Complexity Results in Optimistic/Pessimistic Preference Reasoning

ICTAI, November 2016, San Jose

Christian Bessiere, Remi Coletta, Gaelle Hisler and Anastasia Paparrizou
Summary

• Background in preference reasoning
  – Semantics
  – Main problems

• Contributions
  – Complexity study of main problems in preference reasoning
  – Knowledge compilation: improving through pre-processing
How express a preference?

Suppose the preference « I prefer salad to tomato »

• Quantitative preference
  – “I like a salad with weight 0.7 and tomato with weight 0.3”

• Qualitative preference
  – “I prefer salad rather than tomato”
Main problems

• Undominated
  – “Does it exist an outcome which is the most preferred?”

• Dominance
  – “Given two outcomes $\omega, \omega'$, is $\omega$ strictly preferred as $\omega'$?”

• Consistency
  – “Is the network consistent?” (absence of dominance cycle)
## Complexity Results in Optimistic/Pessimistic Preference Reasoning – G. Hisler

### Formalisms - Complexity

<table>
<thead>
<tr>
<th></th>
<th>Dominance</th>
<th>Consistency</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Conditionnal Logic[2]</strong></td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>
Semantics — How to interpret a preference?

Suppose the preference « I prefer salad to tomato »

1: Optimistic  
[Pearl, 1990]

at least one salad meal is strictly preferred to all tomato meals

2: Strong (called strict)  
[Boutilier, 1994]

all salad meals are strictly preferred to all tomato meals regardless main dish / wine

3: Ceteris paribus  
[Hansson, 1996]

Salad meals are strictly preferred to tomato meals all other things being equal

4: Pessimistic  
[Benferhat et al., 2002]

at least one tomato meal is less preferred to all salad meals
Choosing a meal according to

Starter = \{ \text{tomato, lettuce} \}

Main dish = \{ \text{steak, fish} \}

Wine = \{ \text{red, white wine} \}

• An outcome $\omega$ is a complete assignment

• The set of all outcomes $\Omega$
Framework – Conditionnal logic formalism

\[ p_1 = \text{Optimistic} > \text{Pessimistic} \]
\[ p_2 = \text{Pessimistic} > \text{Optimistic} \]
\[ p_3 = \text{Optimistic Strong=}p_1 \]

Semantic = Optimistic
Layer_0

Pessimistic
Layer_1

Optimistic
Layer_2

Complexity Results in Optimistic/Pessimistic Preference Reasoning – G.Hisler
State of the art - Conditionnal logic formalism

Layers are explicitly built:

Algorithm Exponential in space

• Undominated
Choose an outcome in Layer_0

• Dominance
Layer _0 < Layer _1

Yes

• Consistency
Check that Layer_0 ∪ ... ∪ Layer_n = Ω
## Contributions - Complexity map

<table>
<thead>
<tr>
<th>Problem</th>
<th>Strong Preferences</th>
<th>Optimistic</th>
<th>Pessimistic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Undominated</td>
<td>All</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Not All</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Consistency</td>
<td>All</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Not All</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dominance</td>
<td>All</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Not All</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Contributions - Complexity map

<table>
<thead>
<tr>
<th>Problem</th>
<th>Strong Preferences</th>
<th>Optimistic</th>
<th>Pessimistic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Undominated</td>
<td>All</td>
<td>NP-complete</td>
<td>Polynomial</td>
</tr>
<tr>
<td></td>
<td>Not All</td>
<td>NP-complete</td>
<td>NP-complete</td>
</tr>
<tr>
<td>Consistency</td>
<td>All</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Not All</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dominance</td>
<td>All</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Not All</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Complexity depends on the semantic**
### Contributions - Complexity map

<table>
<thead>
<tr>
<th>Problem</th>
<th>Strong Preferences</th>
<th>Optimistic</th>
<th>Pessimistic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Undominated</td>
<td>All</td>
<td>NP-complete</td>
<td>Polynomial</td>
</tr>
<tr>
<td></td>
<td>Not All</td>
<td>NP-complete</td>
<td>NP-complete</td>
</tr>
<tr>
<td>Consistency</td>
<td>All</td>
<td>Polynomial</td>
<td>Polynomial</td>
</tr>
<tr>
<td></td>
<td>Not All</td>
<td>NP-complete</td>
<td>NP-complete</td>
</tr>
<tr>
<td>Dominance</td>
<td>All</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Not All</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Complexity depends if all preferences are strong

Complexity Results in Optimistic/Pessimistic Preference Reasoning – G.Hisler
### Contributions - Complexity map

