: The key is to observe that sub-protocol p' requires overall $n + n$ bits, as there may only be n agents saying 'no' and n agents saying 'yes' throughout the whole run of the Diver. In the main loop of the protocol, the question requires 2 bits to be answered. This makes overall $2n + 2n = 4n$ bits.

Lower bound on the communication complexity of IS-PO

Definition: Fooling set

In our context, the fooling set will be a collection of profiles $\mathcal{F} = \langle L_1, \ldots L_K \rangle$ such that:

1. for any $i \in \{1, \ldots K\}$ CHECKPO's answer on $\langle \mathcal{N}, \mathcal{O}, L_i, \pi^0 \rangle$ is yes.
2. for any $i \neq j$, there exists $L' = \langle \succ_1^{L_i|L_j}, \cdots \succ_n^{L_i|L_j} \rangle$, such that CHECKPO's answer on $\langle \mathcal{N}, \mathcal{O}, L', \pi^0 \rangle$ is no.

By a standard result in communication complexity, it is known that $\log |\mathcal{F}|$ is a lower bound on the communication complexity of the problem [7] .

[7] Kushilevitz and Nisan *Communication Complexity* (1996)

Lower bound on the communication complexity of IS-PO

PROPOSITION:

The communication complexity of CHECKPO IS $\Omega(n)$.

PROOF: We consider consensual profile where every agent has the same preferences written $\succ$. We show that the set $\mathcal{F}$ of the 2^{n-1} consensual profiles constitutes a fooling set.

First, for a consensual profile, every allocation is PO, hence for every profile in $\mathcal{F}$, the answer of IS-PO is yes.

Lower bound on the communication complexity of IS-PO

PROPOSITION:

The communication complexity of CHECKPO IS $\Omega(n)$.

PROOF: Consider any pair of profiles (L_i, L_j). Because these profiles are different, there is (r_p, r_q) such that $r_p \succ r_q$ in L_i, while $r_q \succ r_p$ in L_j. We call a_p (resp. a_q) the agent holding r_p (resp. r_q) in π^0. Let us now consider a mixed profile L' such that:

$$\forall k \in \mathcal{N}, \succ_k^{L'} = \begin{cases} \succ_k^{L_i} & \text{if } k \neq p, \\ \succ_k^{L_j} & \text{if } k = p. \end{cases}$$

Observe that now a_p and a_q have opposite preferences for r_p and r_q and would prefer to swap, i.e. CHECKPO's answer on $\langle \mathcal{N}, \mathcal{O}, L', \pi^0 \rangle$ is no.

5. Conclusion

The story

- Presentation of TTC and Crawler, two procedures individually rational, Pareto optimal and strategy-proof on the single-peaked domain.

The story

- Presentation of TTC and Crawler, two procedures individually rational, Pareto optimal and strategy-proof on the single-peaked domain.
- These procedures are centralized which implies many drawbacks

The story

- Presentation of TTC and Crawler, two procedures individually rational, Pareto optimal and strategy-proof on the single-peaked domain.
- These procedures are centralized which implies many drawbacks
- The swap-deal procedure is decentralized, individually rational and Pareto optimal. It show very nice results in practice but is not strategy-proof.

The story

- Presentation of TTC and Crawler, two procedures individually rational, Pareto optimal and strategy-proof on the single-peaked domain.
- These procedures are centralized which implies many drawbacks
- The swap-deal procedure is decentralized, individually rational and Pareto optimal. It show very nice results in practice but is not strategy-proof.
- The Crawler can be modified to obtain a procedure optimal in terms of communication complexity to check whether an allocation is Pareto-optimal.