Computational Social Choice Theory

Tomasz Michalak
tomasz@liv.ac.uk

25.10.2010
Plan

Part I: What is Computational Social Choice?
- Computational Social Choice among other sciences
- Why it is worthwhile to study Computational Social Choice

Part II: Sample Computational Social Choice problems

Part III: Representing coalitional games - example
Part I: What is Computational Social Choice?
What is Social Choice Theory?

Social Choice Theory is a science about aggregating individual preferences in order to make collective decisions.

Sample questions:
- How to make "good" joint decision?
- How to design a fair voting protocol?
- What is a fair division of a cake?

Unsatisfactory answers: designing optimal decision-making protocols is impossible
What is Computational Social Choice?

Novel area in which research problems of Social Choice Theory are analyzed from the *computational perspective*.

Sample questions (Ulle Endriss):
- What if the number of alternatives is very large or has a combinatorial structure? knowledge representation
- Can we make sure that the individual agents are going to report their true preferences? game theory & complexity theory
- Once we have settled on a particular mechanism, how can we execute it in a reasonable amount of time? algorithm design
What is Computational Social Choice?

Computational Social Choice combines (Ulle Endris):

1. **Computer science** - algorithmic theory, complexity theory, knowledge representation, logic, artificial intelligence, multi-agent systems.

2. **Economics** - social choice theory, game theory, microeconomics.

Different names for virtually the same phenomenon: Algorithmic Game Theory, Algorithmic Decision Theory, Social Software.
Related courses at other universities

- Ulle Endriss’s course Computational Social Choice at University of Amsterdam
- Toumas Sandholm’s course Foundation of Electronic Market Places at Carnegie Mellon University
- Vincent Conitzer’s course Computational Game Theory and Mechanism Design at Duke University
- David Parkes’s course Computational Mechanism Design at Harvard
- Al Roth and Peter Coles’s course Market Design at Harvard
- Noam Nisan’s course at Hebrew University
Related courses at other universities

- Utku Ünver’s course Matching Market Design at University of Pittsburgh
- Robert Wilson’s course Market Design at Stanford
- Tim Roughgarden’s course Introduction to Algorithmic Game Theory at Stanford
- Christos Papadimitriou’s course at University of California, Berkeley
- Amy Greenwald’s course at Brown University
- Yoav Shoham’s course Computer Science and Game Theory at Stanford
Part II: Sample problems in Computational Social Choice
Voting Paradox

Rationality requires that preferences over alternatives are \textit{transitive}.

\textbf{Irrational:} \( A \succ B \) and \( B \succ C \) and \( C \succ A \). Here preferences are \textit{cyclic, i.e. non-transitive}.

\textbf{Rational:} \( A \succ B \) and \( B \succ C \) and \( A \succ C \). Here preferences are \textit{acyclic, i.e. transitive}.

In the late 18th century, Marquis de Condorcet noted that:

\begin{itemize}
  \item there is a group of individual \textbf{rational} voters (\textit{i.e.} with acyclic preferences);
  \item nevertheless, collective preferences can be \textbf{irrational} (\textit{i.e.} cyclic).
\end{itemize}
Voting Paradox – Cntd.

*Majority rule* – is a decision rule that selects one of two alternatives, based on which has more than half the votes. Consider the following example of 3 voters and 3 alternatives $A$, $B$ and $C$.

- Agent 1: $A ≻ B ≻ C$
- Agent 2: $B ≻ C ≻ A$
- Agent 3: $C ≻ A ≻ B$

Results:

- A majority (agents 1 and 3) prefers $A$ over $B$; AND
- A majority (agents 1 and 2) also prefers $B$ over $C$; AND
- A majority (agents 2 and 3) prefers $C$ over $A$.

Thus, the requirement of majority rule then provides no clear winner. So the “social preference ordering” induced by the seemingly natural majority rule fails. It is not transitive, so it is *irrational.*
Vote Manipulation

(Example from Ulle Endriss's lecture Computational Social Choice, U. of Amsterdam)

**Plurality rule** - in this voting system each voter has one vote. In simple plurality, the winning alternative is that one which had most votes. There is no requirement that the winner gained an absolute majority of votes, *etc.* Various forms of plurality rule are used in most real-world situations.

Assume the preferences of the people in, say, Florida are as follows:

- 49%: *Bush* ≻ *Gore* ≻ *Nader*
- 20%: *Gore* ≻ *Nader* ≻ *Bush*
- 20%: *Gore* ≻ *Bush* ≻ *Nader*
- 11%: *Nader* ≻ *Gore* ≻ *Bush*

So even if nobody is cheating, Bush will win in a plurality contest.  
**Issue I**: In a pairwise competition, Gore would have defeated anyone.  
**Issue II**: It would have been in the interest of the Nader supporters to manipulate, *i.e.* to misrepresent their preferences.
Other Sample Issues

Other sample issues in Computational Social Choice:

- **ELECTING A COMMITTEE** - suppose we are supposed to elect a committee of five students from the class. There are 20 students and $\binom{20}{5} = 15504$. The domain of alternatives has a combinatorial structure. Should we report the entire preference function? Is it reasonable from the voters’ point of view? Is it going to be computationally tractable to calculate the outcome?

- **MULTIAGENT RESOURCE ALLOCATION** - there is a system of autonomous agents that bargain for an allocation of a set of tasks. Each agent has individual preferences and tries to influence the final outcome. The overview of sample applications can be found in Chevaleyre et al. Issues in Multiagent Resource Allocation, Informatica, 30:3–31, 2006. These include:
  - Industrial procurement
  - Earth observation satellites
  - Modern manufacturing systems
  - Grid computing
Other Sample Issues – Cntd.

- **Simple auctions**: English, Dutch, etc.
- **Combinatorial auctions**: Bundles, not single items are auctioned.
- **Cake-cutting problem**: also known as fair division problem is the problem. How to divide a resource so that all parties involved feel that they received a fair amount? Sometimes it can be impossible.
- **Preference elicitation**: how to elicit preference from an agent so they are accurate, there are no hidden preferences and redundancies are avoided. It is a key issue for any reliable decision support system.
Combinatorial auctions

- Extremely useful when multiple interrelated goods are auctioned. Example: car and tyres, "overvalued" Sony Vaio and "overpriced" Dell;
- Substitution and complementarity property of goods;

Assume that \( n \) goods are being sold and that there are \( m \) bidders. You are one of the bidders. For how many different bundles can you bid? How many is it for \( n = 30 \)?
Combinatorial auctions

- Extremely useful when multiple interrelated goods are auctioned. Example: car and tyres, ”overvalued” Sony Vaio and ”overpriced” Dell;
- Substitution and complementarity property of goods;

Assume that $n$ goods are being sold and that there are $m$ bidders. You are one of the bidders.
For how many different bundles can you bid? $2^n - 1$
How many is it for $n = 30$? 1,073,741,823
Combinatorial auctions - applications

- Auctions for landing time slots at the airport;
- Auctions for TV programs sold to cable operators;
- Supply chain management;
- Equity markets;
- Wholesale electricity markets;
- Bandwidth markets;
- Pollution right auctions;
Interesting literature introducing basic concepts of Computational Social Choice:

Applications of Computational Social Choice

- Electronic commerce
- Military (BEA Systems, decentralised systems, etc.)
- Medicine - organ exchanges

<table>
<thead>
<tr>
<th>Ebay</th>
<th>Allegro</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google</td>
<td>Yahoo</td>
</tr>
<tr>
<td>AllGovernments</td>
<td>Almost all public institutions</td>
</tr>
</tbody>
</table>