Tutorial on Computational Social Choice

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

[http://www.illc.uva.nl/~ulle/teaching/ecai-2010/]

Table of Contents

Introduction

Social Choice Theory

SCT studies collective decision making: how should we aggregate the preferences of the members of a group to obtain a "social preference"?

$$\triangle \succ_1 \bigcirc \succ_1 \square$$

$$\square \succ_2 \triangle \succ_2 \bigcirc$$

$$\bigcirc \succ_3 \square \succ_3 \triangle$$

$$\overline{}$$

$$?$$

SCT is traditionally studied in Economics and Political Science, but now also by "us": *Computational Social Choice*.

The Research Area

Social choice theory studies mechanisms for collective decision making: voting, preference aggregation, fair division, judgment aggregation, matching, coalition formation ...

- Precursors: Condorcet, Borda (18th century) and others
- serious scientific discipline since 1950s
- Classics: Black, Arrow, May, Sen, Gibbard, Satterthwaite, ...

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

- "classical" papers: $\sim$1990 (Bartholdi, Tovey, Trick, Orlin)
- active research area with regular contributions since $\sim$2002
- name "COMSOC" and biannual workshop since 2006

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

The COMSOC Research Community

- International Workshop on Computational Social Choice:
  - 1st edition: COMSOC-2006 in Amsterdam, December 2006
    48 paper submissions and 80 participants (14 countries)
  - 2nd edition; COMSOC-2008 in Liverpool, September 2008
    55 paper submissions and ~80 participants (~20 countries)
  - 3rd edition: COMSOC-2010 in Düsseldorf, September 2010
    58 paper submissions
- Special issues in international journals:
  - *Mathematical Logic Quarterly*, vol. 55, no. 4, 2009
  - *Journal of Autonomous Agents and Multiagent Systems* (2011)
  - *Mathematical Social Sciences* (in preparation)
- Journals and conferences in AI, MAS, TCS, Logic, Econ, ...
- COMSOC website: `http://www.illc.uva.nl/COMSOC/`

Social Choice and AI (1)

Social choice theory has natural applications in AI:

- *Search Engines:* to determine the most important sites based on links ("votes") + to aggregate the output of several search engines
- *Recommender Systems:* to recommend a product to a user based on earlier ratings by other users
- *Multiagent Systems:* to aggregate the beliefs + to coordinate the actions of groups of autonomous software agents
- *AI Competitions:* to determine who has developed *the best* trading agent / SAT solver / RoboCup team

But not all of the classical assumptions will fit these new applications. So AI needs to develop *new models* and *ask new questions*.

Social Choice and AI (2)

Vice versa, techniques from AI, and computational techniques in general, are useful for advancing the state of the art in social choice:

- *Algorithms and Complexity:* to develop algorithms for (complex) voting procedures + to understand the hardness of "using" them
- *Knowledge Representation:* to compactly represent the preferences of individual agents over large spaces of alternatives
- *Logic and Automated Reasoning:* to formally model problems in social choice + to automatically verify (or discover) theorems

Indeed, you will find many papers on social choice at AI conferences (e.g., IJCAI, ECAI, AAAI, AAMAS) and many AI researchers participate in events dedicated to social choice (e.g., COMSOC).

This Tutorial

The aim of this tutorial is to give a tentative overview of the COMSOC research area, highlighting contributions of and opportunities for AI.

We will concentrate on these topics:

- Preference Aggregation
- Manipulation of Voting Procedures
- Voting in Combinatorial Domains
- Judgment Aggregation
- Fair Division

Preference Aggregation

Preference Aggregation: Formal Framework

Problem: How can we amalgamate the preferences of a group of individuals into a collective preference order?

Basic terminology and notation:

- finite set of *individuals* $\mathcal{N} = \{1, \ldots, n\}$
- (usually finite) set of *alternatives* $\mathcal{X} = \{x_1, x_2, x_3, \ldots\}$
- Denote the set of *linear orders* on $\mathcal{X}$ by $\mathcal{L}(\mathcal{X})$.
  Preferences are assumed to be elements of $\mathcal{L}(\mathcal{X})$.

A *social welfare function* is a function $F : \mathcal{L}(\mathcal{X})^n \to \mathcal{L}(\mathcal{X})$, mapping profiles of individual preference orders to a societal preference order.

Axioms: Some Natural Desiderata

It seems reasonable to postulate that any aggregator F should satisfy the following list of axioms:

- **(WP)** F should satisfy the *weak Pareto condition:* if every individual prefers x over y, then so should society.
- **(IIA)** F should satisfy *independence of irrelevant alternatives:* the relative social ranking of x and y should depend only on the relative individual rankings of x and y.
- **(ND)** F should be *nondictatorial:* no single individual should be able to impose the social preference ordering.

Arrow's Theorem

This is widely regarded as *the* seminal result in Social Choice Theory. Kenneth J. Arrow received the Nobel Prize in Economics in 1972.

Theorem 1 (Arrow, 1951) *There exists no social welfare function for $\geqslant 3$ alternatives that satisfies all of (WP), (IIA) and (ND).*

Remarks:

- Note that this is a *surprising* result!
- Note that the theorem does *not* hold for *two* alternatives.

K.J. Arrow. *Social Choice and Individual Values*. 2nd edition. Cowles Foundation, Yale University Press, 1963.

J. Geanakoplos. Three Brief Proofs of Arrow's Impossibility Theorem. *Economic Theory*, 26(1):211–215, 2005.

Logic and Automated Reasoning

Logic has long been used to *formally specify* computer systems, facilitating formal or even *automatic verification* of various properties.

Can we apply this methodology also to *social choice* mechanisms?

