Voting with partial orders

Lirong Xia\textsuperscript{1}  Vincent Conitzer\textsuperscript{1}  Jérôme Lang\textsuperscript{2}  Mingsheng Ying \textsuperscript{3}

\textsuperscript{1}Department of Computer Science  
Duke University  
Durham, NC, USA

\textsuperscript{2}IRIT  
Université Paul Sabatier  
Toulouse, France

\textsuperscript{3}DCST  
Tsinghua University  
Beijing, China

Outline

1. Introduction
2. On the neutrality and efficiency of decomposable voting rules
3. Computing possible/necessary unique/co-winner
4. Summary
Outline

1 Introduction

2 On the neutrality and efficiency of decomposable voting rules

3 Computing possible/necessary unique/co-winner

4 Summary
Basic setting of voting

- **Candidates**: A set of candidates $\mathcal{X}$.
- **Voters**: A set of voters $\mathcal{A}$.
- **Preference**: $\mathcal{D}_\mathcal{X}$, the preference structure over candidate set $\mathcal{X}$ (linear order, subset, etc).
- **Profile**: $\mathcal{D}_\mathcal{X}^{\mathcal{A}}$, each voter picks a vote from the preference structure $\mathcal{D}_\mathcal{X}$.
- **Collective decision**: A mapping from $\mathcal{D}_\mathcal{X}^{\mathcal{A}}$ to specific range:
  - **Rule**: $r : \mathcal{D}_\mathcal{X}^{\mathcal{A}} \rightarrow \mathcal{X}$.
  - **Correspondence**: $c : \mathcal{D}_\mathcal{X}^{\mathcal{A}} \rightarrow 2^\mathcal{X} \setminus \emptyset$. 
Why partial orders?

- Voters themselves might not be able/willing to give linear order
- Space/ Time consuming

Take partial orders as preferences
CP-net $\mathcal{N}$ [C. Boutilier et al 99]

- Set of variables $I = \{x_1, \ldots, x_p\}$, taking values in $D_1, \ldots, D_p$
  
  Combinatorial domain $\mathcal{X} = D_1 \times \ldots \times D_p$

- Directed acyclic graph $G = (I, E)$
- CPT for each $x_i$ indicating the conditional preference on $D_i$

A CP-net naturally induces a partial order over $\mathcal{X} = D_1 \times \ldots \times D_p$. 
CP-net: an example

The partial order $\mathcal{N}$ induces is

$$xyz \leftrightarrow x\bar{y}\bar{z} \leftrightarrow x\bar{y}z \rightarrow \bar{x}\bar{y}z \rightarrow \bar{x}yz \rightarrow \bar{x}y\bar{z}$$
Extension of partial order

A linear order $V$ on $X$ is said to

- extend a partial order $P$, if $P \subseteq V$. We say $V$ is an extension (completion) of $P$.
- extend a CP-net $\mathcal{N}$ over $X$, if $V$ extends the partial order that $\mathcal{N}$ induces.
- be compatible with a linear order $O$ on $I$, if it extends a CP-net that is compatible with $O$. 

Voting with partial orders
Voting on combinatorial domain with preferences modeled by CP-nets [J.Lang 07]

- A set of issues \( I = \{x_1, \ldots, x_p\} \) on \( X = D_1 \times \ldots \times D_p \).
- A linear order \( O \) over \( I \), for example \( O = x_1 > \ldots > x_p \).
- \( p \) local voting rules \( r_1, \ldots, r_p \).
- Input: A profile \( P = (V_1, \ldots, V_N) \) s.t. \( V_j \) is compatible with \( O \).
Output: \((d_1, \ldots, d_p) \in \mathcal{X}\) through a \(p\)-step process

1. Select \(d_1\) by \(r_1\) from \(P|_{x_1}\).
2. Select \(d_2\) by \(r_2\) from \(P|_{x_2|x_1=d_1}\).
   
   \[\vdots\]
3. \(p\). Select \(d_p\) by \(r_p\) from \(P|_{x_p|x_1=d_1,\ldots,x_{p-1}=d_{p-1}}\).

