Computational Social Choice: Voting Theory, Automated Reasoning, Explainability

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

Plan for Today

Yesterday we saw how we can use *axioms* to motivate the use of specific *voting rules* to take collective decisions. That's classical SCT.

Today we will explore two strands of modern COMSOC:

- *Automated Reasoning for Social Choice*

  Using computers (and specifically: SAT solvers) to support scientists in reasoning about social choice scenarios.

- *Explainability in Social Choice*

  Providing explanations for why a given collective choice is the right one, by linking it directly to axioms that can justify it.

Formal Model of Voting

Fix a finite set $X = \{a, b, c, \ldots\}$ of *alternatives*, with $|X| = m \geqslant 2$.

Let $\mathcal{L}(X)$ denote the set of all strict linear orders R on X. We use elements of $\mathcal{L}(X)$ to model (true) *preferences* and (declared) *ballots*.

Each member i of a finite set $N = \{1, \ldots, n\}$ of *voters* supplies us with a ballot R_i, giving rise to a *profile* $\boldsymbol{R} = (R_1, \ldots, R_n) \in \mathcal{L}(X)^n$.

Today we restrict attention to *voting rules* that are *resolute:*

$$F : \mathcal{L}(X)^n \to X$$

Exercise: *How to adapt this definition for arbitrary voting rules?*

Formalising Axioms

Let us formalise some of the definitions of axioms we saw yesterday:

- F satisfies the *Pareto Principle* if $N^{\boldsymbol{R}}_{x \succ y} = N$ implies $F(\boldsymbol{R}) \neq y$.
- F is *strategyproof* (or: immune to manipulation) if for no $i \in N$ there are a profile $\boldsymbol{R}$ (including the "truthful preference" R_i of i) and a ranking R'_i (representing an "untruthful" ballot of i) s.t.:

  $$F(R'_i, \boldsymbol{R}_{-i}) \text{ is ranked above } F(\boldsymbol{R}) \text{ according to } R_i$$

- F is *surjective* if for every alternative $x \in X$ there is a profile $\boldsymbol{R}$ such that $F(\boldsymbol{R}) = x$. So no x is excluded from winning *a priori*.

Notation: $N^{\boldsymbol{R}}_{x \succ y}$ is the set of voters ranking x above y in profile $\boldsymbol{R}$.

Notation: $(R'_i, \boldsymbol{R}_{-i})$ is what we get when in $\boldsymbol{R}$ we replace R_i by R'_i.

The Gibbard-Satterthwaite Theorem

F is *dictatorial* if there exists an $i \in N$ such that $F(\boldsymbol{R}) = \mathrm{top}(R_i)$ holds for every profile $\boldsymbol{R}$. Then voter i is the dictator.

Exercise: *How many different rules are there that are dictatorial?*

Theorem 1 (Gibbard-Satterthwaite) *There exists no resolute rule for $\geqslant 3$ alternatives that is surjective, strategyproof, and nondictatorial.*

Remarks:

- Common confusion: dictatorship *vs.* "local dictatorship"
- The theorem does not hold for $m = 2$ alternatives. (*Why?*)
- The theorem is trivially true for $n = 1$ voter. (*Why?*)

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

M.A. Satterthwaite. Strategy-proofness and Arrow's Conditions. *Journal of Economic Theory*, 1975.

Proving G-S

G-S is a deep result that long proved elusive:

- People tried and failed to design strategyproof rules for centuries.
- After Arrow's seminal impossibility theorem (for different axioms) a result *à la* G-S seemed to be "in the air".
- It still took two decades to find the right formulation and prove it.
- The original proofs are hard to digest (the original proof of Arrow's impossibility even was wrong—though not the theorem itself).

Today the proof of G-S is well understood (see expository paper below). But new results of this kind are still hard to identify and then prove.

K.J. Arrow. *Social Choice and Individual Values*. John Wiley and Sons, 2nd edition, 1963. First edition published in 1951.

U. Endriss. Logic and Social Choice Theory. In A. Gupta and J. van Benthem (eds.), *Logic and Philosophy Today*. College Publications, 2011.

