Computational social choice
Combinatorial voting

Lirong Xia

Sep 19, 2013
Miscellaneous

• Report your preferences over papers by 9/30 via email! Then
  – meeting 1: before making slides
  – meeting 2: after making the slides
• Start to think about the topic for project
• Homework
  – e-version is preferred
    • but please write your name in the first page
  – write proofs in full detail: if there is a calculation, show the calculation
  – ask questions on piazza
Last class: the easy-to-compute axiom

• We hope that the outcome of a social choice mechanism can be computed in p-time
  – P: positional scoring rules, maximin, Copeland, ranked pairs, etc
  – NP-hard: Kemeny, Slater, Dodgson

• But sometimes P is not enough
  – input size: $nm \log m$
  – preference representation: ask a human to give a full ranking over 2000 alternatives
  – preference aggregation
In California, voters voted on 11 binary issues (✓/✗)
- \(2^{11} = 2048\) combinations in total
- \(5/11\) are about budget and taxes

Today: Combinatorial voting

- Prop.30 Increase sales and some income tax for education
- Prop.38 Increase income tax on almost everyone for education
Combinatorial domains (Multi-issue domains)

• The set of alternatives can be uniquely characterized by multiple issues

• Let $I=\{x_1,...,x_p\}$ be the set of $p$ issues

• Let $D_i$ be the set of values that the $i$-th issue can take, then $A=D_1 \times ... \times D_p$

• Example:
  - Issues={ Main course, Wine }
  - Alternatives={ } $\times$ {  }
Potential problems

• Preference representation
• Communication
• Preference aggregation
• Which one do you think is the most serious problem?
Where is the bottleneck?

- **Ballot propositions**
  - preference representation: big problem
    - rank 2000 alternatives
  - communication: not a big problem
    - internet is fast and almost free for use
  - Computation: not a big problem
    - computers can easily handle 2000 alternatives
Where is the bottleneck?

• Robots on Mars
  – preference representation: sometimes not a big problem
    • robots can come up a ranking over millions of alternatives
  – communication: big problem
  – computation: sometimes not a big problem
Where is the bottleneck?

- Use a compact representation
  - preference representation: a big problem
    - tradeoff between efficiency and expressiveness
  - communication: not a problem
  - computation: a big problem
    - many voting rules becomes NP-hard to compute
Econ vs. CS in Combinatorial voting

<table>
<thead>
<tr>
<th>Combinatorial voting</th>
<th>Economics</th>
<th>CS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Representation</td>
<td>one value per issue</td>
<td>CP-nets</td>
</tr>
<tr>
<td>Aggregation</td>
<td>issue-by-issue voting</td>
<td>sequential voting</td>
</tr>
<tr>
<td>Evaluation</td>
<td>paradoxes</td>
<td>“numerical” paradoxes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>satisfiability of axioms</td>
</tr>
<tr>
<td>Strategic behavior</td>
<td>equilibrium analysis</td>
<td>evaluation of equilibrium outcome</td>
</tr>
</tbody>
</table>
Issue-by-issue voting

- Issue-by-issue voting (binary variables)
  - representation: each voter marks one value for each issue
  - similar to the plurality rule
  - for each issue, use the majority rule to decide the winner

Alice

Bob

Carol
Computational aspects of issue-by-issue voting

• Language
  – one value per issue
  – $\sum_i \log |D_i|$  

• Low communication

• Fast computation
Social choice aspects of issue-by-issue voting

• Representation
  – agents are likely to feel uncomfortable with reporting unconditional preferences

• Hard to analyze
  – not clear what an agent will report

• Outcome is sometimes extremely bad
  – multiple-election paradoxes
    • winner ranked in the bottom
    • winner is not Pareto optimal

• No issue-by-issue voting rule satisfies neutrality or Pareto efficient [Benoit & Kornhauser GEB-10]
  – If the domain is not composed of two binary issues

• Strategic aspects: [Ahn & Oliveros Econometrica-12]
Separable preferences

- Agents are comfortable reporting their preferences when these preferences are **separable**
  - for any issue $i$, any agent’s preferences over issue $i$ does not depend on the value of other issues
  - for any agent $j$, any $a_i, b_i \in D_i$ and any $c_{-i}, d_{-i} \in D_{-i}$,
    
    \[(a_i, c_{-i}) >_j (b_i, c_{-i}) \text{ if and only if } (a_i, d_{-i}) >_j (b_i, d_{-i})\]
Sequential voting [Lang IJCAI-07]

• Given
  – an order over issues, w.l.o.g. \( x_1 \rightarrow \ldots \rightarrow x_k \)
  – \( k \) local rules \( r_1, \ldots, r_k \)

  \( r_j \) is a social choice mechanism for \( x_j \)
Seems better, but

- Practically: hard to have all agents vote for \( p \) times
- Theoretically: How to formally analyze this process?
  - are agents more comfortable?
  - any multiple-election paradoxes?
  - axiomatic properties?
Preference representation: CP-nets
[Boutilier et al. JAIR-04]

Variables: $x, y, z$. $D_x = \{x, \bar{x}\}, D_y = \{y, \bar{y}\}, D_z = \{z, \bar{z}\}$.