- What logic fits best?
- Which automated reasoning methods are useful?

Computer-aided Proof of Arrow's Theorem

Tang and Lin (2009) prove two inductive lemmas:

- If there exists an Arrovian aggregator for n individuals and $m+1$ alternatives, then there exists one for n and m (if $n \geqslant 2$, $m \geqslant 3$).
- If there exists an Arrovian aggregator for $n+1$ individuals and m alternatives, then there exists one for n and m (if $n \geqslant 2$, $m \geqslant 3$).

Tang and Lin then show that the *"base case"* of Arrow's Theorem with 2 individuals and 3 alternatives can be modelled in *propositional logic*.

A SAT solver can *verify* $\textsc{Arrow}(2,3)$ to be correct in < 1 second — that's $(3!)^{3! \times 3!} \approx 10^{28}$ aggregators to check.

Discussion: Opens up opportunities for quick sanity checks of hypotheses regarding new impossibility theorems.

P. Tang and F. Lin. Computer-aided Proofs of Arrow's and other Impossibility Theorems. *Artificial Intelligence*, 173(11):1041–1053, 2009.

Related Work

- Ågotnes et al. (2010) propose a modal logic to model preferences and their aggregation that can express Arrow's Theorem.
- Arrow's Theorem holds *iff* the set $T_{\textsc{Arrow}}$ of FOL formulas (defined in the paper) has no finite models (Grandi and E., 2009).
- Nipkow (2009) formalises and verifies a known *proof* of Arrow's Theorem in the HOL proof assistant ISABELLE.

T. Ågotnes, W. van der Hoek, and M. Wooldridge. On the Logic of Preference and Judgment Aggregation. *J. Auton. Agents and Multiagent Sys*. In press (2010).

U. Grandi and U. Endriss. *First-order Logic Formalisation of Arrow's Theorem*. Proc. 2nd Internat. Workshop on Logic, Rationality and Interaction (LORI-2009).

T. Nipkow. Social Choice Theory in HOL. *Journal of Automated Reasoning*, 43(3):289–304, 2009.

Literature

Arrow's Theorem is the seminal example for the use of the *axiomatic method* in social choice theory. To learn more, refer to the textbooks by Gaertner (2009) and Taylor (2005).

A comprehensive approach for formalisation (with *symbolic* languages) of concepts from social choice theory and applications of automated reasoning technology is yet to be developed. For a host of ideas along (roughly) those lines, read Parikh (2002) on *"social software"*.

W. Gaertner. *A Primer in Social Choice Theory*. Revised edition. LSE Perspectives in Economic Analysis. Oxford University Press, 2009.

A.D. Taylor. *Social Choice and the Mathematics of Manipulation*. Cambridge University Press, 2005.

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

Manipulation of Voting Procedures

Example

Suppose the *plurality rule* is used to decide an election: the candidate receiving the highest number of votes wins.

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

49%:	Bush $\succ$ Gore $\succ$ Nader
20%:	Gore $\succ$ Nader $\succ$ Bush
20%:	Gore $\succ$ Bush $\succ$ Nader
11%:	Nader $\succ$ Gore $\succ$ Bush

So even if nobody is cheating, Bush will win this election. But:

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

Is there a better voting procedure that avoids these problems?

Voting Procedures

There are *many* different voting procedures, including these:

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

S.J. Brams and P.C. Fishburn. *Voting Procedures*. In K.J. Arrow *et al.* (eds.), *Handbook of Social Choice and Welfare*, Elsevier, 2002.

Voting: Formal Framework

Basic terminology and notation:

- finite set of *voters* $\mathcal{N} = \{1, \ldots, n\}$, the *electorate*
- (usually finite) set of *alternatives*/*candidates* $\mathcal{X} = \{x_1, x_2, x_3, \ldots\}$
- Denote the set of *linear orders* on $\mathcal{X}$ by $\mathcal{L}(\mathcal{X})$. *Preferences* are assumed to be elements of $\mathcal{L}(\mathcal{X})$. *Ballots* are elements of $\mathcal{L}(\mathcal{X})$.

A *voting procedure* is a function $F : \mathcal{L}(\mathcal{X})^n \to 2^{\mathcal{X}} \setminus \{\emptyset\}$, mapping profiles of ballots to nonempty sets of alternatives (*winners*).

F is called *resolute* if $F(\underline{b})$ is a singleton for any profile $\underline{b} \in \mathcal{L}(\mathcal{X})^n$.

Remark: AV does not fit in this framework; everything else does.

Strategy-Proofness

Let $\succ_i$ be the true preference of voter i and let b_i be the ballot of i.

A resolute voting procedure F is *strategy-proof* if there exist no profile $\underline{b} = (b_1, \ldots, b_n)$ and no voter i s.t. $F(\underline{b}) \succ_i F(b_1, \ldots, \succ_i, \ldots, b_n)$, with $\succ_i$ lifted from alternatives to singletons in the natural manner.

That is:

- A voting procedure is strategy-proof if it never gives a voter an incentive to misrepresent her true preferences.

Remarks:

- We have seen that the *plurality rule* is *not* strategy-proof.
- Any *dictatorial* rule (where the unique winner is always the top choice of the dictator) clearly is strategy-proof.

The Gibbard-Satterthwaite Theorem

A resolute voting procedure F is *surjective* if for any alternative x there exists a ballot profile $\underline{b}$ such that $F(\underline{b}) = \{x\}$.

Theorem 2 (Gibbard-Satterthwaite) *Any resolute voting procedure for $\geqslant 3$ alternatives that is surjective and strategy-proof is dictatorial.*

Remarks:

- Again, *surprising*. Again, not applicable for *two* alternatives.
- *Random* procedures don't count (but might be "strategy-proof").