Automated Reasoning for Social Choice

Thus: *Need much better methodology to reason about social choice!*

Maybe *automated reasoning*, as studied in AI, can help? *Yes!*

In particular, *SAT solvers* have been used successfully to prove a wide range of (impossibility) theorems in SCT (and related areas):

- automated *verification* of classical results
- automated *proofs* of new theorems
- automated *discovery* of new theorems

P. Tang and F. Lin. Computer-aided Proofs of Arrow's and other Impossibility Theorems. *Artificial Intelligence*, 2009.

C. Geist and D. Peters. Computer-Aided Methods for Social Choice Theory. In U. Endriss (ed.), *Trends in Computational Social Choice*. AI Access, 2017.

Outline of the Approach

The smallest nontrivial case of G-S is that of $n = 2$ voters and $m = 3$ alternatives. If we can prove it, larger cases will be unsurprising.

So focus on this base case, using this approach:

- express the requirements on F in logic
- show that the resulting formula is not satisfiable

If we can express our requirements in *propositional (boolean) logic*, then we can use (very efficient!) *SAT solvers* for the second step.

Describing Voting Rules in Logic

Consider the propositional (boolean) language with this set of variables:

$$\{\, p_{\boldsymbol{R},x} \mid \boldsymbol{R} \in \mathcal{L}(X)^n \text{ and } x \in X \,\}$$

Intuition: Variable $p_{\boldsymbol{R},x}$ is *true* iff we elect alternative x in profile $\boldsymbol{R}$.

Exercise: *Count the variables for $n = 2$ voters and $m = 3$ alternatives!*

Now assignments of truth values to variables correspond to voting rules.

Exercise: *This is almost true, but not quite. What is the problem?*

Voting Rules as Truth Assignments

Not every possible truth assignment corresponds to a voting rule.

We need to ensure at *least one alternative* is elected in each profile:

$$p_{\boldsymbol{R},a_1} \vee p_{\boldsymbol{R},a_2} \vee \cdots \vee p_{\boldsymbol{R},a_m} \quad \text{(for all profiles } \boldsymbol{R}\text{)}$$

We also need to ensure *at most one alternative* is elected:

$$\neg(p_{\boldsymbol{R},x} \wedge p_{\boldsymbol{R},y}) \quad \text{(for all profiles } \boldsymbol{R} \text{ and alternatives } x \neq y\text{)}$$

If φ_{rule} is the conjunction of all of these formulas, then there is a direct correspondence between models of φ_{rule} and resolute voting rules.

Axioms as Formulas

We now can add to our requirements by expressing axioms as formulas.

Here is the formula for *strategyproofness:*

$$\varphi_{\mathsf{sp}} = \bigwedge_{i \in N} \left(\bigwedge_{\boldsymbol{R} \in \mathcal{L}(X)^n} \left(\bigwedge_{\substack{\boldsymbol{R}' \in \mathcal{L}(X)^n \\ \text{s.t. } \boldsymbol{R} =_{-i} \boldsymbol{R}'}} \left(\bigwedge_{x \in X} \left(\bigwedge_{\substack{y \in X \\ \text{s.t. } i \in N^{\boldsymbol{R}}_{x \succ y}}} \neg \left(p_{\boldsymbol{R}, y} \wedge p_{\boldsymbol{R}', x} \right) \right) \right) \right) \right)$$

Exercise: *Understand the encoding! (Hint: $\boldsymbol{R}$ is the truthful profile.)*

Script to Generate the Master Formula

We need to determine whether the "*master formula*" is satisfiable:

$$\varphi \;=\; \varphi_{\text{rule}} \wedge \varphi_{\text{sp}} \wedge \varphi_{\text{sur}} \wedge \varphi_{\text{nd}}$$

Aside: φ is a disjunction of 1,445 clauses (over 108 variables).

Using the so-called *DIMACS format*, we can represent any given formula in CNF on the computer as a list of lists of integers.

Example: `[[1,-2,3], [-1,4]]` represents $(p \vee \neg q \vee r) \wedge (\neg p \vee s)$.

We omit all details, but it is clear that *writing a script* (say, in Python) to generate this representation of our master formula is possible.

For full details (to copy-paste) see the slides of my Amsterdam course.

U. Endriss. Slide set for "Advanced Topics in Computational Social Choice". ILLC, University of Amsterdam, 2021. Available at `http://bit.ly/adv-comsoc-21`.

Running the SAT Solver

We now can run the SAT solver on our master formula φ ...

