Binary Aggregation with Integrity Constraints

Umberto Grandi

Institute for Logic, Language and Computation
University of Amsterdam

27 June 2012

Everything Starts From Paradoxical Situations

Elections in the U.S. and in many other countries are decided using the plurality rule: the candidate who gets the most votes win.

Assume that the preferences of the individuals in Florida are as follows:

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

Bush results as the winner of the election, but:

- Gore wins against any other candidate in pairwise election.
- Nader supporters have an incentive to manipulate.

What About Computer Science?

Suppose three agents in a multi-agent system need to decide whether to perform a collective decision A. The decision is performed if two parameters T_1 and T_2 exceed a given threshold. Consider the following situation:

	T_1	T_2	A
Agent 1	Yes	Yes	Yes
Agent 2	No	Yes	No
Agent 3	Yes	No	No
Majority	Yes	Yes	No

Should the agents perform action A or not?

- A majority of agents think the first parameter exceeds the threshold.
- A majority of agents think the second parameter exceeds the threshold.
- But: a majority of agents think action A should not be performed!!

Outline

1. A very brief introduction to Computational Social Choice
2. Part I: Binary aggregation with integrity constraints
3. Part II: A general definition of paradox
4. Part III: Lifting rationality assumptions and axiomatic properties
5. Last question: Can we avoid all paradoxes?

Computational Aspects of Collective Decision Making

Different social choice problems studied:

- Choosing a winner given individual preferences over candidates.
- Allocate resources to users in an optimal way.
- Finding a stable matching of students to schools.

Different computational techniques used:

- Algorithm design to implement complex mechanisms.
- Complexity theory to understand limitations.
- Knowledge representation techniques to compactly model problems.

Algorithmic Game Theory is a related research area: how games can be used in CS, and how CS can be used in the study of games.

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

Example I: Computational Complexity as a Barrier to Manipulation

Recall the initial example: Nader supporters have an incentive to manipulate the outcome of the election and switch to vote for Gore.

How difficult is it to manipulate an election?

For certain voting rules like plurality it is easy (=polynomial). In other cases, like in Australia where a sequential rule called STV is used, this problem is computationally hard (NP-complete):

Computational complexity can thus be a barrier to manipulation.

Bartholdi, Tovey, and Trick. The Computational Difficulty of Manipulating an Election. Social Choice and Welfare, 1989.

Faliszewski and Procaccia. AI's War on Manipulation: Are We Winning? AI Magazine, 2010.

Example II: Combinatorial Voting

When the set of alternatives has a combinatorial structure, representation of preferences may be exponential! Examples:

- Electing a committee (3 members out of 8 candidates: 112 combinations)
- Multiple referendums, deciding a menu.
- Multiple variables to describe alternatives.

CS provides tools for compact representation of preferences. But how to perform collective decision making? Vote sequentially? Vote on combinations? How to elicit preferences? How hard is to compute a winner?

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.

If you want to know more: COMSOC-2012 in Kraków, from 11-13 September.

Part I:
Binary Aggregation with Integrity Constraints

A Paradoxical Example

One more look at the initial example of three agents with sensors:

	T_1	T_2	A
Agent 1	Yes	Yes	Yes
Agent 2	No	Yes	No
Agent 3	Yes	No	No
Majority	Yes	Yes	No

Two questions:

- Why is this a paradox?
- Why does this happen?

Individual Rationality in Decision Theory

The problem:	Individual choosing over a set of alternatives $\mathcal{X}$
Rational behaviour:	Maximise a weak order over $\mathcal{X}$ (transitive, complete and reflexive binary relation)

- Linear orders to avoid ties
- Partial orders over large domains
- Acyclic relations defined from choice functions

Individual Rationality in Decision Theory

The problem:	Individual choosing over a set of alternatives $\mathcal{X}$
Rational behaviour:	Maximise a weak order over $\mathcal{X}$ (transitive, complete and reflexive binary relation)

- Linear orders to avoid ties
- Partial orders over large domains
- Acyclic relations defined from choice functions

Remark: we do not talk about uncertainties.