A. Gibbard. Manipulation of Voting Schemes: A General Result. *Econometrica*, 41(4):587–601, 1973.

M.A. Satterthwaite. Strategy-proofness and Arrow's Conditions. *Journal of Economic Theory*, 10:187–217, 1975.

J.-P. Benoît. The Gibbard-Satterthwaite Theorem: A Simple Proof. *Economic Letters*, 69:319–322, 2000.

What to do?

The Gibbard-Satterthwaite tells us (essentially) that there are no reasonable voting procedures that are strategy-proof. *That's very bad!*

We will consider three possible avenues to circumvent this problem:

- Restricting the domain (the classical approach)
- Changing the formal framework a little
- Making manipulation computationally hard

Domain Restriction: Single-Peaked Preferences

An electorate $\mathcal{N}$ has *single-peaked* preferences if there exists a "left-to-right" ordering $\gg$ on the alternatives such that any voter prefers x to y if x is between y and her top alternative wrt. $\gg$.

The same definition can be applied to profiles of ballots.

Remarks:

- Quite natural: classical spectrum of political parties; decisions involving agreeing on a number (e.g., legal drinking age); ...
- But certainly not universally applicable.

Black's Median Voter Theorem

For simplicity, assume the number of voters is *odd*.

For a given left-to-right ordering $\gg$, the *median voter rule* asks each voter for their top alternative and elects the alternative proposed by the voter corresponding to the median wrt. $\gg$.

Theorem 3 (Black's Theorem, 1948) *If an odd number of voters submit single-peaked ballots, then there exists a Condorcet winner (= an alternative beating any other alternative in pairwise contests) and it will get elected by the median voter rule.*

And: if an odd number of voters have single-peaked preferences, then the median voter rule is *strategy-proof* (Gibbard-Satterthwaite fails).

D. Black. On the Rationale of Group Decision-Making. *The Journal of Political Economy*, 56(1):23–34, 1948.

Beyond Classical Voting Theory

Another potential way out is to relax the confinements of classical voting theory, and to consider other types of preferences and ballots. In one such approach (E. et al., 2009),

- *preferences* and *ballots* can be different structures; and
- a notion of *sincerity* replaces the standard notion of *truthfulness* (because the ballot language may *not allow* you to be truthful).

Now you can get positive results for certain combinations:

- Under *approval voting* with standard (linear) preferences, you can never benefit from not voting sincerely.
- If you have *dichotomous preferences*, you can never benefit from not voting sincerely for a wide range of voting procedures.

U. Endriss, M.S. Pini, F. Rossi, and K.B. Venable. *Preference Aggregation over Restricted Ballot Languages: Sincerity and Strategy-Proofness*. Proc. IJCAI-2009.

Mechanism Design with Numerical Preferences

If we move from ordinal preferences to *valuation functions*, then designing strategy-proof mechanisms *is* possible:

- Suppose we want to sell a single item in an auction.
- *Vickrey auction:* each bidder submits an offer in a sealed envelope; highest bidder wins but pays *second highest price*
- Each bidder has an incentive to declare their *truthful valuation*.

(But note that this setting is quite different from voting.)

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

Complexity as a Barrier against Manipulation

The Gibbard-Satterthwaite Theorem shows that manipulation is always possible. But how hard is it to find a manipulating ballot?

The seminal paper by Bartholdi, Tovey and Trick (1989) starts by showing that manipulation is in fact *easy* for a range of commonly used voting rules, and then presents one system (a variant of the Copeland rule) for which manipulation is NP-complete. Next:

- We first present a couple of these easiness results, namely for *plurality* and for the *Borda rule*.
- We then present a result from a follow-up paper by Bartholdi and Orlin (1991): the manipulation of *STV* is *NP-complete*.

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.

Manipulability as a Decision Problem

We can cast the problem of manipulability, for a particular voting procedure F, as a decision problem:

> $\textsc{Manipulability}(F)$
> **Instance:** Set of ballots for all but one voter; alternative x.
> **Question:** Is there a ballot for the final voter such that x wins?

In practice, a manipulator would have to solve $\textsc{Manipulability}(F)$ for all alternatives, in order of her preference.

If the $\textsc{Manipulability}(F)$ is computationally intractable, then manipulability may be considered less of a worry for procedure F.

Manipulating the Plurality Rule

Recall the plurality rule:

- Each voter submits a ballot showing the name of one of the alternatives. The alternative receiving the most votes wins.

The plurality rule is easy to manipulate (trivial):

- Simply vote for x, the alternative to be made winner by means of manipulation. If manipulation is possible at all, this will work. Otherwise not.

That is, we have $\textsc{Manipulability}(\mathit{plurality}) \in \mathrm{P}$.

General: $\textsc{Manipulability}(F) \in \mathrm{P}$ for any rule F with polynomial winner determination problem and polynomial number of ballots.

Manipulating the Borda Rule

Recall Borda: submit a ranking (super-polynomially many choices!) and give $m-1$ points to 1st ranked, $m-2$ points to 2nd ranked, etc.

The Borda rule is also easy to manipulate. Use a *greedy algorithm:*

- Place x (the alternative to be made winner through manipulation) at the top of your declared preference ordering.
- Then inductively proceed as follows: Check if any of the remaining alternatives can be put next into the preference ordering without preventing x from winning. If yes, do so.
  If no, terminate and say that manipulation is impossible.

After convincing ourselves that this algorithm is indeed correct, we also get $\textsc{Manipulability}(\mathit{Borda}) \in \mathrm{P}$.

