Approximability of optimal social welfare in multiagent resource allocation with cardinal and ordinal preferences

Nhan-Tam Nguyen
Heinrich-Heine Universität Düsseldorf

This is in part joint work with Dorothea Baumeister, Jörg Rothe, Sylvain Bouveret, Jérôme Lang, Trung Thanh Nguyen, and Abdallah Saffidine.

Meeting of COST Action IC 1205
Sibiu, October 20–22, 2014
Outline

Resource allocation under...

1. Cardinal preferences
2. Ordinal preferences & Restricted model
Part One

Cardinal preferences
Resource allocation

Example

Undirected graph with
- parallel edges
- self loops
- nonuniform weights
Example

Undirected graph with
- parallel edges
- self loops
- nonuniform weights

**Goal:** Orient the edges so that we maximize the minimum sum of incoming weights.
Resource allocation

Example

Undirected graph with
- parallel edges
- self loops
- nonuniform weights

**Goal:** Orient the edges so that we maximize the minimum sum of incoming weights.
Undirected graph with
- parallel edges
- self loops
- nonuniform weights

**Goal:** Orient the edges so that we maximize the minimum sum of incoming weights.

- vertex = agent
- edge = object
- sum of incoming weights = utility
What kind of resource allocation problem do we deal with?

- indivisible and non-shareable goods
- centralized
- no payments
- non-strategic agents
What kind of resource allocation problem do we deal with?
- indivisible and non-shareable goods
- centralized
- no payments
- non-strategic agents

Formal model:
- a finite set of objects \( \mathcal{O} = \{o_1, \ldots, o_m\} \)
- a finite set of agents \( \mathcal{A} = \{1, \ldots, n\} \)
- each agent \( i \in \mathcal{A} \) has utility function \( u_i : 2^{\mathcal{O}} \rightarrow \mathbb{Q} \)
Resource allocation

Representations of utility functions

**Problem:** Utility functions have an exponential-size domain.

Each $u_i$ is represented ...
**Problem:** Utility functions have an exponential-size domain.

Each $u_i$ is represented . . .

- in the **bundle form** by a list of pairs $(S, u_i(S))$ with $u_i(S) \neq 0$,
Problem: Utility functions have an exponential-size domain.

Each $u_i$ is represented . . .

- in the **bundle form** by
  a list of pairs $(S, u_i(S))$ with $u_i(S) \neq 0$,

- in the **$k$-additive form** by
  coefficients $\alpha^S_i$ for each bundle $S \subseteq \mathcal{O}$ with $\|S\| \leq k$ such that

$$u_i(T) = \sum_{S \subseteq T, \|S\| \leq k} \alpha^S_i.$$
Problem: Utility functions have an exponential-size domain.

Each $u_i$ is represented . . .

- in the bundle form by a list of pairs $(S, u_i(S))$ with $u_i(S) \neq 0$,
- in the $k$-additive form by coefficients $\alpha_i^S$ for each bundle $S \subseteq O$ with $\|S\| \leq k$ such that

$$u_i(T) = \sum_{S \subseteq T, \|S\| \leq k} \alpha_i^S.$$ 

Other representations are e.g., straight-line programs, bidding languages, weighted goals.
Restrictions on utility functions

Usually we have assumptions such as...

- no externalities
- **monotonicity** (free disposal)
  
  \[ S \subseteq T \implies u(S) \leq u(T), \]

- **normalization**
  
  \[ u(\emptyset) = 0. \]
Restrictions on utility functions

Usually we have assumptions such as...

- no externalities
- **monotonicity** (free disposal)
  \[ S \subseteq T \implies u(S) \leq u(T), \]
- **normalization**
  \[ u(\emptyset) = 0. \]

We may put additional restrictions on utility functions:

- **additivity**
  \[ u(S) = \sum_{o \in S} u(\{o\}) \]
Restrictions on utility functions

Usually we have assumptions such as...

- no externalities
- monotonicity (free disposal)

\[ S \subseteq T \implies u(S) \leq u(T), \]

- normalization

\[ u(\emptyset) = 0. \]

We may put additional restrictions on utility functions:

- additivity

\[ u(S) = \sum_{o \in S} u(\{o\}) \]

- submodularity

\[ u(S \cup T) + u(S \cap T) \leq u(S) + u(T) \]
Resource allocation

Restrictions on utility functions

Usually we have assumptions such as...

- no externalities
- monotonicity (free disposal)

\[ S \subseteq T \implies u(S) \leq u(T), \]

- normalization

\[ u(\emptyset) = 0. \]

We may put additional restrictions on utility functions:

- additivity

\[ u(S) = \sum_{o \in S} u(\{o\}) \]

- submodularity

\[ u(S \cup T) + u(S \cap T) \leq u(S) + u(T) \]

- subadditivity

\[ u(S \cup T) \leq u(S) + u(T) \]
A *solution* in this model is a partition of O into n disjoint subsets.