Such a rule is defined to be the *sequential composition* of \(r_1, \ldots, r_p\), denoted by \(\text{Seq}(r_1, \ldots, r_p)\). It is said to be *decomposable*.

Special case: Seat-by-seat voting: No edges in CP-net, all issues are voted on separately.
Output: \((d_1, \ldots, d_p) \in \mathcal{X}\) through a \(p\)-step process

1. Select \(d_1\) by \(r_1\) from \(P|_{x_1}\).
2. Select \(d_2\) by \(r_2\) from \(P|_{x_2|x_1=d_1}\).
   
   \[\vdots\]

\(p\). Select \(d_p\) by \(r_p\) from \(P|_{x_p|x_1=d_1, \ldots, x_{p-1}=d_{p-1}}\).

Such a rule is defined to be the sequential composition of \(r_1, \ldots, r_p\), denoted by \(\text{Seq}(r_1, \ldots, r_p)\). It is said to be decomposable.

Special case: Seat-by-seat voting: No edges in CP-net, all issues are voted on separately.
Output: $(d_1, \ldots, d_p) \in \mathcal{X}$ through a $p$-step process

1. Select $d_1$ by $r_1$ from $P_{x_1}$.
2. Select $d_2$ by $r_2$ from $P_{x_2|x_1=d_1}$.

\vdots

p. Select $d_p$ by $r_p$ from $P_{x_p|x_1=d_1, \ldots, x_{p-1}=d_{p-1}}$.

Such a rule is defined to be the sequential composition of $r_1, \ldots, r_p$, denoted by $\text{Seq}(r_1, \ldots, r_p)$. It is said to be decomposable.

Special case: Seat-by-seat voting: No edges in CP-net, all issues are voted on separately.
Output: \((d_1, \ldots, d_p) \in \mathcal{X}\) through a \(p\)-step process

1. Select \(d_1\) by \(r_1\) from \(P|_{x_1}\).

2. Select \(d_2\) by \(r_2\) from \(P|_{x_2|x_1=d_1}\).

\[\vdots\]

p. Select \(d_p\) by \(r_p\) from \(P|_{x_p|x_1=d_1, \ldots, x_{p-1}=d_{p-1}}\).

Such a rule is defined to be the \textit{sequential composition} of \(r_1, \ldots, r_p\), denoted by \(\text{Seq}(r_1, \ldots, r_p)\). It is said to be \textit{decomposable}.

Special case: Seat-by-seat voting: No edges in CP-net, all issues are voted on separately.
Output: \((d_1, \ldots, d_p) \in \mathcal{X}\) through a \(p\)-step process

1. Select \(d_1\) by \(r_1\) from \(P|_{x_1}\).
2. Select \(d_2\) by \(r_2\) from \(P|_{x_2|x_1=d_1}\).

\[
\vdots
\]

\(p\). Select \(d_p\) by \(r_p\) from \(P|_{x_p|x_1=d_1,\ldots,x_{p-1}=d_{p-1}}\).

Such a rule is defined to be the \textit{sequential composition} of \(r_1, \ldots, r_p\), denoted by \(\text{Seq}(r_1, \ldots, r_p)\). It is said to be \textit{decomposable}.

Special case: Seat-by-seat voting: No edges in CP-net, all issues are voted on separately.
Outline

1. Introduction

2. On the neutrality and efficiency of decomposable voting rules

3. Computing possible/necessary unique/co-winner

4. Summary
Question: Is there any neutral or Pareto efficient decomposable voting rules or correspondences?

- **Neutrality**: the voting rule is insensitive to any permutation of candidates
- **Pareto efficiency**: for any two candidates $c_1, c_2$, if $c_1$ is preferred to $c_2$ by all voters, then $c_2$ cannot be the winner
1. If the domain is not the product of two binary issues, then the only Pareto efficient seat-by-seat voting rule is a dictatorship.
   i.e. \( \{a_1, b_1\} \times \{a_2, b_2, c_2\} \).