Through a Python interface, it will looks something like this:

```
>>> cnf = cnfRule() + cnfSP() + cnfSur() + cnfND()
>>> solve(cnf)
'UNSAT'
```

So φ really is unsatisfiable! Thus, G-S for $n = 2$ and $m = 3$ is true! ✓

Discussion: *Does this count? Do we believe in computer proofs?*

Missing Pieces

We can proof-read our *Python script* just like we would proof-read a mathematical proof. And we can use multiple *SAT solvers* and check they agree. So we can have some confidence in the result.

But some pieces are still missing:

- *Why does the theorem hold?* This proof does not tell us.

  But SAT technology can help here as well: *MUS extraction*

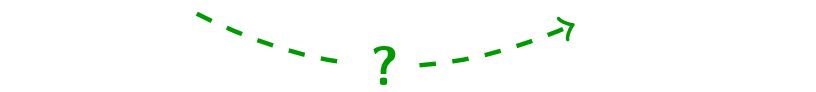
- *Does the theorem generalise to arbitrary* $n \geqslant 2$ *and* $m \geqslant 3$*?*

  Intuitively almost obvious, though technically not that easy.
  Basic idea: *induction* over both n and m

Explainability in Social Choice

How do you explain why a given collective decision is the right one?

The axiomatic method seems relevant, given that axioms can motivate voting rules, which in turn produce decisions when applied to profiles.



First Attempt

Can the axiomatic method help us *explain* / *justify* why some outcome might be *the right outcome* for a given profile? Maybe:

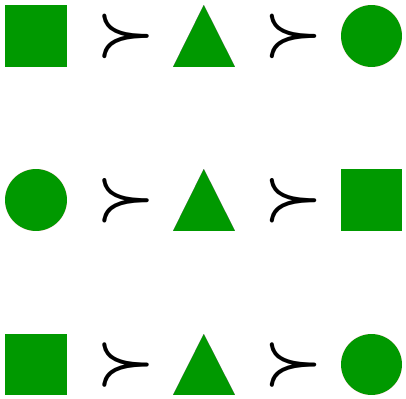
- suppose for profile $\boldsymbol{R}^\star$ we want to justify the choice of outcome $X^\star$
- suppose $F(\boldsymbol{R}^\star) = X^\star$ for some voting rule F
- suppose F is characterised by the set of axioms $\mathcal{A}$
- suppose we consider the axioms in $\mathcal{A}$ to be normatively appealing
- then we might say that we have an argument for choosing $X^\star$ in $\boldsymbol{R}^\star$

But there are a number of problems here:

- *few characterisation results*, some with *unattractive axioms*
- some appealing axioms also feature in *impossibility results*
- we hardly can expect our audience to *understand* the results used
- overkill: we just care about $\boldsymbol{R}^\star$, *not all profiles*

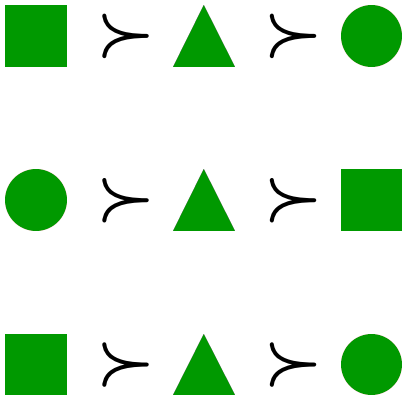
Can we instead justify outcomes by appealing to axioms directly?

Example



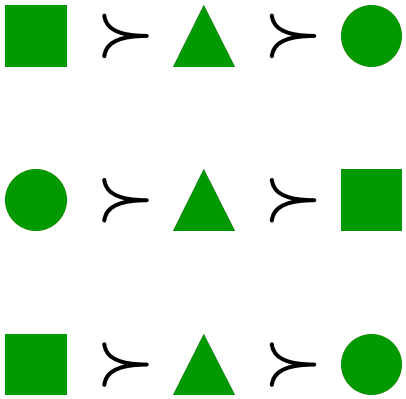
Exercise: *Can you think of a voting rule that makes* ■ *win?*

Example



Exercise: *Can you think of a voting rule that makes ▲ win?*

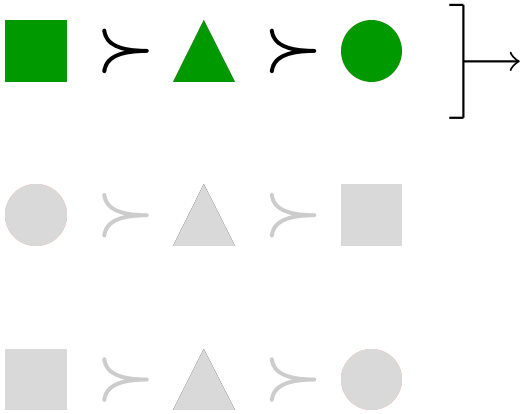
Example



What's a good outcome?