**Question:** How to assess the quality of a solution?
A **solution** in this model is a partition of $\mathcal{O}$ into $n$ disjoint subsets.

**Question:** How to assess the quality of a solution?

We can ask questions such as
A **solution** in this model is a partition of $\mathcal{O}$ into $n$ disjoint subsets.

**Question:** How to assess the quality of a solution?

We can ask questions such as

- Which *(fairness) properties* are satisfied?
A solution in this model is a partition of $\mathcal{O}$ into $n$ disjoint subsets.

**Question:** How to assess the quality of a solution?

We can ask questions such as

- Which **(fairness) properties** are satisfied?
- What is the **social welfare**?
We can aggregate utility values with a collective utility function.

- **Utilitarian social welfare**
  \[ sw_u(\pi) = \sum_{i \in A} u_i(\pi) \]

- **Egalitarian (Rawlsian) social welfare**
  \[ sw_e(\pi) = \min_{i \in A} u_i(\pi) \]

- **Nash product**
  \[ sw_N(\pi) = \prod_{i \in A} u_i(\pi) \]
We can aggregate utility values with a **collective utility function**.

- Utilitarian social welfare

  \[ sw_u(\pi) = \sum_{i \in A} u_i(\pi) \]

- Egalitarian (Rawlsian) social welfare

  \[ sw_e(\pi) = \min_{i \in A} u_i(\pi) \]

- Nash product

  \[ sw_N(\pi) = \prod_{i \in A} u_i(\pi) \]

We also consider the lexicographic minimum (leximin) over \((u_i(\pi))_{i \in A}\).
We can aggregate utility values with a collective utility function.

- **Utilitarian social welfare**
  \[ sw_u(\pi) = \sum_{i \in A} u_i(\pi) \]

- **Egalitarian (Rawlsian) social welfare**
  \[ sw_e(\pi) = \min_{i \in A} u_i(\pi) \]

- **Nash product**
  \[ sw_N(\pi) = \prod_{i \in A} u_i(\pi) \]

We also consider the lexicographic minimum (leximin) over \((u_i(\pi))_{i \in A}\).

There are also approaches using inequality indices, e.g., Gini index.
Resource allocation

Example

<table>
<thead>
<tr>
<th></th>
<th>$o_1$</th>
<th>$o_2$</th>
<th>$o_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>agent 1</td>
<td>5</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>agent 2</td>
<td>1</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>
Resource allocation

Example

<table>
<thead>
<tr>
<th></th>
<th>$o_1$</th>
<th>$o_2$</th>
<th>$o_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>agent 1</td>
<td>5</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>agent 2</td>
<td>1</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>

Utilitarian social welfare:

$\pi = \langle \{o_1\}, \{o_2, o_3\} \rangle \rightarrow sw_u(\pi) = 5 + (4 + 6) = 15$
Example

<table>
<thead>
<tr>
<th></th>
<th>$o_1$</th>
<th>$o_2$</th>
<th>$o_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>agent 1</td>
<td>5</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>agent 2</td>
<td>1</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>

Utilitarian social welfare:

$\pi = \langle \{o_1\}, \{o_2, o_3\} \rangle \rightarrow sw_u(\pi) = 5 + (4 + 6) = 15$

$\pi' = \langle \{o_1, o_2\}, \{o_3\} \rangle \rightarrow sw_u(\pi') = (5 + 1) + 6 = 12$
Resource allocation

Example

<table>
<thead>
<tr>
<th></th>
<th>$o_1$</th>
<th>$o_2$</th>
<th>$o_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>agent 1</td>
<td>5</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>agent 2</td>
<td>1</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>

**Utilitarian social welfare:**

$\pi = \langle \{o_1\}, \{o_2, o_3\} \rangle \rightarrow sw_u(\pi) = 5 + (4 + 6) = 15$

$\pi' = \langle \{o_1, o_2\}, \{o_3\} \rangle \rightarrow sw_u(\pi') = (5 + 1) + 6 = 12$

**Egalitarian social welfare:**

$\pi = \langle \{o_1\}, \{o_2, o_3\} \rangle \rightarrow sw_e(\pi) = \min(5, 4 + 6) = 5$
Resource allocation

Example

<table>
<thead>
<tr>
<th></th>
<th>$o_1$</th>
<th>$o_2$</th>
<th>$o_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>agent 1</td>
<td>5</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>agent 2</td>
<td>1</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>

**Utilitarian social welfare:**

\[\pi = \langle \{o_1\}, \{o_2, o_3\} \rangle \rightarrow sw_u(\pi) = 5 + (4 + 6) = 15\]

\[\pi' = \langle \{o_1, o_2\}, \{o_3\} \rangle \rightarrow sw_u(\pi') = (5 + 1) + 6 = 12\]