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.

Intractability of Manipulating STV

Recall STV: eliminate plurality losers until an alternative gets $> 50\%$

Theorem 4 (Bartholdi and Orlin, 1991) *Manipulation of STV is NP-complete.*

Discussion: NP is a worst-case notion. What about average complexity?

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

P. Faliszewski and A.D. Procaccia. AI's War on Manipulation: Are We Winning? *AI Magazine*. In press (2010).

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*? See Faliszewski et al. (2009) for a survey.
- After some of the ballots have been counted, certain candidates may be *possible winners* or even *necessary winners*. How hard is it to check this? See e.g. Konczak and Lang (2005).

P. Faliszewski, E. Hemaspaandra, L.A. Hemaspaandra, and J. Rothe. *A Richer Understanding of the Complexity of Election Systems*. In *Fundamental Problems in Computing*, Springer-Verlag, 2009.

K. Konczak and J. Lang. *Voting Procedures with Incomplete Preferences*. Proc. Advances in Preference Handling 2005.

Even More on Complexity of Voting

- 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? See Conitzer and Sandholm (2005).
- After having counted part of the vote, can we *compile* this information into a more compact form than just storing all the ballots? And how complex is it to reason about this information? See Chevaleyre et al. (2009).

V. Conitzer and T. Sandholm. *Communication Complexity of Common Voting Rules*. Proc. ACM Conference on Electronic Commerce 2005.

Y. Chevaleyre, J. Lang, N. Maudet, and G. Ravilly-Abadie. *Compiling the Votes of a Subelectorate*. Proc. IJCAI-2009.

Literature

The manipulation problem is discussed at great length in the textbook by Taylor (2005), and Gaertner (2001) provides an in-depth discussion of domain restrictions.

Faliszewski and Procaccia (2010) review the literature on exploiting computational complexity as a barrier against manipulation in voting.

A.D. Taylor. *Social Choice and the Mathematics of Manipulation*. Cambridge University Press, 2005.

W. Gaertner. *Domain Conditions in Social Choice Theory*. Cambridge University Pres, 2001.

P. Faliszewski and A.D. Procaccia. AI's War on Manipulation: Are We Winning? *AI Magazine*. In press (2010).

Voting in Combinatorial Domains

Voting in Combinatorial Domains

Besides the complexity-theoretic properties of voting procedures, another computational concern in voting is raised by the fact that the alternatives to vote for often have a *combinatorial structure:*

- Electing a committee of k members from amongst n candidates.
- During a referendum (in Switzerland, California, places like that), voters may be asked to vote on several propositions.

We will see an example and look into several possible approaches . . .

Based on J. Lang's "5 solutions". Read it in Chevaleyre et al. (2008).

Y. Chevaleyre, U. Endriss, J. Lang, and N. Maudet. Preference Handling in Combinatorial Domains: From AI to Social Choice. *AI Magazine*, 29(4):37–46, 2008.

Example

Suppose 13 voters are asked to each vote *yes* or *no* on three issues; and we use the plurality rule for each issue independently to select a winning combination:

- 3 voters each vote for YNN, NYN, NNY.
- 1 voter each votes for YYY, YYN, YNY, NYY.
- No voter votes for NNN.

But then NNN wins: 7 out of 13 vote *no* on each issue.

This is an instance of the *paradox of multiple elections:* the winning combination receives the fewest number of votes.

S.J. Brams, D.M. Kilgour, and W.S. Zwicker. The Paradox of Multiple Elections. *Social Choice and Welfare*, 15(2):211–236, 1998.

Basic Solution Attempts

- Solution 1: just vote for combinations directly
  - only feasible for *very* small problem instances
  - Example: 3-seat committee, 10 candidates $\rightsquigarrow \binom{10}{3} = 120$
- Solution 2: vote for top k combinations only (e.g., $k = 1$)
  - does address communication problem of Solution 1
  - possibly nobody gets more than one vote (tie-breaking decides)
- Solution 3: make a small preselection of combinations to vote on
  - does solve the computational problems
  - but who should select? (strategic control)

Combinatorial Vote

Idea: Ask voters to report their ballots by means of expressions in a *compact preference representation language* and apply your favourite voting procedure to the succinctly encoded ballots received.

Lang (2004) calls this approach *combinatorial vote*.

Discussion: This seems the most promising approach so far, although not too much is known to date what would be good choices for preference representation languages or voting procedures, or what algorithms to use to compute the winners. Also, complexity can be expected to be very high.

J. Lang. Logical Preference Representation and Combinatorial Vote. *Annals of Mathematics and Artificial Intelligence*, 42(1–3):37–71, 2004.

Example

Use the language defined by the *leximin ordering over prioritised goals* with the *Borda rule* (goals are labelled by their rank):

- Voter 1: $\{A{:}0, B{:}1\}$ induces order $AB \succ_1 A\bar{B} \succ_1 \bar{A}B \succ_1 \bar{A}\bar{B}$
- Voter 2: $\{A \vee \neg B{:}0\}$ induces order $A\bar{B} \sim_2 AB \sim_2 \bar{A}\bar{B} \succ_2 \bar{A}B$
- Voter 3: $\{\neg A{:}0, B{:}0\}$ induces order $\bar{A}B \succ_3 \bar{A}\bar{B} \sim_3 AB \succ_3 A\bar{B}$

As the induced orders need not be strict linear orders, we use a *generalisation of the Borda rule:* a candidate gets as many points as she dominates other candidates. So we get these Borda counts:

$$\begin{array}{ll} AB : 3+1+1=5 & \bar{A}B : 1+0+3=4 \\ A\bar{B} : 2+1+0=3 & \bar{A}\bar{B} : 0+1+1=2 \end{array}$$

So combination AB wins.

