Overview of the COMSOC Research Area
Tutorial at COMSOC-2008

Ulle Endriss
Institute for Logic, Language and Computation
University of Amsterdam

Computational Social Choice

Social choice theory studies mechanisms for collective decision making, such as voting procedures or fair division protocols.

Computational social choice adds a computational perspective, and also explores the use of concepts from social choice in computing.

This tutorial will introduce some of the fundamental terms and ideas of the field, by going through a series of examples.

Caveat: I will omit most citations. The survey paper mentioned below and the proceedings of COMSOC-2006 and COMSOC-2008 are good starting points for finding those papers.

Y. Chevaleyre, U. Endriss, J. Lang, and N. Maudet. *A Short Introduction to Computational Social Choice*. Proc. SOFSEM-2007.

Condorcet Paradox

In 1785 the Marquis de Condorcet noticed a problem ...

Agent 1: $A \succ B \succ C$

Agent 2: $B \succ C \succ A$

Agent 3: $C \succ A \succ B$

How should we *aggregate the individual preferences* of these three agents into a *social preference* ordering?

A beats B and B beats C in pairwise contests. So probably we should have $A \succ B$ and $B \succ C$ in the social preference ordering. And by transitivity also $A \succ C$. But C beats A in a pairwise contest. This is known as the *Condorcet paradox*.

M. le Marquis de Condorcet. *Essai sur l'application de l'analyse à la probabilté des décisions rendues a la pluralité des voix.* Paris, 1785

Arrow's Impossibility Theorem

It seems reasonable to require a *social welfare function* (SWF), mapping profiles of individual preference orderings to a social preference ordering, to satisfy the following axioms:

- *Pareto Condition* (PAR): if every individual prefers alternative x over alternative y, then so should society
- *Independence of Irrelevant Alternatives* (IIA): social preference of x over y should only depend on individual pref's over x and y
- *Non-Dictatorship* (ND): no single individual should be able to impose a social preference ordering

Theorem 1 (Arrow, 1951) *For more than two alternatives, there exists no SWF that satisfies all of (PAR), (IIA) and (ND).*

K.J. Arrow. *Social Choice and Individual Values.* 2nd edition, Wiley, 1963.

Judgement Aggregation

Preferences are not the only structures that we may wish to aggregate. JA studies the aggregation of *judgements* on logically inter-connected propositions. Example:

	A	B	C
Judge 1:	yes	yes	yes
Judge 2:	no	yes	no
Judge 3:	yes	no	no
Majority:	yes	yes	no

A: witness is reliable

B: *if* witness is reliable *then* guilty

C: guilty

note that $A \wedge B \rightarrow C$

While each individual set of judgements is logically consistent, the collective judgement produced by the majority rule is not.

Related: *belief merging* ($\approx$ multiagent belief revision)

Example from Voting

Suppose the *plurality rule* (as in most real-world situations) is used to decide the outcome of an election: the candidate receiving the highest number of votes wins.

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

$$\begin{array}{ll} 49\%: & \text{Bush} \succ \text{Gore} \succ \text{Nader} \\ 20\%: & \text{Gore} \succ \text{Nader} \succ \text{Bush} \\ 20\%: & \text{Gore} \succ \text{Bush} \succ \text{Nader} \\ 11\%: & \text{Nader} \succ \text{Gore} \succ \text{Bush} \end{array}$$

So even if nobody is cheating, Bush will win this election. <u>Issues:</u>

- In a *pairwise contest*, Gore would have defeated anyone (so-called *Condorcet winner*).
- It would have been in the interest of the Nader supporters to *manipulate*, i.e. to misrepresent their preferences.

Voting Procedures

- *Plurality:* elect the candidate ranked first most often
- *Borda:* each voter gives $n-1$ points to the candidate they rank first, $n-2$ to the candidate they rank second, etc., and the candidate with the most points wins
- *Copeland:* award 1 point to a candidate for each pairwise majority contest won and $\frac{1}{2}$ points for each draw, and elect the candidate with the most points
- *Single Transferable Vote (STV)*: keep eliminating the plurality loser until someone has an absolute majority
- *Approval:* voters can approve of as many candidates as they wish, and the candidate with the most approvals wins

The Gibbard-Satterthwaite Theorem

A *voting rule* is a function mapping each profile of individual preference orderings over candidates to a winning candidate.