**Egalitarian social welfare:**

\[\pi = \langle \{o_1\}, \{o_2, o_3\} \rangle \rightarrow sw_e(\pi) = \min(5, 4 + 6) = 5\]

\[\pi' = \langle \{o_1, o_2\}, \{o_3\} \rangle \rightarrow sw_e(\pi') = \min(5 + 1, 6) = 6\]
Example

<table>
<thead>
<tr>
<th></th>
<th>$o_1$</th>
<th>$o_2$</th>
<th>$o_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>agent 1</td>
<td>5</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>agent 2</td>
<td>1</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>

**Utilitarian social welfare:**
\[
\pi = \langle \{o_1\}, \{o_2, o_3\} \rangle \rightarrow sw_u(\pi) = 5 + (4 + 6) = 15
\]
\[
\pi' = \langle \{o_1, o_2\}, \{o_3\} \rangle \rightarrow sw_u(\pi') = (5 + 1) + 6 = 12
\]

**Egalitarian social welfare:**
\[
\pi = \langle \{o_1\}, \{o_2, o_3\} \rangle \rightarrow sw_e(\pi) = \min(5, 4 + 6) = 5
\]
\[
\pi' = \langle \{o_1, o_2\}, \{o_3\} \rangle \rightarrow sw_e(\pi') = \min(5 + 1, 6) = 6
\]

**Nash product:**
\[
\pi = \langle \{o_1\}, \{o_2, o_3\} \rangle \rightarrow sw_N(\pi) = 5 \times (4 + 6) = 50
\]
Example

<table>
<thead>
<tr>
<th></th>
<th>$o_1$</th>
<th>$o_2$</th>
<th>$o_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>agent 1</td>
<td>5</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>agent 2</td>
<td>1</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>

**Utilitarian social welfare:**

$\pi = \langle \{o_1\}, \{o_2, o_3\} \rangle \rightarrow sw_u(\pi) = 5 + (4 + 6) = 15$

$\pi' = \langle \{o_1, o_2\}, \{o_3\} \rangle \rightarrow sw_u(\pi') = (5 + 1) + 6 = 12$

**Egalitarian social welfare:**

$\pi = \langle \{o_1\}, \{o_2, o_3\} \rangle \rightarrow sw_e(\pi) = \min(5, 4 + 6) = 5$

$\pi' = \langle \{o_1, o_2\}, \{o_3\} \rangle \rightarrow sw_e(\pi') = \min(5 + 1, 6) = 6$

**Nash product:**

$\pi = \langle \{o_1\}, \{o_2, o_3\} \rangle \rightarrow sw_N(\pi) = 5 \times (4 + 6) = 50$

$\pi' = \langle \{o_1, o_2\}, \{o_3\} \rangle \rightarrow sw_N(\pi') = (5 + 1) \times 6 = 36$
How hard is it to compute allocations of optimal social welfare?
How hard is it to compute allocations of optimal social welfare?

**Bundle form:** Utilitarian, egalitarian, and Nash product social welfare optimization are hard.
How hard is it to compute allocations of optimal social welfare?

**Bundle form:**
Utilitarian, egalitarian, and Nash product social welfare optimization are hard.

**$k$-additive form, $k \geq 1$:**
Utilitarian (except for $k = 1$), egalitarian, and Nash product social welfare optimization are hard.

But for $m = n$ it is easy.
Since most social welfare optimization problems are \textit{hard}, we have to relax our requirements.
Since most social welfare optimization problems are **hard**, we have to relax our requirements.

- Approximate notions of properties
Since most social welfare optimization problems are **hard**, we have to relax our requirements.

- Approximate notions of properties
- Suboptimal social welfare
Since most social welfare optimization problems are hard, we have to relax our requirements.

- Approximate notions of properties
- Suboptimal social welfare
- Restricted model of computation (Part two)
## Utilitarian social welfare

<table>
<thead>
<tr>
<th>Bundle form</th>
<th>Approximability</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>general</td>
<td>NP-hard in factor $m^{e-1/2}$</td>
<td>[LOS99]</td>
</tr>
<tr>
<td>submodular</td>
<td>$1 - (1/e)$</td>
<td>[FGMS06],[CCPV07],[Von08]</td>
</tr>
<tr>
<td></td>
<td>NP-hard in factor $1 - (1/e) + \varepsilon$</td>
<td>[KLMM08]</td>
</tr>
<tr>
<td>subadditive</td>
<td>hard in factor $1/m^{1/4}$</td>
<td>[DS06]</td>
</tr>
<tr>
<td></td>
<td>$1/m^{1/2}$</td>
<td>[DNS10]</td>
</tr>
</tbody>
</table>
## Approximating social welfare