Combinatorial vote *proper* would be to compute the winner *directly* from the goal bases, without the detour via the induced orders.

Preference Representation Languages

Several languages have been proposed for the compact representation of preference orders. See Lang (2004) for an overview. Examples:

- *Weighted goals:* use propositional logic to express goals; assign weights to express importance; aggregate (e.g., lexicographically)
- *CP-nets:* use a directed graph to express dependence between issues; use conditional preference tables to specify preferences on issue assuming those it depends on are fixed

J. Lang. Logical Preference Representation and Combinatorial Vote. *Annals of Mathematics and Artificial Intelligence*, 42(1–3):37–71, 2004.

C. Boutilier, R.I. Brafman, C. Domshlak, H.H. Hoos, and D. Poole. CP-nets: A Tool for Representing and Reasoning with Conditional *Ceteris Paribus* Preference Statements. *Journal of AI Research*, 21:135–191, 2004.

Other Approaches

Vote on each issue separately *but* —

- identify *conditions* under which this does not lead to undesirable outcomes ("separable preferences")
- find a *novel way of aggregating* the ballots to select a winner
  - Example: elect the combination minimising the maximal Hamming distance to any of the ballots (Brams et al., 2007)
- vote *sequentially* rather than simultaneously
  - Example: Lang and Xia (2009) use CP-nets to represent ballots and use the underlying graph as an agenda

S.J. Brams, D.M. Kilgour, and M.R. Sanver. A Minimax Procedure for Electing Committees. *Public Choice*, 132:401–420, 2007.

J. Lang and L. Xia. Sequential Composition of Voting Rules in Multi-issue Domains. *Mathematical Social Sciences*, 57(3):304–324, 2009.

Literature

Chevaleyre et al. (2008) give an introduction to social choice in combinatorial domains.

An important reference is the paper by Lang (2004), which also includes a lot of information regarding the compact representation of (ordinal) preferences.

Y. Chevaleyre, U. Endriss, J. Lang, and N. Maudet. Preference Handling in Combinatorial Domains: From AI to Social Choice. *AI Magazine*, 29(4):37–46, 2008.

J. Lang. Logical Preference Representation and Combinatorial Vote. *Annals of Mathematics and Artificial Intelligence*, 42(1–3):37–71, 2004.

Judgment Aggregation

The Doctrinal Paradox

Story: three judges have to decide whether the defendant is guilty ...

	p	$p \rightarrow q$	q
Judge 1:	Yes	Yes	Yes
Judge 2:	No	Yes	No
Judge 3:	Yes	No	No
Majority:	Yes	Yes	No

Paradox: each *individual* judgment set is *consistent*, but the *collective* judgment arrived at using the *majority rule* is not

L.A. Kornhauser and L.G. Sager. The One and the Many: Adjudication in Collegial Courts. *California Law Review*, 81(1):1–59, 1993.

Judgment Aggregation: Formal Framework

Notation: ${\sim}\alpha := \beta$ if $\alpha = \neg\beta$ and ${\sim}\alpha := \neg\alpha$ otherwise (*complement*)

An *agenda* Φ is a finite nonempty set of propositional formulas (w/o double negation) closed under complementation: $\alpha \in \Phi \;\Rightarrow\; {\sim}\alpha \in \Phi$.

A *judgment set* J on an agenda Φ is a subset of Φ. We call J:

- *complete* if $\alpha \in J$ or ${\sim}\alpha \in J$ for all $\alpha \in \Phi$
- *complement-free* if $\alpha \notin J$ or ${\sim}\alpha \notin J$ for all $\alpha \in \Phi$
- *consistent* if there exists an assignment satisfying all $\alpha \in J$

Let $\mathcal{J}(\Phi)$ be the set of all complete and consistent subsets of Φ.

Now a finite set of *individuals* $\mathcal{N} = \{1, \ldots, n\}$ with $n \geqslant 3$ express judgments on Φ, giving rise to a *profile* $\mathbf{J} = (J_1, \ldots, J_n)$.

An *aggregation procedure* for agenda Φ and a set of n individuals is a function mapping a profile of complete and consistent individual judgment sets to a single collective judgment set: $F : \mathcal{J}(\Phi)^n \to 2^\Phi$.

Axioms

Use *axioms* to express desiderata for F. Examples:

Anonymity (A): For any profile $\mathbf{J}$ and any permutation $\sigma : \mathcal{N} \to \mathcal{N}$ we have $F(J_1, \ldots, J_n) = F(J_{\sigma(1)}, \ldots, J_{\sigma(n)})$.

Neutrality (N): For any φ, ψ in the agenda Φ and profile $\mathbf{J} \in \mathcal{J}(\Phi)$, if for all i we have $\varphi \in J_i \Leftrightarrow \psi \in J_i$, then $\varphi \in F(\mathbf{J}) \Leftrightarrow \psi \in F(\mathbf{J})$.

Independence (I): For any φ in the agenda Φ and profiles $\mathbf{J}$ and $\mathbf{J}'$ in $\mathcal{J}(\Phi)$, if $\varphi \in J_i \Leftrightarrow \varphi \in J_i'$ for all i, then $\varphi \in F(\mathbf{J}) \Leftrightarrow \varphi \in F(\mathbf{J}')$.

Systematicity (S) $=$ (N) $+$ (I)

C. List and C. Puppe. Judgment Aggregation: A Survey. *Handbook of Rational and Social Choice*. Oxford University Press, 2009.

Impossibility Theorem

We have seen that the majority rule is not consistent.