Many Rationalities?

Judges in a court (cf. judgment aggregation):

“O.J.Simpson is guilty” “O.J.Simpson wore bloody gloves”
“If O.J.Simpson wore the glove then he is guilty”

Rational judges?
Consistent and complete judgment sets

Many Rationalities?

Judges in a court (cf. judgment aggregation):

"O.J.Simpson is guilty" "O.J.Simpson wore bloody gloves"
"If O.J.Simpson wore the glove then he is guilty"

Rational judges?
Consistent and complete judgment sets

Committee deciding over multiple issues:

"Cut pensions" "Cut the number of MPs"
"Liberalise the market"

Rational members?
No political power to enforce all three austerity measures:
Ballots with at most 2 yes

Binary Aggregation with Integrity Constraints

"Everything is binary"

- Individuals express yes/no ballots over a finite set of issues $\mathcal{I}$
- A propositional language can be interpreted over ballots
- Rationality assumptions/integrity constraints are formulas in this language

Binary Aggregation

Ingredients:

- A finite set $\mathcal{N}$ of individuals
- A finite set $\mathcal{I} = \{1, \ldots, m\}$ of issues
- A boolean combinatorial domain: $\mathcal{D} = D_1 \times \cdots \times D_m$ with $|D_i| = 2$

Definition

An aggregation procedure is a function $F : \mathcal{D}^{\mathcal{N}} \to \mathcal{D}$ mapping each profile of ballots $\mathbf{B} = (\mathbf{B}_1, \ldots, \mathbf{B}_n)$ to an element of the domain $\mathcal{D}$.

Dokow and Holzman (2010), Grandi and Endriss (AAAI-2010)

Binary Aggregation

Ingredients:

- A finite set $\mathcal{N}$ of individuals
- A finite set $\mathcal{I} = \{1, \ldots, m\}$ of issues
- A boolean combinatorial domain: $\mathcal{D} = D_1 \times \cdots \times D_m$ with $|D_i| = 2$

Definition

An aggregation procedure is a function $F : \mathcal{D}^{\mathcal{N}} \to \mathcal{D}$ mapping each profile of ballots $\mathbf{B} = (\mathbf{B}_1, \ldots, \mathbf{B}_n)$ to an element of the domain $\mathcal{D}$.

Example: Three agents with sensors

- $\mathcal{N} = \{1, 2, 3\}$
- $\mathcal{I} = \{T_1, T_2, A\}$
- Individuals submit ballots in $\mathcal{D} = \{0, 1\}^3$

$B_1 = (0, 1, 0)$ the first individual think the action should not be performed.

Integrity Constraints

A propositional language $\mathcal{L}$ to express integrity constraints on $D = \{0,1\}^m$

- One propositional symbol for every issue: $PS = \{p_1, \ldots, p_m\}$
- $\mathcal{L}_{PS}$ closing under connectives $\wedge$, $\vee$,$\neg$, $\rightarrow$ the set of atoms PS

Given an integrity constraint IC $\in \mathcal{L}_{PS}$, a rational ballot is $B \in \mathrm{Mod}(\mathrm{IC})$

Integrity Constraints

A propositional language $\mathcal{L}$ to express integrity constraints on $D = \{0,1\}^m$

- One propositional symbol for every issue: $PS = \{p_1, \dots, p_m\}$
- $\mathcal{L}_{PS}$ closing under connectives $\wedge$, $\vee$,$\neg$, $\rightarrow$ the set of atoms PS

Given an integrity constraint IC $\in \mathcal{L}_{PS}$, a rational ballot is $B \in \mathrm{Mod}(\text{IC})$

Example: Three agents with sensors

Perform action A if both parameters exceed the thresholds.
Propositional constraint: $\text{IC} = (T_1 \wedge T_2) \rightarrow A$

Individual 1 submits $B_1 = (1,1,1)$:	B_1 satisfies IC ✓
Individual 2 submits $B_2 = (0,1,0)$:	$B_2 \models \text{IC}$ ✓
Individual 3 submits $B_3 = (1,0,0)$:	$B_3 \models \text{IC}$ ✓

Majority aggregation outputs $(1,1,0)$: IC not satisfied.