### Egalitarian social welfare

<table>
<thead>
<tr>
<th>Bundle form</th>
<th>Approximability</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>general</td>
<td>NP-hard in any factor</td>
<td>[NRR13]</td>
</tr>
<tr>
<td>submodular</td>
<td>$1/(m - n + 1)$</td>
<td>[Gol05]</td>
</tr>
<tr>
<td></td>
<td>$1/(m^{1/2} n^{1/4} \log m \log^{3/2} n)$</td>
<td>[GHIM09]</td>
</tr>
<tr>
<td>subadditive</td>
<td>$1/(2n - 1)$</td>
<td>[KP07]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$k$-additive form</th>
<th>Approximability</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-additive</td>
<td>NP-hard in factor $1/2 + \varepsilon$</td>
<td>[BD05]</td>
</tr>
<tr>
<td></td>
<td>$1/m^\varepsilon$, $\varepsilon \in \mathcal{O}(1)$</td>
<td>[CCK09]</td>
</tr>
<tr>
<td>1-additive, Santa Claus</td>
<td>$\mathcal{O}(1)$</td>
<td>[Fei08],[HSS11]</td>
</tr>
<tr>
<td>$k$-additive, $k \geq 2$</td>
<td>NP-hard in any factor</td>
<td>[NRR13]</td>
</tr>
</tbody>
</table>
## Nash product

<table>
<thead>
<tr>
<th>Restriction</th>
<th>Approximability</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bundle form</td>
<td>NP-hard in any factor</td>
<td>[NRR13]</td>
</tr>
<tr>
<td>1-additive</td>
<td>$1/(m - n + 1)^n$</td>
<td>[NR13]</td>
</tr>
<tr>
<td>2-additive</td>
<td>NP-hard in factor $21/22 + \varepsilon$</td>
<td>[NRR13]</td>
</tr>
<tr>
<td>$k$-additive, $k \geq 3$</td>
<td>NP-hard in any factor</td>
<td>[NRR13]</td>
</tr>
</tbody>
</table>
Part Two

Ordinal preferences & Restricted model
Ordinal preferences

Our approach

Problem: What to do if no clear numerical scale (e.g., money) is involved...?
Problem: What to do if no clear numerical scale (e.g., money) is involved...?

Idea: *Cardinalize ordinal preferences with the help of scoring vectors.*
Here we take inspiration from voting theory.
Here we take inspiration from voting theory.

We assume that:

1. **Ranking**: Each agent $i$ has a ranking $\succ_i$ over $O$ (e.g., $o_6 \succ o_1 \succ o_4 \succ o_5 \succ o_2 \succ o_3$).

2. **Scoring**: We have a common scoring vector $s = (s_1, \ldots, s_m)$ (with $s_1 \geq \cdots \geq s_m$) mapping each rank to a utility.

3. **Additivity**: These utilities are additive.
About scoring vectors

Here we take inspiration from voting theory.

We assume that:

1. **Ranking**: Each agent $i$ has a ranking $\succ_i$ over $\mathcal{O}$ (ex: $o_6 \succ o_1 \succ o_4 \succ o_5 \succ o_2 \succ o_3$)
About scoring vectors

Here we take inspiration from voting theory.

We assume that:

1. **Ranking**: Each agent $i$ has a ranking $\succ_i$ over $\mathcal{O}$ (ex: $o_6 \succ o_1 \succ o_4 \succ o_5 \succ o_2 \succ o_3$)

2. **Scoring**: We have a common scoring vector $s = (s_1, \ldots, s_m)$ (with $s_1 \geq \cdots \geq s_m$) mapping each rank to a utility.
About scoring vectors

Here we take inspiration from voting theory.

We assume that:

1. **Ranking:** Each agent $i$ has a ranking $\succ_i$ over $\mathcal{O}$ (ex: $o_6 \succ o_1 \succ o_4 \succ o_5 \succ o_2 \succ o_3$)

2. **Scoring:** We have a common scoring vector $s = (s_1, \ldots, s_m)$ (with $s_1 \geq \cdots \geq s_m$) mapping each rank to a utility.

3. **Additivity:** These utilities are additive.
Scoring allocation rules

About scoring vectors

Here we take inspiration from voting theory.

We assume that:

1. **Ranking:** Each agent $i$ has a ranking $\succ_i$ over $\mathcal{O}$ (ex: $o_6 \succ o_1 \succ o_4 \succ o_5 \succ o_2 \succ o_3$)

2. **Scoring:** We have a common scoring vector $s = (s_1, \ldots, s_m)$ (with $s_1 \geq \cdots \geq s_m$) mapping each rank to a utility.

3. **Additivity:** These utilities are additive.