Is there a reasonable procedure that is?

Theorem 5 (List and Pettit, 2002) *If the agenda contains at least P, Q and $P \wedge Q$, then no aggregation procedure producing consistent and complete judgment sets satisfies both* (A) *and* (S).

C. List and P. Pettit. Aggregating Sets of Judgments: An Impossibility Result. *Economics and Philosophy*, 18(1):89–110, 2002.

Research in Judgment Aggregation

Some of the questions that have been investigated:

- What impossibilities are there? How can we circumvent them?
  - Relax some of the axioms? Which? Why?
  - Impose domain restrictions?
- What properties characterise agendas for which a given combination of axioms permits a consistent procedure?
- What happens if we change the underlying logic?
- What are acceptable procedures in practice? (that guarantee consistent outcomes but may violate other axioms)
- What are the precise connections to preference aggregation?

C. List and C. Puppe. Judgment Aggregation: A Survey. *Handbook of Rational and Social Choice*. Oxford University Press, 2009.

Complexity of Judgment Aggregation

There are a number of natural questions arising in JA of which we might want to know their complexity. This is known to date:

- *Safety of the Agenda:* deciding whether an agenda can guarantee consistency for a given class of aggregators (characterised by standard axioms) is typically Π_2^p-*complete* (worse than NP).
- *Winner Determination:* the complexity of computing the collective judgment set can range from *P* (*premise-based procedure*) to *NP-complete* (*distance-based merging*).
- *Manipulation:* in at least one case there is a *complexity gap* between winner determination and manipulation: manipulation is *NP-complete* for the *premise-based procedure*.

U. Endriss, U. Grandi, and D. Porello. *Complexity of Judgment Aggregation: Safety of the Agenda*. Proc. AAMAS-2010.

U. Endriss, U. Grandi, and D. Porello. *Complexity of Winner Determination and Strategic Manipulation in Judgment Aggregation*. Proc. COMSOC-2010.

JA and Argumentation in AI

As an example for uses of JA in AI, Caminada and Pigozzi (2009) apply JA methodology in work on abstract argumentation:

- A directed graph describes the *attack*-relation between arguments.
- Rules for consistently labelling arguments as *accepted*, *rejected* or *undecided* provide a semantics.
- Individuals may propose different labellings. How should we aggregate this information? What are relevant axioms?

Other applications of JA in AI are yet to be explored (e.g., there seem to be obvious applications in multiagent systems).

M. Caminada and G. Pigozzi. On Judgment Aggregation in Abstract Argumentation. *Journal of Autonomous Agents and Multiagent Systems*. In press (2009).

Literature

List (2010) gives a good introduction to the field of JA. Refer to List and Puppe (2009) for additional technical detail.

There are parallels between work in JA and *belief merging* in AI (cf. Konieczny and Pino Pérez, 2002), which are yet to be fully explored.

C. List. The Theory of Judgment Aggregation: An Introductory Review. *Synthese*. In press (2010).

C. List and C. Puppe. Judgment Aggregation: A Survey. *Handbook of Rational and Social Choice*. Oxford University Press, 2009.

S. Konieczny and R. Pino Pérez. Merging Information under Constraints: A Logical Framework. *Journal of Logic and Computation*, 12(5):773–808, 2002.

Fair Division

Fair Division

Fair division is the problem of dividing one or several goods amongst two or more agents in a way that satisfies a suitable fairness criterion.

This can be considered a problem of *social choice:*

- A group of agents each have individual preferences over a collective agreement (the allocation of goods to be found).
- But: in fair division preferences are often assumed to be cardinal (*utility functions*) rather than ordinal (as in voting)
- And: fair division problems come with some *internal structure* often absent from other social choice problems (e.g., I will be indifferent between allocations giving me the same set of goods)

U. Endriss. *Lecture Notes on Fair Division*. ILLC, University of Amsterdam, 2010.

Fair Division: Formal Framework

- Let $\mathcal{N} = \{1, \ldots, n\}$ be a set of *agents* (or *players*, or *individuals*) who need to share several *goods* (or *resources*, *items*, *objects*).
- An *allocation* A is a mapping of agents to *bundles* of goods.
- Each agent $i \in \mathcal{N}$ has a *utility function* u_i (or *valuation function*), mapping allocations to the reals, to model their preferences.
  - Typically, u_i first defined on bundles, so: $u_i(A) = u_i(A(i))$.
  - Discussion: preference intensity, interpersonal comparison
- An allocation A gives rise to a *utility vector* $\langle u_1(A), \ldots, u_n(A) \rangle$.
- Sometimes, we are going to define social preference structures directly over utility vectors $u = \langle u_1, \ldots, u_n \rangle$ (elements of $\mathbb{R}^n$), rather than speaking about the allocations generating them.

Pareto Efficiency

Allocation A is *Pareto dominated* by allocation A' if $u_i(A) \leqslant u_i(A')$ for all agents $i \in \mathcal{N}$ and this inequality is strict in at least one case.

An allocation A is *Pareto efficient* if there is no other feasible allocation A' such that A is Pareto dominated by A'.

The idea goes back to Vilfredo Pareto (Italian economist, 1848–1923).

Discussion:

- Pareto efficiency is very often considered a minimum requirement for any reasonable allocation. It is a very weak criterion.
- Only the ordinal content of preferences is needed to check Pareto efficiency (no preference intensity, no interpersonal comparison).