Every individual satisfies the same rationality assumption IC...
...what about the collective outcome?

Definition

A paradox is a triple $(F, \mathbf{B}, IC)$*, where:*

- F *is an aggregation procedure*
- $\mathbf{B} = (B_1, \ldots, B_n)$ *a profile*
- $IC \in \mathcal{L}_{PS}$ *an integrity constraint*

such that $B_i \models IC$ *for all* $i \in \mathcal{N}$ *but* $F(\mathbf{B}) \not\models IC$*.*

Part I:
A Common Definition of Paradox

Condorcet Paradox

Preference aggregation (Condorcet, 1785, Arrow, 1963) studies how to obtain a collective ranking of alternatives from individual preferences. Used in voting, political theory, and CS (e.g. aggregate rankings of search engines).

In 1785 Monsieur le Marquis de Condorcet pointed out that:

$$\begin{array}{c} \triangle \succ_1 \bigcirc \succ_1 \square \\ \square \succ_2 \triangle \succ_2 \bigcirc \\ \bigcirc \succ_3 \square \succ_3 \triangle \\ \hline \triangle \succ \bigcirc \succ \square \succ \triangle \end{array}$$

The collective ranking is a cycle!

Preference Aggregation

Linear order $<$ over alternatives $\mathcal{X}$ $\Leftrightarrow$ Ballot $B_{\leqslant}$ over issues $\mathcal{I} = \{ab \mid a \neq b \in \mathcal{X}\}$

$\mathrm{IC}_{<}$ encodes the rationality assumption for decision theory:

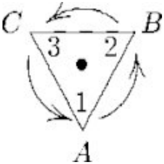
Irreflexivity: $\neg p_{aa}$ for all $a \in \mathcal{X}$

Completeness: $p_{ab} \vee p_{ba}$ for all $a \neq b \in \mathcal{X}$

Transitivity: $p_{ab} \wedge p_{bc} \rightarrow p_{ac}$ for $a, b, c \in \mathcal{X}$ pairwise distinct

Condorcet Paradox Revisited

	△○	○□	△□
Agent 1	1	1	1
Agent 2	0	1	0
Agent 3	1	0	0
Maj	1	1	0



Our definition of paradox:

- F is issue by issue majority rule
- the profile is the one described in the table
- IC that is violated is $p_{\triangle\bigcirc} \wedge p_{\bigcirc\square} \rightarrow p_{\triangle\square}$

Judgment Aggregation

Judgment Aggregation (Kornhauser and Sager, 1986, List and Pettit, 2002) studies the aggregation of judgments over a set of correlated propositions.

The discursive dilemma is a paradoxical situation that can arise when a court of judges decides on more than three correlated issues:

> *Suppose an individual is considered guilty if she has broken a contract and the contract was valid. Call α the proposition "the contract was broken", β the proposition "the contract was valid". The individual is guilty if and only if $\alpha \wedge \beta$ is judged true. Consider the following situation...*

Kornauser and Sager. Unpacking the court. Yale Law Journal, 1986.

Discursive Dilemma

	α	β	$\alpha \wedge \beta$
Agent 1	1	1	1
Agent 2	0	1	0
Agent 3	1	0	0
Majority	1	1	0!!

Our definition of paradox:

- F is issue by issue majority rule
- profile described in the table
- IC that is violated is $\neg(p_\alpha \wedge p_{\neg\beta} \wedge p_{(\alpha\wedge\beta)})$

Common feature: Three issues

The Common Structure of Paradoxes

Proposition (Grandi and Endriss, IJCAI-2011)

The majority rule does not generate a paradox with respect to IC if and only if IC is equivalent to a conjunction of clauses of size $\leqslant 2$.

$$\mathcal{IC}(Maj) = \text{2-CNF}$$

Common feature of all paradoxes:
clauses of size 3 are not lifted by majority

Part II:
Lifting Rationality Assumptions

Collective Rationality

Definition

F is collectively rational (CR) for IC $\in \mathcal{L}_{PS}$ if for all profiles $\mathbf{B}$ such that $\mathbf{B}_i \models$ IC for all $i \in N$ then $F(\mathbf{B}) \models$ IC.