4 natural scoring vectors:

<table>
<thead>
<tr>
<th>$\succ_i$</th>
<th>$o_6$</th>
<th>$o_1$</th>
<th>$o_4$</th>
<th>$o_5$</th>
<th>$o_2$</th>
<th>$o_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Borda</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>
Here we take inspiration from voting theory.

We assume that:

1. **Ranking**: Each agent $i$ has a ranking $\succ_i$ over $O$ (ex: $o_6 \succ o_1 \succ o_4 \succ o_5 \succ o_2 \succ o_3$)

2. **Scoring**: We have a common scoring vector $s = (s_1, \ldots, s_m)$ (with $s_1 \geq \cdots \geq s_m$) mapping each rank to a utility.

3. **Additivity**: These utilities are additive.

4 natural scoring vectors:

<table>
<thead>
<tr>
<th>$\succ_i$</th>
<th>$o_6$</th>
<th>$o_1$</th>
<th>$o_4$</th>
<th>$o_5$</th>
<th>$o_2$</th>
<th>$o_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Borda</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Lexicographic</td>
<td>32</td>
<td>16</td>
<td>8</td>
<td>4</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

Approximability of optimal social welfare
Here we take inspiration from voting theory.

We assume that:

1. **Ranking**: Each agent $i$ has a ranking $\succ_i$ over $O$ (ex: $o_6 \succ o_1 \succ o_4 \succ o_5 \succ o_2 \succ o_3$)

2. **Scoring**: We have a common scoring vector $s = (s_1, \ldots, s_m)$ (with $s_1 \geq \cdots \geq s_m$) mapping each rank to a utility.

3. **Additivity**: These utilities are additive.

4 natural scoring vectors:

<table>
<thead>
<tr>
<th>$\succ_i$</th>
<th>$o_6$</th>
<th>$o_1$</th>
<th>$o_4$</th>
<th>$o_5$</th>
<th>$o_2$</th>
<th>$o_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Borda</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Lexicographic</td>
<td>32</td>
<td>16</td>
<td>8</td>
<td>4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Quasi-Indifference</td>
<td>$1 + \frac{s_1}{M}$</td>
<td>$1 + \frac{s_2}{M}$</td>
<td>$1 + \frac{s_3}{M}$</td>
<td>$1 + \frac{s_4}{M}$</td>
<td>$1 + \frac{s_5}{M}$</td>
<td>$1 + \frac{s_6}{M}$</td>
</tr>
</tbody>
</table>
Scoring allocation rules

About scoring vectors

Here we take inspiration from voting theory.

We assume that:

1. **Ranking**: Each agent $i$ has a ranking $\succ_i$ over $\mathcal{O}$ (ex: $o_6 \succ o_1 \succ o_4 \succ o_5 \succ o_2 \succ o_3$)

2. **Scoring**: We have a common scoring vector $s = (s_1, \ldots, s_m)$ (with $s_1 \geq \cdots \geq s_m$) mapping each rank to a utility.

3. **Additivity**: These utilities are additive.

4 natural scoring vectors:

<table>
<thead>
<tr>
<th>$\succ_i$</th>
<th>$o_6$</th>
<th>$o_1$</th>
<th>$o_4$</th>
<th>$o_5$</th>
<th>$o_2$</th>
<th>$o_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Borda</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Lexicographic</td>
<td>32</td>
<td>16</td>
<td>8</td>
<td>4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Quasi-Indifference</td>
<td>$1 + \frac{s_1}{M}$</td>
<td>$1 + \frac{s_2}{M}$</td>
<td>$1 + \frac{s_3}{M}$</td>
<td>$1 + \frac{s_4}{M}$</td>
<td>$1 + \frac{s_5}{M}$</td>
<td>$1 + \frac{s_6}{M}$</td>
</tr>
<tr>
<td>$k$-Approval</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Approximability of optimal social welfare
Example

5 objects, 3 agents...

1: $o_1 \succ o_2 \succ o_3 \succ o_4 \succ o_5$

2: $o_4 \succ o_2 \succ o_5 \succ o_1 \succ o_3$

3: $o_1 \succ o_3 \succ o_5 \succ o_4 \succ o_2$
Example

5 objects, 3 agents...

1: $o_1 \succ o_2 \succ o_3 \succ o_4 \succ o_5$

2: $o_4 \succ o_2 \succ o_5 \succ o_1 \succ o_3$

3: $o_1 \succ o_3 \succ o_5 \succ o_4 \succ o_2$

Let’s consider allocation $\pi = \langle \{o_1\}, \{o_4, o_2\}, \{o_3, o_5\} \rangle$. 
Scoring allocation rules

Example

5 objects, 3 agents...

1: \( o_1 \succ o_2 \succ o_3 \succ o_4 \succ o_5 \)

2: \( o_4 \succ o_2 \succ o_5 \succ o_1 \succ o_3 \)

3: \( o_1 \succ o_3 \succ o_5 \succ o_4 \succ o_2 \)

Let's consider allocation \( \pi = \langle \{ o_1 \}, \{ o_4, o_2 \}, \{ o_3, o_5 \} \rangle \).