Collective Utility Functions

A *collective utility function* (CUF) is a function $\mathrm{SW} : \mathbb{R}^n \to \mathbb{R}$ mapping utility vectors to the reals. Three important examples:

- the *utilitarian* CUF ranking allocations by total utility

$$\mathrm{SW}_{\mathrm{util}}(u) \quad = \quad \sum_{i \in \mathcal{N}} u_i$$

- the *egalitarian* CUF that ties social welfare to the poorest agent

$$\mathrm{SW}_{\mathrm{egal}}(u) \quad = \quad min\{u_i \mid i \in \mathcal{N}\}$$

- the *Nash* CUF that tries to balance fairness and efficiency

$$\mathrm{SW}_{\mathrm{nash}}(u) \quad = \quad \prod_{i \in \mathcal{N}} u_i$$

H. Moulin. *Axioms of Cooperative Decision Making*. Econometric Society Monographs, Cambridge University Press, 1988.

Envy-Freeness

An allocation is called *envy-free* if no agent would rather have one of the bundles allocated to any of the other agents:

$$u_i(A(i)) \quad \geqslant \quad u_i(A(j))$$

Recall that $A(i)$ is the bundle allocated to agent i in allocation A.

Remark: Envy-free allocations do not always *exist* (at least not if we require either complete or Pareto efficient allocations).

Degrees of Envy

As we cannot always ensure envy-free allocations, another approach would be to try to reduce the *degree of envy* as much as possible.

- Envy between two agents:
  $\max\{u_i(A(j)) - u_i(A(i)), 0\}$ or
  1 if $u_i(A(j)) > u_i(A(i))$ and 0 otherwise
- Degree of envy of a single agent:
  $\max$, sum
- Degree of envy of a society:
  $\max$, sum [or indeed any CUF]

Y. Chevaleyre, U. Endriss, S. Estivie, and N. Maudet. *Reaching Envy-free States in Distributed Negotiation Settings*. Proc. IJCAI-2007.

Indivisible Goods: Formal Framework

Fixing goods to be *indivisible* results in a more concrete framework:

- Set of *agents* $\mathcal{N} = \{1, \ldots, n\}$ and finite set of indivisible *goods* $\mathcal{G}$.
- An *allocation* A is a partitioning of $\mathcal{G}$ amongst the agents in $\mathcal{N}$.
  Example: $A(i) = \{a, b\}$ — agent i owns items a and b
- Each agent $i \in \mathcal{N}$ has got a *valuation function* $v_i : 2^{\mathcal{G}} \rightarrow \mathbb{R}$.
  Example: $v_i(A) = v_i(A(i)) = 577.8$ — agent i is pretty happy
- If agent i receives bundle B and the sum of her payments is x, then her *utility* is $u_i(B, x) = v_i(B) - x$ ("quasi-linear utility").

For fair division of indivisible goods *without money*, assume that payment balances are always equal to 0 (and utility = valuation).

Preference Representation Languages

Example: Allocating 10 goods to 5 agents means $5^{10} = 9765625$ allocations and $2^{10} = 1024$ bundles for each agent to think about.

So we need to choose a good *language* to compactly represent preferences over such large numbers of alternative bundles, e.g.:

- Logic-based languages (weighted goals)
- Bidding languages for combinatorial auctions (OR/XOR)
- Program-based preference representation (straight-line programs)
- CP-nets and CI-nets (for ordinal preferences)

The choice of language affects both *algorithm design* and *complexity*.

See Chevaleyre et al. (2008) for references.

Y. Chevaleyre, U. Endriss, J. Lang, and N. Maudet. Preference Handling in Combinatorial Domains: From AI to Social Choice. *AI Magazine*, 29(4):37–46, 2008.

Complexity Results

Before we look into the "how", here are some complexity results:

- Checking whether an allocation is *Pareto efficient* is coNP-complete.
- Finding an allocation with maximal *utilitarian* social welfare is NP-hard. If all valuations are *modular* (additive) then it is polynomial.
- Finding an allocation with maximal *egalitarian* social welfare is also NP-hard, even when all valuations are modular.
- Checking whether an *envy-free* allocation exists is NP-complete; checking whether an allocation that is both Pareto efficient and envy-free exists is even Σ_2^p-complete.

References to these results may be found in the "MARA Survey".

Y. Chevaleyre, P.E. Dunne, U. Endriss, J. Lang, M. Lemaître, N. Maudet, J. Padget, S. Phelps, J.A. Rodríguez-Aguilar and P. Sousa. Issues in Multiagent Resource Allocation. *Informatica*, 30:3–31, 2006.

Negotiating Socially Optimal Allocations

In AI, one approach has been to design simple *negotiation protocols* allowing agents to make deals regarding the exchange of goods and to analyse the dynamics of those systems.

We can distinguish two perspectives:

- The *local/individual* view: what deals will agents make, given their preferences? Example: myopic rationality
- The *global/social* view: how will the overall allocation evolve in terms of social welfare? Example: utilitarian collective utility

One of the main questions is whether negotiation will *converge* to a social optimum, or under what circumstances it will.

U. Endriss, N. Maudet, F. Sadri and F. Toni. Negotiating Socially Optimal Allocations of Resources. *Journal of AI Research*, 25:315–348, 2006.

Cake-Cutting: Formal Framework

"Cake-cutting" is the problem of fair division of a single *divisible* (and heterogeneous) good between n *agents* (or *players*).

The *cake* is represented by the unit interval $[0, 1]$:

```
|-----------------------|
0                       1
```

Each agent i has a *utility function* u_i (or *valuation*, *measure*) mapping finite unions of subintervals of $[0, 1]$ to the reals, satisfying:

- Non-negativity: $u_i(B) \geqslant 0$ for all $B \subseteq [0, 1]$
- Normalisation: $u_i(\emptyset) = 0$ and $u_i([0, 1]) = 1$
- Additivity: $u_i(B \cup B') = u_i(B) + u_i(B')$ for disjoint $B, B' \subseteq [0, 1]$
- u_i is continuous: the Intermediate-Value Theorem applies and single points do not have any value.