2. If there are three or more issues, and each local rule satisfies efficiency, then the only neutral seat-by-seat voting rule is a dictatorship.
   i.e. \( \{a_1, b_1\} \times \{a_2, b_2\} \times \{a_3, b_3\} \).
Our impossibility theorems

1. If the domain is not the product of two binary issues, then the only Pareto efficient seat-by-seat voting rule or correspondence is a dictatorship or a trivial one.

2. If the domain is not the product of two binary issues, then the only neutral seat-by-seat voting rule or correspondence is a dictatorship, or an anti-dictatorship, or a trivial one.
More on neutrality and Pareto efficiency

- Sequential composition of two plurality rules on two binary issues is neutral and Pareto efficient

- The theorems can be easily extended to sequential voting rules/correspondences

---

1 L.Xia, J. Lang, M. Ying 07.
Outline

1. Introduction
2. On the neutrality and efficiency of decomposable voting rules
3. Computing possible/necessary unique/co-winner
4. Summary
Possible (necessary) winner, [Konczak and Lang, 2005]

Definition

A candidate $c$ is a possible (necessary) winner for a set of partial orders $P$, w.r.t. voting rule $r$, if there exists a (for any) set of linear orders $P'$ extending $P$, $r(P') = c$.

Need to distinguish between unique winner and co-winner ($c \in r(P')$).
Example: Possible (unique) winner under Plurality rule

Preferences:

- $V_1$: $a < b < c$
- $V_2$: $a > b > c$
- $V_3$: $a > b$

$a$ is a possible (unique) winner:

- $V_1$: $a < b < c$
- $V_2$: $a > b > c$
- $V_3$: $a > b$

$a$ is not a necessary winner:

- $V_1$: $a < b < c$
- $V_2$: $a > b > c$
- $V_3$: $a > b$

L. Xia, V. Conitzer, J. Lang, M. Ying

Duke University, IRIT, Tsinghua University

Voting with partial orders
Example: Necessary (unique) winner

Preferences:

\[
\begin{array}{c}
V_1 \\
\begin{array}{c}
V_2 \\
\begin{array}{c}
V_3 \\
\end{array}
\end{array}
\end{array}
\]

Candidate \( a \) will always win!
Possible/necessary unique/co-winner determination

Input:
1. A voting rule $r$
2. A candidate $c$
3. A set of partial orders $P$.

Question: is $c$ is a possible/necessary unique/co-winner?
The votes are unweighted.
## Complexity results

<table>
<thead>
<tr>
<th>Voting rule</th>
<th>Possible winner</th>
<th>Necessary winner</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scoring$^1$</td>
<td>NP-hard</td>
<td>$O(nm^2)$</td>
</tr>
<tr>
<td>STV</td>
<td>NP-hard$^2$</td>
<td>coNP-hard$^2$</td>
</tr>
<tr>
<td>Copeland$^1$</td>
<td>NP-hard</td>
<td>coNP-hard</td>
</tr>
<tr>
<td>Maximin$^1$</td>
<td>NP-hard</td>
<td>$O(nm^3)$</td>
</tr>
<tr>
<td>Bucklin$^1$</td>
<td>NP-hard</td>
<td>$O(nm^2)$</td>
</tr>
<tr>
<td>Ranked pairs$^1$</td>
<td>NP-hard</td>
<td>coNP-hard</td>
</tr>
</tbody>
</table>

**n**: # votes  
**m**: # candidates

- Holds for both unique/co-winner.

---

$^1$ Even when the # incomparable (unknown) pairs in each vote is small (less than 16)  
$^2$ Even when the partial orders are modeled by CP-nets.
Summary

- Impossibility theorem about seat-by-seat voting
- Complexity of computing possible/necessary unique/co-winners given partial orders

Future work

- Possible/necessary unique/co-winners when inputs are CP-nets
- Is it usually easy to compute possible/necessary unique/co-winners