Why?

Example

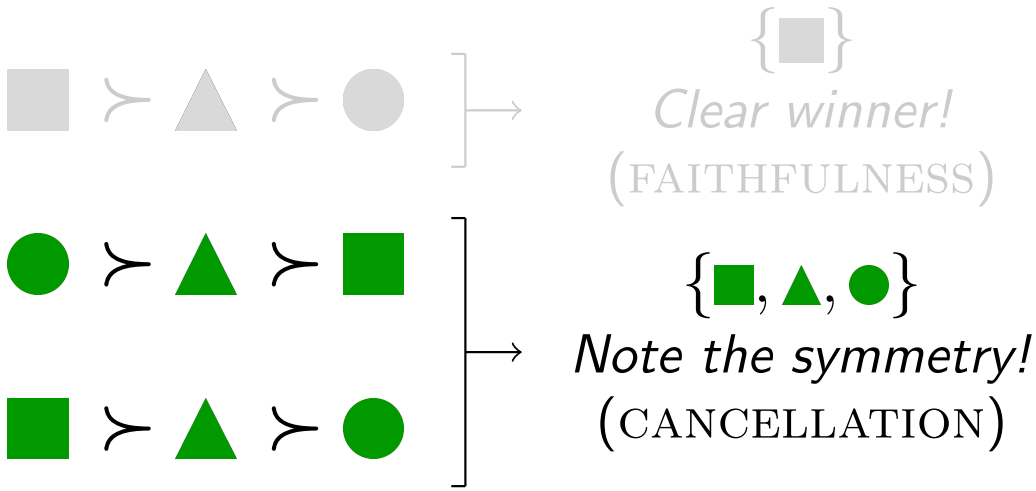


$\{\blacksquare\}$

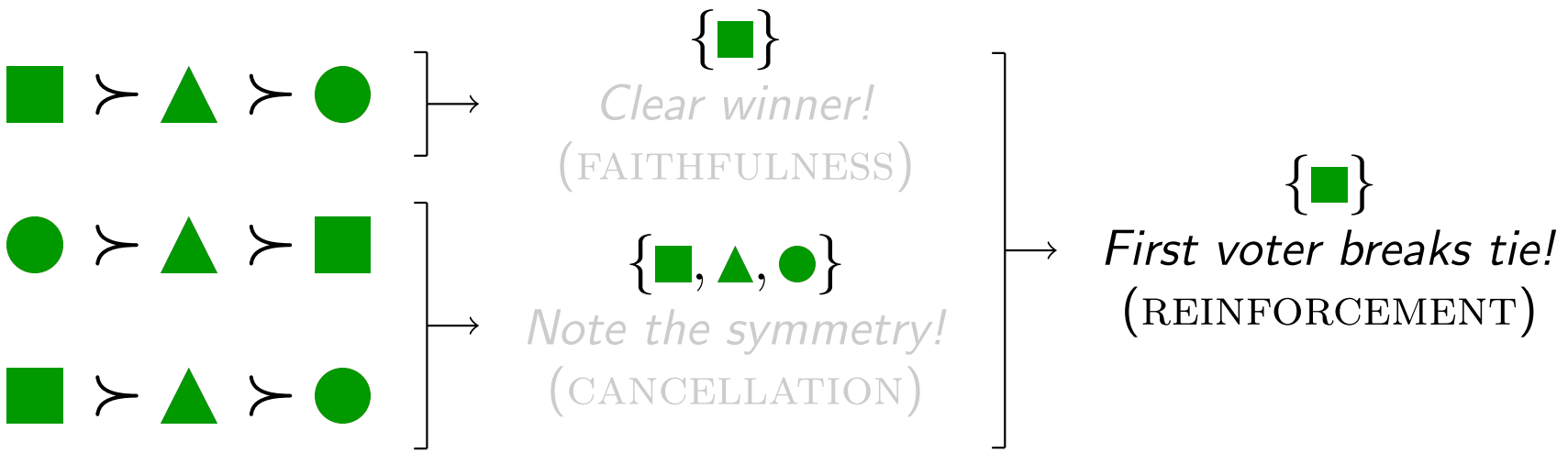
Clear winner!