Possible properties of voting rules:

- A voting rule is *dictatorial* if the winner is always the top candidate of a particular voter (the dictator).
- A voting rule is *manipulable* if it may give a voter an incentive to misrepresent their preferences.

Theorem 2 (Gibbard-Satterthwaite) *For more than two candidates, every voting rule is either dictatorial or manipulable.*

Possible ways around this problem: only consider certain preference structures (e.g. "single-peaked preferences"), ...

A. Gibbard. Manipulation of Voting Schemes. *Econometrica*, 1973.

M.A. Satterthwaite. Strategy-proofness and Arrow's Conditions. *JET*, 1975.

Complexity of Manipulation

By the Gibbard-Satterthwaite Theorem, *any* voting rule for choosing from ≥ 3 candidates is manipulable (or dictatorial).

Idea: So it's always *possible* to manipulate, but maybe it's *difficult*! Tools from *complexity theory* can be used to make this idea precise.

- Yes: manipulation of *STV* is *NP-complete*.
- No: for the *Borda rule*, a greedy algorithm solves the manipulation problem efficiently.
- Recent work: what about average complexity?

J.J. Bartholdi III, C.A. Tovey, and M.A. Trick. The Computational Difficulty of Manipulating an Election. *Soc. Choice and Welfare*, 6(3):227–241, 1989.

J.J. Bartholdi III and J.B. Orlin. Single Transferable Vote Resists Strategic Voting. *Social Choice and Welfare*, 8(4):341–354, 1991.

More on Complexity of Voting

Other questions that have been investigated include:

- What is the complexity of other forms of election manipulation, such as *bribery* and *control*?
- What is the complexity of the *winner determination* problem? For *Dodgson's rule* (electing the candidate requiring the fewest "flips" in ballots to become a Condorcet winner) it is NP-hard.
- Given a subset of ballots, certain candidates may be *possible* or even *necessary winners*. How hard is it to check this?
- What is the *communication complexity* of different voting rules, i.e. how much information needs to be exchanged to determine the winner of an election?

Electing a Committee

Suppose we have to elect a *committee* (not just a single candidate):

- If there are k *seats* to be filled from a pool of m *candidates*, then there are $\binom{m}{k}$ possible outcomes.
- For $k = 5$ and $m = 12$, for instance, that's 792 alternatives.
- The domain of alternatives has a *combinatorial structure*.

It does not seem reasonable to ask voters to submit their full preferences over all alternatives to the collective decision making mechanism. What would be a reasonable form of balloting?

Example for a social choice problem in a combinatorial domain . . .

Social Choice in Combinatorial Domains

Social choice in combinatorial domains first of all requires suitable representation. Languages for compact preference representation include *CP-nets*, *weighted formulas*, or *bidding languages*.

Properties of compact preference representation languages:

- *Expressive power* and correspondence to classes of structures
- Relative *succinctness* of different languages
- *Complexity*, e.g. of finding an optimal alternative

Approaches to *voting* in combinatorial domains:

- Apply voting rules to ballots expressed using a compact representation language
- Analyse to what extent we can vote issue-by-issue, possibly sequentially

Multiagent Resource Allocation

Another example for social choice in combinatorial domains is the problem of allocating indivisible goods to agents.

Issues of interest include:

- Algorithms for finding an optimal allocation
- Procedures that work without central control
- Questions of computational and communication complexity
- Will agents report their true preferences to the mechanism?
- What does "(socially) optimal" actually mean?

Mechanism Design

We have seen that *manipulation* is a serious problem in voting. In domains other than voting we can sometimes do better.

Suppose we want to sell a single item in an auction.