- **Borda**: \( u_1(\pi) = 5; u_2(\pi) = 5 + 4 = 9; u_3(\pi) = 4 + 3 = 7. \)
- **Lexicographic**: \( u_1(\pi) = 16; u_2(\pi) = 24; u_3(\pi) = 12. \)
- **s-QI**: \( u_1(\pi) = 1 + s_1/M; u_2(\pi) = 2 + s_1 + s_2/M; u_3(\pi) = 2 + s_2 + s_3/M. \)
- **2-approval**: \( u_1(\pi) = 1; u_2(\pi) = 2; u_3(\pi) = 1. \)
Back to our resource allocation problem...
Back to our resource allocation problem...

<table>
<thead>
<tr>
<th>Interpretation:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Borda SF</td>
</tr>
<tr>
<td>Lexicographic SF</td>
</tr>
<tr>
<td>Quasi-Indifference SF</td>
</tr>
<tr>
<td>k-Approval SF</td>
</tr>
</tbody>
</table>
Scoring allocation rules

Back to our resource allocation problem...

Maximize: $\sum_i u_i(\pi)$

<table>
<thead>
<tr>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Borda SF</td>
</tr>
<tr>
<td>Lexicographic SF</td>
</tr>
<tr>
<td>Quasi-Indifference SF</td>
</tr>
<tr>
<td>$k$-Approval SF</td>
</tr>
</tbody>
</table>
Back to our resource allocation problem...

<table>
<thead>
<tr>
<th>Interpretation</th>
<th>Maximize: $\sum_i u_i(\pi)$</th>
<th>$\min_i u_i(\pi)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Borda SF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lexicographic SF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quasi-Indifference SF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$k$-Approval SF</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Approximability of optimal social welfare
Scoring allocation rules

Back to our resource allocation problem...

<table>
<thead>
<tr>
<th>Interpretation</th>
<th>Maximize: $\sum_i u_i(\pi)$</th>
<th>$\min_i u_i(\pi)$</th>
<th>lexicmin($u_1(\pi), \ldots, u_n(\pi)$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Borda SF</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lexicographic SF</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quasi-Indifference SF</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$k$-Approval SF</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Back to our resource allocation problem...

<table>
<thead>
<tr>
<th>Interpretation</th>
<th>( \sum u_i(\pi) )</th>
<th>( \min u_i(\pi) )</th>
<th>( \text{leximin}(u_1(\pi), \ldots, u_n(\pi)) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Borda SF</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lexicographic SF</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quasi-Indifference SF</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( k )-Approval SF</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\( \sim \) 12 positional scoring allocation rules
Scores are not necessarily agents’ utilities.

**Question:** What are we actually optimizing in the end?
How to interpret scores?

Scores are not necessarily agents’ utilities.

**Question:** What are we actually optimizing in the end?

**Two interpretations of scores:**
- Compromise between all agents (domain knowledge, learned, . . . )
Scores are not necessarily agents’ utilities.

**Question:** What are we actually optimizing in the end?

**Two interpretations of scores:**
1. Compromise between all agents (domain knowledge, learned, ...)
2. Perception of the center
Scoring allocation rules

Properties

1. **Separability**: Violated by almost all our rules
2. **Monotonicity**: Satisfied by all our rules
3. **Global monotonicity**: Violated by almost all our rules with strictly decreasing scoring vector
4. **Pos. object monotonicity**: Satisfied by $+$ for $n = 2$, but violated for $n \geq 3$ and strictly decreasing scoring vector

---

What is the precise complexity of these allocation rules?
What is the precise complexity of these allocation rules?

For each pair (scoring vector, social criterion), what is the complexity of...

1. **Optimal Allocation Value (OAV):** is it possible to find an allocation of social welfare $\geq K$?
2. **Optimal Allocation (OA):** does $\pi$ belong to the set of optimal allocations?
3. **Find Optimal Allocation (FOA):** find an optimal allocation.
For $\sum_i u_i(\pi)$ (classical utilitarianism), everything is polynomial!  

**Idea:** give each item to the agent that ranks it the best.
For $\sum_i u_i(\pi)$ (classical utilitarianism), everything is polynomial!

**Idea:** give each item to the agent that ranks it the best.

For $\min_i u_i(\pi)$ (egalitarianism):

- **Bad news:** hard (NP-complete, coNP-complete, NP-hard for OAV, OA, FOA resp.) for Borda, lexicographic and QI scoring vectors.

  (all by reduction from [X3C])
For $\sum_i u_i(\pi)$ (classical utilitarianism), everything is polynomial!