F lifts the rationality assumption given by IC
from the individual to the collective level.

$$\boxed{\mathcal{CR}[\mathcal{L}] = \{F : \mathcal{D}^{\mathcal{N}} \to \mathcal{D} \mid \mathcal{N} \text{ is finite and } F \text{ is CR for all IC} \in \mathcal{L}\}}$$

where $\mathcal{L} \subseteq \mathcal{L}_{PS}$ is a sublanguage

Language

Definition

A *language for integrity constraints* *is a subset $\mathcal{L} \subseteq \mathcal{L}_{PS}$ that is closed under conjunctions and logical equivalence.*

- if F is CR wrt. φ and to ψ then is CR wrt. $\varphi \wedge \psi$
- if F is CR wrt. φ then it is so for every equivalent formulas

Axioms

Aggregation procedures have been studied using the axiomatic method, listing axioms as desirable properties of the functions.

Classical axioms from social choice theory can be translated in this framework:

Unanimity (U): For any profile $\mathbf{B} \in X^N$ and any $x \in \{0, 1\}$, if $\mathbf{B}_{i,j} = x$ for all $i \in N$, then $F(\mathbf{B})_j = x$.

Independence (I): For any issue $j \in \mathcal{I}$ and any two profiles $\mathbf{B}, \mathbf{B}' \in X^N$, if $\mathbf{B}_{i,j} = \mathbf{B}'_{i,j}$ for all $i \in N$, then $F(\mathbf{B})_j = F(\mathbf{B}')_j$.

...

Languages and Axioms

Several languages for integrity constraints:

- *cubes*: conjunctions
- *k-pclauses*: positive disjunctions of size $\leqslant k$
- *XOR*: conjunctions of $p \leftrightarrow \neg q$
- ...

Several axioms to classify aggregation procedures:

- Unanimity (U): For any profile $\mathbf{B} \in X^N$ and any $x \in \{0, 1\}$, if $\mathbf{B}_{i,j} = x$ for all $i \in N$, then $F(\mathbf{B})_j = x$.
- Independence, Neutrality...

Characterisation Results

Different lists of axioms AX define classes of functions:

$$\mathcal{F}_{\mathcal{L}}[\mathrm{AX}] \;=\; \{F : \mathcal{D}^{\mathcal{N}} \to \mathcal{D} \mid F_{\restriction \mathrm{Mod}(\mathsf{IC})^N} \text{ sat. } \mathrm{AX} \text{ for all } \mathsf{IC} \in \mathcal{L}\}$$

Recall that the class of CR procedures for a language $\mathcal{L}$ is:

$$\mathcal{CR}[\mathcal{L}] \;=\; \{F : \mathcal{D}^{\mathcal{N}} \to \mathcal{D} \mid F \text{ is CR for all IC} \in \mathcal{L}\}$$

What we want:

$$\mathcal{CR}[\mathcal{L}] = \mathcal{F}_{\mathcal{L}}[\mathrm{AX}]$$

Characterisation Results: Examples

Cubes (conjunctions of literals) are lifted iff the procedure satisfies unanimity:

Proposition

$\mathcal{CR}[cubes] = \mathcal{F}_{cubes}[Unanimity]$.

Similar results can be proven for language of equivalences (issue-neutrality), XOR formulas (domain-neutrality), positive implications (neutral-monotonicity).

Characterisation Results: Examples

Cubes (conjunctions of literals) are lifted iff the procedure satisfies unanimity:

Proposition

$\mathcal{CR}[\textit{cubes}] = \mathcal{F}_{\textit{cubes}}[\textit{Unanimity}]$.