<table>
<thead>
<tr>
<th>Problem</th>
<th>Strong Preferences</th>
<th>Optimistic</th>
<th>Pessimistic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Undominated</td>
<td>All</td>
<td>NP-complete</td>
<td>Polynomial</td>
</tr>
<tr>
<td></td>
<td>Not All</td>
<td>NP-complete</td>
<td>NP-complete</td>
</tr>
<tr>
<td>Consistency</td>
<td>All</td>
<td>Polynomial</td>
<td>Polynomial</td>
</tr>
<tr>
<td></td>
<td>Not All</td>
<td>NP-complete</td>
<td>NP-complete</td>
</tr>
<tr>
<td>Dominance</td>
<td>All</td>
<td>Polynomial</td>
<td>Polynomial</td>
</tr>
<tr>
<td></td>
<td>Not All</td>
<td>DP-complete</td>
<td>DP-complete</td>
</tr>
</tbody>
</table>

- Dominance is the only problem called several times
  - Many different pairs \((\omega, \omega')\)
  - Unfortunately DP-complete
Knowledge compilation - Background

Take as much time as needed

Q?\[\rightarrow\]

Data Structure DS

|DS'| = Polynomial |DS|

Q?\[\rightarrow\]

Data Structure DS'

Complexity of Q? : Hard

Complexity of Q? : Easier

Complexity Results in Optimistic/Pessimistic Preference Reasoning – G. Hisler
Knowledge compilation - Dominance

\[ \omega >? \omega' \]

DP-complete

\[ \langle P_0, \ldots, P_n \rangle \]

Polynomial
Compiled preferences - Equivalence

- Outcomes in Layer$_i$ are those which:
  - satisfy all preferences in $P_i$
  - violate at least one preference in $P_{i-1} \setminus P_i$

$P = P_0 = \{ p_1, p_2, p_3 \}$

Layer$_0$: $p_1p_2p_3$

Layer$_1$: $p_1p_2$

Layer$_2$: $\emptyset$

Optimistic Strong = $p_1$

Satisfaction:
- $p_1$
- $p_2$
- $p_3$

Complexity Results in Optimistic/Pessimistic Preference Reasoning – G. Hisler
Compiled preferences - Complexity

\[
\begin{align*}
\mathbf{P} = & P_0 \quad \begin{array}{ccc} p_1 & p_2 & p_3 \end{array} \\
= & \begin{array}{c} p_1 \end{array} \\
= & P_1
\end{align*}
\]

Optimistic Strong = p_1

Satisfaction

\[
\begin{align*}
p_1 & > \\
p_2 & > \\
p_3 & >
\end{align*}
\]

Full Satisfaction

\[
\begin{align*}
p_1 & \lor p_2 & \lor p_3 \\
p_1 & \land p_2 & \land p_3
\end{align*}
\]

Deactivation of p_1

\[
\{\text{fully satisfaction } p_1\} = \{\text{satisfaction } p_1 \text{ and } p_2 \text{ and } p_3\}
\]

The deactivation of a strong preference is Polynomial

Complexity Results in Optimistic/Pessimistic Preference Reasoning – G. Hisler
Compiled preferences - Complexity

\[ p_1 = \text{Food} \succ \text{Drink} \]
\[ p_2 = \text{Drink} \succ \text{Food} \]
\[ p_3 = \text{Fish} \succ \text{Drink} \]

Optimistic Strong = p1

\[ P = P_0 \]
\[ P_1 \]
\[ P_2 \]

Satisfaction
\[ p_1 \text{ } \text{Food} \]
\[ p_2 \text{ } \text{Drink} \]
\[ p_3 \text{ } \text{Fish} \]

Full Satisfaction
\[ p_1 \text{ } \text{Food} \]
\[ p_2 \text{ } \text{Drink} \]
\[ p_3 \text{ } \text{Fish} \]

Deactivation of \( p_2 \)
\[ \{ \text{fully sat } p_2 \} \cap \{ \text{sat } p_1 \text{ and } p_2 \text{ and } p_3 \} \neq \emptyset \]

\[ P_1 \cap P_2 = P_0 \]

\( p_2 \) is deactivated

The deactivation of a preference is NP-complete

Complexity Results in Optimistic/Pessimistic Preference Reasoning – G. Hisler
Contributions: Dominance($<P_0,\ldots,P_n>$) is Polynomial

$\mathbf{p}_1 = \text{Cabbage} >$  
$\mathbf{p}_2 = \text{Steak} : \text{Wine} >$  
$\mathbf{p}_3 = \text{Fish} : \text{Wine} >$

Optimistic Strong = $\mathbf{p}_1$

Layer 1

$P = P_0$

Layer 2

$P_1$

$\emptyset$

Layer 1 = Layer 1 $< \text{Layer 2}$

*Linear with an index encoding

Complexity Results in Optimistic/Pessimistic Preference Reasoning – G. Hisler
• Complexity study of main problems in preference reasoning in an existing framework
  – Depends on the semantic $S$
  – Depends on the set of strong preferences $Str$

• Dominance is compilable to polynomial time
Thank you for your attention