**Idea:** give each item to the agent that ranks it the best.

For $\min_i u_i(\pi)$ (egalitarianism):

- **Bad news:** hard (NP-complete, coNP-complete, NP-hard for OAV, OA, FOA resp.) for Borda, lexicographic and QI scoring vectors. (all by reduction from [X3C])

- **Good news:** easy (polynomial)
  - if the number of objects is fixed (obvious);
  - if the number of agents is fixed (dynamic programming);
  - for $k$-approval ("known" problem).
For $\sum_i u_i(\pi)$ (classical utilitarianism), everything is polynomial!

**Idea:** give each item to the agent that ranks it the best.

For $\min_i u_i(\pi)$ (egalitarianism):
- **Bad news:** hard (NP-complete, coNP-complete, NP-hard for OAV, OA, FOA resp.) for Borda, lexicographic and QI scoring vectors. (all by reduction from [X3C])
- **Good news:** easy (polynomial)
  - if the number of objects is fixed (obvious);
  - if the number of agents is fixed (dynamic programming);
  - for $k$-approval ("known" problem).

Most results for $\min$ carry over to lexicimin.
Most cases are **hard**...

**Question:** *Is it possible to efficiently compute good (but potentially suboptimal) allocations?*
Approximation

Most cases are **hard**...

**Question:** *Is it possible to efficiently compute good (but potentially suboptimal) allocations?*

**Our approach:** Instead of giving general approximation results, we:

- focus on a *simple* allocation protocol [Bouveret and Lang, 2011];
- and try to analyze how good the allocations it gives are.

---

A general elicitation-free protocol for allocating indivisible goods.
In *Proceedings of IJCAI’11*, pages 73–78. IJCAI.
An elicitation-free protocol...

Ask the agents to pick in turn their most preferred object among the remaining ones, according to some predefined sequence $\sigma$.

**Example**

3 agents 1, 2, 3 / 6 objects / sequence 123321 $\rightarrow$ 1 chooses first (and takes her preferred object), then 2, then 3, then 3 again...
Scoring allocation rules

An elicitation-free protocol...

Ask the agents to pick in turn their most preferred object among the remaining ones, according to some predefined sequence $\sigma$.

**Example**

3 agents 1, 2, 3 / 6 objects / sequence 123321 $\rightarrow$ 1 chooses first (and takes her preferred object), then 2, then 3, then 3 again...

Arguably a very simple (and natural) protocol! **No elicitation required!**
Ask the agents to pick in turn their most preferred object among the remaining ones, according to some predefined sequence $\sigma$.

**Example**

3 agents 1, 2, 3 / 6 objects / sequence 123321 $\rightarrow$ 1 chooses first (and takes her preferred object), then 2, then 3, then 3 again...

Arguably a very simple (and natural) protocol! **No elicitation required**!

Here we focus on regular sequences $\sigma$ of the kind $(1 \ldots n)^*$ and **Borda**.
Arguably a very simple (and natural) protocol...
Scoring allocation rules

**Price of elicitation-freeness**

Arguably a very simple (and natural) protocol... 

...but obviously suboptimal!

**Multiplicative Price of Elicitation-Freeness:**
worst case ratio $\frac{sw_{opt}}{sw(\sigma)}$, for a sequence $\sigma$

**Additive Price of Elicitation-Freeness:**
worst case difference $sw_{opt} - sw(\sigma)$, for a sequence $\sigma$
Arguably a very simple (and natural) protocol... 

...but obviously suboptimal!

**Question:** What is the loss in social welfare we incur by using such a simple protocol instead of computing the optimal allocation?
Arguably a very simple (and natural) protocol... ...but obviously suboptimal!

**Question:** What is the loss in social welfare we incur by using such a simple protocol instead of computing the optimal allocation?

**Multiplicative** Price of Elicitation-Freeness:
worst case ratio $sw^\text{opt} / sw(\sigma)$, for a sequence $\sigma$

**Additive** Price of Elicitation-Freeness:
worst case difference $sw^\text{opt} - sw(\sigma)$, for a sequence $\sigma$
Experimental results

For classical utilitarianism ($\sum_i u_i(\pi)$):

![Graph showing suboptimality for classical utilitarianism with objects ranging from 0 to 200.]

For egalitarianism ($\min_i u_i(\pi)$):

![Graph showing suboptimality for egalitarianism with objects ranging from 0 to 50.]

Approximability of optimal social welfare
For $m = kn$ objects,

$$MPEF_+ \geq 1 + \frac{mn - m - n^2 + n}{m^2 + mn}$$

Why is this true?
Lower bounds for MPEF

Classical utilitarianism

For $m = kn$ objects,

$$MPEF_+ \geq 1 + \frac{mn - m - n^2 + n}{m^2 + mn}$$

Why is this true?