Cut-and-Choose

The classical approach for dividing a cake between *two agents:*

- *One agent cuts the cake in two pieces (which she considers to be of equal value), and the other one chooses one of the pieces (the piece she prefers).*

The cut-and-choose procedure is *proportional:*

- Each agent is guaranteed at least one half (general: $1/n$) according to her own valuation.

Discussion: In fact, the first agent (if she is risk-averse) will receive exactly $1/2$, while the second will usually get more.

What if there are more than two agents?

The Banach-Knaster Last-Diminisher Procedure

In the first ever paper on fair division, Steinhaus (1948) reports on a solution for *arbitrary n* proposed by Banach and Knaster.

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

(2) That piece is passed around the agents. Each agent 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 agent to cut something off (the "last diminisher") is obliged to take it.

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

Each agent is guaranteed a *proportional* piece. Takes $O(n^2)$ steps.

H. Steinhaus. The Problem of Fair Division. *Econometrica*, 16:101–104, 1948.

The Even-Paz Divide-and-Conquer Procedure

Even and Paz (1984) investigated *upper bounds* for the number of *queries* (cuts or marks) required for n *agents*.

They introduced the following *divide-and-conquer* protocol:

(1) Ask each agent to cut the cake at her $\lfloor \frac{n}{2} \rfloor$ / $\lceil \frac{n}{2} \rceil$ mark.

(2) Associate the union of the leftmost $\lfloor \frac{n}{2} \rfloor$ pieces with the agents who made the leftmost $\lfloor \frac{n}{2} \rfloor$ cuts (group 1), and the rest with the others (group 2).

(3) Recursively apply the same procedure to each of the two groups, until only a single agent is left. ✓

Each agent is guaranteed a *proportional* piece. Takes $O(n \log n)$ steps.

S. Even and A. Paz. A Note on Cake Cutting. *Discrete Applied Mathematics*, 7:285–296, 1984.

Envy-Free Procedures

Achieving *envy-freeness* is much harder than achieving proportionality:

- For $n = 2$ the problem is easy: cut-and-choose does the job.
- For $n = 3$ things are already quite complicated: we will see a solution involving *four simultaneously moving knives*.
- For $n = 4$, to date, no procedure producing *contiguous pieces* is known. Barbanel and Brams (2004), for example, give a moving-knife procedure requiring up to 5 cuts.
- For $n \geqslant 6$, to date, only procedures requiring an *unbounded* number of cuts are known (see e.g. Brams and Taylor, 1995).

J.B. Barbanel and S.J. Brams. Cake Division with Minimal Cuts. *Mathematical Social Sciences*, 48(3):251–269, 2004.

S.J. Brams and A.D. Taylor. An Envy-free Cake Division Protocol. *American Mathematical Monthly*, 102(1):9–18, 1995.

The Stromquist Procedure

Stromquist (1980) found an envy-free procedure for $n = 3$ producing *contiguous* pieces, though requiring four simultaneously *moving knifes:*

- A referee slowly moves a knife across the cake, from left to right (supposed to cut somewhere around the $1/3$ mark).
- At the same time, each agent is moving her own knife so that it would cut the righthand piece in half (wrt. her own valuation).
- The first agent to call "stop" receives the piece to the left of the referee's knife. The righthand part is cut by the middle one of the three agent knifes. If neither of the other two agents hold the middle knife, they each obtain the piece at which their knife is pointing. If one of them does hold the middle knife, then the other one gets the piece at which her knife is pointing. ✓

W. Stromquist. How to Cut a Cake Fairly. *American Mathematical Monthly*, 87(8):640–644, 1980.

Literature

See my lecture notes for a compact introduction to fair division.

Both the book by Brams and Taylor (1996) and that by Robertson and Webb (1998) cover the cake-cutting problem in great depth.

Our JAIR 2006 paper is a possible starting point for finding out more about the problem of fairly allocating indivisible goods.

U. Endriss. *Lecture Notes on Fair Division*. ILLC, University of Amsterdam, 2010.

S.J. Brams and A.D. Taylor. *Fair Division: From Cake-Cutting to Dispute Resolution*. Cambridge University Press, 1996.

J. Robertson and W. Webb. *Cake-Cutting Algorithms: Be Fair if You Can*. A.K. Peters, 1998.

U. Endriss, N. Maudet, F. Sadri and F. Toni. Negotiating Socially Optimal Allocations of Resources. *Journal of AI Research*, 25:315–348, 2006.

Conclusion

Computational Social Choice

Research can be broadly classified along two dimensions —

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

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

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

- algorithm design to implement complex mechanisms
- complexity theory to understand limitations
- logical modelling to fully formalise intuitions
- knowledge representation techniques to compactly model problems
- deployment in a multiagent system

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

Last Slide

- Tried to give an overview of COMSOC *by example*, concentrating on preference aggregation, voting, judgment aggregation, and fair division. Omitted e.g. matching and coalition formation.

- Nice topic, particularly for AI people. Still lots to do.

- A website where you can find out more about Computational Social Choice (workshops, mailing list, PhD theses, etc.):

  `http://www.illc.uva.nl/COMSOC/`

- These slides will remain available on the tutorial website, and more extensive materials can be found on the website of my Amsterdam course on Computational Social Choice:
  - `http://www.illc.uva.nl/~ulle/teaching/ecai-2010/`
  - `http://www.illc.uva.nl/~ulle/teaching/comsoc/`