This CP-net encodes the following partial order:

$x \succ \bar{x}$
$x : y \succ \bar{y}$
$\bar{x} : \bar{y} \succ y$

$CPT(x)$  $CPT(y)$  $CPT(z)$

$x : z \succ \bar{z}$
$x \bar{y} : z \succ \bar{z}$
$\bar{x} : \bar{y} \succ z$
$\bar{x}y : \bar{z} \succ z$

Graph

<table>
<thead>
<tr>
<th>CPTs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

This CP-net encodes the following partial order:

$xyz \leftarrow x\bar{y}z$  $x\bar{y}z \rightarrow \bar{x}\bar{y}z \rightarrow \bar{x}yz \rightarrow \bar{xyz} \rightarrow \bar{x}y\bar{z}$
Sequential voting under CP-nets

- Issues: main course, wine
- Order: main course > wine
  - agents’ CP-nets are compatible with this order
- Local rules are majority rules
- \( V_1 \): 
- \( V_2 \): 
- \( V_3 \): 
- Step 1: 
- Step 2: given 
  - is the winner for wine
- Winner: ( , )
Computational aspects of sequential voting

- More flexible
  - separable preferences are a special case (CP-nets with no edges)
- Language
  - CP-nets
  - CPT for $x_i$: $2^{\text{#parents of } x_i} \log |D_i|$
  - Total: $\sum_i 2^{\text{#parents of } x_i} \log |D_i|$
- Low-high communication
- Fast computation
Social choice aspects of sequential voting

- **Representation**
  - agents feel more comfortable than using issue-by-issue voting

- **Easier to analyze**

- **Outcome is sometimes very bad, but better than issue-by-issue voting**
  - *multiple-election paradoxes* when agents’ preferences are represented by CP-nets compatible with the same order
    - winner ranked almost in the bottom
    - winner is not Pareto optimal

- **No sequential voting rule satisfies neutrality or Pareto efficient** [Xia&Lang IJCAI-09]
  - If the domain is not composed of two binary issues
  - Strategic behavior: next
Other social choice axioms?

- Depends on whether “local” rules satisfy the property [LX MSS-09, CLX IJCAI-11]
  - E.g., the sequential rule satisfies anonymity ⇔ all local rules satisfy anonymity

<table>
<thead>
<tr>
<th>Axiom</th>
<th>Global to local</th>
<th>Local to global</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anonymity</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Monotonicity</td>
<td>Only last local rule</td>
<td>Only last local rule</td>
</tr>
<tr>
<td>Consistency</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Participation</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Strong monotonicity</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>

- Other axioms: open
Bottom line

- Design the language for your application
  - other languages: GAI networks, soft constraints, TCP nets
    - cf combinatorial auctions
  - coding theory may help

Computational efficiency

Tradeoff

Expressiveness
Strategic agents

• Do we need to worry about agents’ strategic behavior?
  – Manipulation, bribery, agenda control…

• Evaluate the effect of strategic behavior
  – Game theory
  – Price of anarchy [KP STACS-99]

  Optimal truthful social welfare

  Social welfare in the worst equilibrium

  Social welfare is not defined for ordinal cases

[AD SIGecom Exchange-10]
Analyzing strategic sequential voting using game theory

Prop.30 ∈ {30, 30}  Order: Prop.30 → Prop.38  Prop.38 ∈ {38, 38}

Alice: (30 38) > (30 38) > (30 38) > (30 38)
Bob: (30 38) > (30 38) > (30 38) > (30 38)
Carol: (30 38) > (30 38) > (30 38) > (30 38)

Voting on Prop.30

Voting on Prop.38

Alice: 30 > 30
Bob: 30 > 30
Carol: 30 > 30

Voting on Prop.38

Alice: 38 > 38
Bob: 38 > 38
Carol: 38 > 38

Backward induction

Majority rule is strategy-proof
Game of strategic sequential voting (SSP) [XCL EC-11]

• $k$ binary issues

• Agents vote simultaneously on issues, one issue after another

• For each issue, the majority rule is used to determine the value

• Complete information

• Observation. SSP (backward induction) winner is unique
Strategic behavior is extremely harmful in the worst case

- **Theorem [xcl ec-11]**. For any \( k \geq 2 \) and any \( n \geq 3 \), there exists a situation such that
  - for every order over issues,
  - the SSP winner is ranked below the \((2^k-2k)\)th position in every agent’s true preferences

- **Average case**: open
## Wrap up

<table>
<thead>
<tr>
<th>Combinatorial voting</th>
<th>Economics</th>
<th>CS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Representation</td>
<td>one value per issue</td>
<td>CP-nets</td>
</tr>
<tr>
<td>Aggregation</td>
<td>issue-by-issue voting</td>
<td>sequential voting</td>
</tr>
<tr>
<td>Evaluation</td>
<td>paradoxes</td>
<td>“numerical” paradoxes</td>
</tr>
<tr>
<td></td>
<td>satisfiability of axioms</td>
<td></td>
</tr>
<tr>
<td>Strategic behavior</td>
<td>equilibrium analysis</td>
<td>evaluation of equilibrium outcome</td>
</tr>
</tbody>
</table>
Next class: the hard-to-manipulate axiom

- So far

- Next class