- *First-price sealed-bid auction:* each bidder submits an offer in a sealed envelope; highest bidder wins and pays what they offered
- *Vickrey auction:* each bidder submits an offer in a sealed envelope; highest bidder wins but pays *second highest price*

In the Vickrey auction each bidder has an incentive to submit their *truthful valuation* of the item!

W. Vickrey. Counterspeculation, Auctions, and Competitive Sealed Tenders. *Journal of Finance* 16(1):8–37, 1961.

Efficiency and Fairness

- *Pareto efficiency:* there is no other agreement that is better for some individuals without being worse for any of the others
- Maximise a suitable *collective utility function* (CUF):
  - *utilitarian CUF* = sum of individual utilities
  - *egalitarian CUF* = minimum individual utility
  - *Nash CUF* = product of individual utilities
- *Proportionality:* each agent gets at least $1/n$ of the full lot, according to their own valuation
- *Envy-freeness:* no agent prefers to take the bundle allocated to one of their peers rather than keeping their own

Cake Cutting

The *Banach-Knaster last-diminisher procedure* for finding a *proportional* division of a cake amongst *n players*:

(1) Player 1 cuts off a piece (that she considers to represent $1/n$).

(2) That piece is passed around the players. Each player either lets it pass (if she considers it too small) or trims it down further (to what she considers $1/n$).

(3) After the piece has made the full round, the last player to cut something off (the "last diminisher") is obliged to take it.

(4) The rest (including the trimmings) is then divided amongst the remaining $n-1$ players. Play "cut-and-choose" once $n = 2$. ✓

Requires $O(n^2)$ *cuts*. Best possible procedure needs $O(n \log n)$ *cuts*.

Guaranteeing *envy-freeness* is *much* more difficult: there is no known procedure for 4 players giving contiguous pieces.

Logical Modelling of Social Choice Mechanisms

Logic has long been used to formally specify computer systems, enabling formal or even automatic verification (e.g. via model checking). Maybe we can apply a similar methodology to social choice mechanisms?

Rohit Parikh has coined the term *social software* for this research agenda.

He proposes logics based on PDL as a good starting point and succeeds in (partly) modelling the Banach-Knaster last-diminisher procedure.

Example for a formula from his formalisation of the procedure:

$$F(m,k) \rightarrow \langle c \rangle (F(m,k{-}1) \wedge F(x,1))$$

This says: if the $m$ain piece is large enough (Fair) for k agents, then there *exists* a $c$ut such that the remaining $m$ain piece is fair for $k{-}1$ players and the piece x that has been cut off is fair for 1 player.

R. Parikh. Social Software. *Synthese*, 132(3):187–211, 2002.

R. Parikh. The Logic of Games and its Applications. *Annals of Discrete Mathematics*, 24:111–140, 1985.

Conclusion

Computational social choice studies collective decision making, with an emphasis on computational aspects. Work in COMSOC can be broadly classified along two dimensions —

The kind of *social choice problem* studied, e.g.:

- aggregating individual judgements into a collective verdict
- electing a winner given individual preferences over candidates
- fairly dividing a cake given individual tastes

The kind *computational technique* employed, e.g.:

- algorithm design to implement complex mechanisms
- complexity theory to understand limitations
- KRR techniques to compactly model problems
- logical modelling to fully formalise intuitions

Last Slide

- I have omitted many important and interesting topics, such as ranking systems, coalition formation, or matching … some (not all) of this is covered in the COMSOC Survey cited below.
- I teach a full-semester *course* on COMSOC in Amsterdam; for more slides and references, as well as exercises, see:
  - ▶ `http://www.illc.uva.nl/~ulle/teaching/comsoc/`
- Subscribe to the COMSOC *mailing list* for announcements:
  - ▶ `http://lists.duke.edu/sympa/info/comsoc`
- Enjoy the workshop!

Y. Chevaleyre, U. Endriss, J. Lang, and N. Maudet. *A Short Introduction to Computational Social Choice*. Proc. SOFSEM-2007.