1: $o_6 \succ o_1 \succ o_2 \succ o_3 \succ o_4 \succ o_5$

2: $o_5 \succ o_6 \succ o_1 \succ o_2 \succ o_3 \succ o_4$

3: $o_4 \succ o_5 \succ o_6 \succ o_1 \succ o_2 \succ o_3$
Lower bounds for MPEF

Classical utilitarianism

For $m = kn$ objects,

$$MPEF_+ \geq 1 + \frac{mn - m - n^2 + n}{m^2 + mn}$$

Why is this true?

1. $o_6 \succ o_1 \succ o_2 \succ o_3 \succ o_4 \succ o_5$
2. $o_5 \succ o_6 \succ o_1 \succ o_2 \succ o_3 \succ o_4$
3. $o_4 \succ o_5 \succ o_6 \succ o_1 \succ o_2 \succ o_3$
For $m = kn$ objects,

$$MPEF_+ \geq 1 + \frac{mn - m - n^2 + n}{m^2 + mn}$$

Why is this true?

1: $o_6 \succ o_1 \succ o_2 \succ o_3 \succ o_4 \succ o_5$

2: $o_5 \succ o_6 \succ o_1 \succ o_2 \succ o_3 \succ o_4$

3: $o_4 \succ o_5 \succ o_6 \succ o_1 \succ o_2 \succ o_3$
For $m = kn$ objects,

$$MPEF_+ \leq 2 - \frac{m - n}{mn + n}$$
For $m = kn$ objects,

$$MPEF_+ \leq 2 - \frac{m - n}{mn + n}$$

Why is this true?

If at a time step agent $j$ gets object $g_{ni+j}$ we learn

- a lower bound on agent $j$’s value for this object
For $m = kn$ objects,

$$MPEF_+ \leq 2 - \frac{m - n}{mn + n}$$

Why is this true?

If at a time step agent $j$ gets object $g_{ni+j}$ we learn

- a lower bound on agent $j$’s value for this object
- an upper bound on all agents’ values for this object
For $m = kn$ objects,

$$MPEF_+ \leq 2 - \frac{m - n}{mn + n}$$

Why is this true?

If at a time step agent $j$ gets object $g_{ni+j}$ we learn

- a lower bound on agent $j$’s value for this object
- an upper bound on all agents’ values for this object

Corollary:

If $n = 2$ and $m = 2k$,

$$1 + \frac{m - 2}{m(m + 2)} \leq MPEF_+ \leq \frac{3}{2} + \frac{3}{2m + 2}$$
For $m = kn$ objects,

$$MPEF_{\text{min}} \leq \frac{2mn - m + n}{mn + 2n - n^2}$$
For $m = kn$ objects,

$$MPEF_{\min} \leq \frac{2mn - m + n}{mn + 2n - n^2}$$

Why is this true?

- Upper bound $MPEF_{\min}$ using best and worst case profile
For $m = kn$ objects,

$$MPEF_{\text{min}} \leq \frac{2mn - m + n}{mn + 2n - n^2}$$

Why is this true?

- Upper bound $MPEF_{\text{min}}$ using best and worst case profile

**Corollary:**
If $n = 2$ and $m = 2k$,

$$MPEF_{\text{min}} \leq \frac{3}{2} + \frac{5}{m + 4}$$
Experimental results

For classical utilitarianism ($\sum_i u_i(\pi)$):

![Graph showing suboptimality for classical utilitarianism for different values of $n$.]

For egalitarianism ($\min_i u_i(\pi)$):

![Graph showing suboptimality for egalitarianism for different values of $n$.]
For $m = kn$ objects,

\[
\frac{(n-1)(m-n)}{2} \leq APEF_+ \leq \frac{(m-n)(mn - m + n^2 + n)}{2n}
\]

\[
APEF_{\text{min}} \leq \frac{m^2n - mn - m^2 + mn^2}{2n^2}
\]
Future Work

- Closing gaps (upper and lower bounds)
- Relationships (rank weighted utilitarianism, inequality indices)
- Exact characterizations, manipulation (scoring allocation rules)
On finding rankings with approximately optimal Kemeny score:

An approximation algorithm for a voting rule is, in effect, a different voting rule; and in real-world elections, voters may feel deceived if a different voting rule is used than the one that was promised to them.

On finding rankings with approximately optimal Kemeny score:

An approximation algorithm for a voting rule is, in effect, a different voting rule; and in real-world elections, voters may feel deceived if a different voting rule is used than the one that was promised to them.


Does this argument hold in the resource allocation setting as well?
On finding rankings with approximately optimal Kemeny score:

An approximation algorithm for a voting rule is, in effect, a different voting rule; and in real-world elections, voters may feel deceived if a different voting rule is used than the one that was promised to them.


Does this argument hold in the resource allocation setting as well?
References