Similar results can be proven for language of equivalences (issue-neutrality), XOR formulas (domain-neutrality), positive implications (neutral-monotonicity).

For the axioms of independence we prove instead a negative result:

Proposition

There is no language $\mathcal{L} \subseteq \mathcal{L}_{PS}$ *such that* $\mathcal{CR}[\mathcal{L}] = \mathcal{F}_{\mathcal{L}}[\mathrm{I}]$.

The same result holds for the axiom of anonymity and monotonicity.

Part IV:
Can We Avoid All Paradoxes?

How to Avoid all Paradoxes?

A generalised dictatorship copies the ballot of a (possibly different) individual (aka local dictatorships, positional dictatorships, rolling dictatorships):

Proposition

F is collectively rational with respect to all IC in $\mathcal{L}_{PS}$ if and only if F is a generalised dictatorship.

This class includes:

- Classical dictatorships $F(B_1, \ldots, B_n) = B_i$ for $i \in \mathcal{N}$
- Average Voter Rule: map $(B_1, \ldots, B_n)$ to the ballot B_i that minimises the sum of the Hamming distance to the others (the "average voter"). An interesting procedure!

The Average Voter Rule

Definition

The average voter rule (AVR) *chooses the individual ballot that minimises the sum of the Hamming distance* $H(B, B') = \sum_{j \in \mathcal{I}} |b_j - b'_j|$ *to all other individual ballots:*

$$\mathrm{AVR}(\mathbf{B}) = \operatorname*{argmin}_{\{B_i | i \in \mathcal{N}\}} \sum_{s \in \mathcal{N}} H(B_i, B_s),$$

The AVR shares interesting axiomatic properties:

Proposition

The AVR *satisfies a non-resolute version of anonymity, unanimity and monotonicity. It does not satisfy independence.*

Complexity of the AVR

$\textsc{WinDet}^\star(F)$	
Instance:	Integrity constraint IC, profile $\mathbf{B}$, issue j.
Question:	Is there a $B \in F(\mathbf{B})$ such that $B_j = 1$?

Fact (very easy proof)

$\textsc{WinDet}^\star(\text{AVR})$ *is in P.*

WinDet$^\star(F)$
Instance: Integrity constraint IC, profile $\mathbf{B}$, issue j.
Question: Is there a $B \in F(\mathbf{B})$ such that $B_j = 1$?

Fact (very easy proof)

WinDet*(AVR) *is in P.*

Manipulable$^\star(F)$
Instance: IC, ballot B, partial profile $\mathbf{B}_{-i} \in \mathrm{Mod}(\mathsf{IC})^{|\mathcal{N}|-1}$.
Question: Is there B' s.t. $H(B, F^\star(B', \mathbf{B}_{-i})) < H(B, F^\star(B, \mathbf{B}_{-i}))$?

Proposition

Manipulability*(AVR) is in P.*

More Results

More results can be obtained in this framework:

1. Computational complexity of safe aggregation: Π_2^p-complete in most cases we considered.
2. Classical impossibility results revisited: new flexible proof of Arrow's Theorem, agenda safety results in judgment aggregation.
3. More rules to avoid paradoxes: premise-based procedure and distance-based procedure.
4. Many more characterisation results on lifting rationality assumptions.

More information and detailed proofs:

Umberto Grandi, Binary Aggregation with Integrity Constraints, PhD Thesis, ILLC, 2012.

Grandi and Endriss. Binary Aggregation with Integrity Constraints, *IJCAI-2011*.

Conclusions

Binary aggregation is the study of the aggregation of individual expressions described by means of binary variables.

- Unifying framework for paradoxes.
- Systematic study of collective rationality.
- Application in preference/judgment aggregation & co.
- Suggests practical aggregation procedures.

Future work:

- Generalisation to non-binary domains.
- Application to voting theory.
- Aggregation of preferential dependencies and CP-nets (Airiau Et Al, 2011).
- Study the parametrized complexity of safe aggregation.

Thanks for your attention!