(FAITHFULNESS)

Example


{■}
Clear winner!
(FAITHFULNESS)
{■, ▲, ●}
Note the symmetry!
(CANCELLATION)

Example


{■}
Clear winner!
(FAITHFULNESS)
{■, ▲, ●}
Note the symmetry!
(CANCELLATION)
{■}
First voter breaks tie!
(REINFORCEMENT)

Axioms: Interpretation and Instances

The *interpretation* of an axiom A is just a set of voting rules:

$$\mathbb{I}(A) \subseteq \text{ set of all voting rules}$$

Example: $\mathbb{I}(\text{NEU}) = \{\text{BORDA}, \text{PLURALITY}, \ldots, F_{4711}, \ldots\}$

An *instance* A' of axiom A (for a specific profile, etc.) is what you think it is, and itself an axiom, with $\mathbb{I}(A) = \bigcap_{A' \in \text{Inst}(A)} \mathbb{I}(A')$.

Example: $\text{Inst}(\text{PAR}) = \{$"*don't elect* c *in* $(abc^{[2]}, bca^{[5]})$*!*", $\ldots\}$

Proposal for a Definition

How can you justify outcome $X^\star$ given profile $\boldsymbol{R}^\star$ (with electorate $N^\star$) using as arguments only axioms from a (large!) corpus $\mathbb{A}$? Slogan:

$$\textit{Justification} = \textit{Normative Basis} + \textit{Explanation}$$

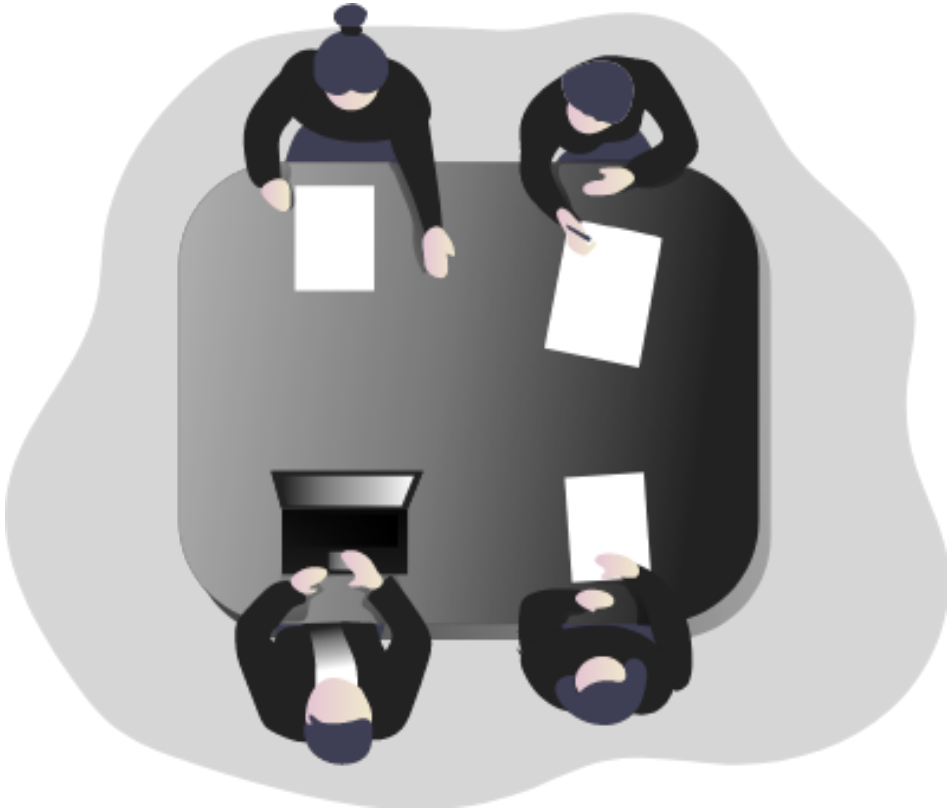
A pair $\langle \mathcal{A}^{\text{NB}}, \mathcal{A}^{\text{EX}} \rangle$ of sets of axioms is a *justification* if it satisfies:

- *Adequacy:* $\mathcal{A}^{\text{NB}} \subseteq \mathbb{A}$
- *Relevance:* $\mathcal{A}^{\text{EX}}$ is a set of instances of the axioms in $\mathcal{A}^{\text{NB}}$
- *Explanatoriness:* $F(\boldsymbol{R}^\star) = X^\star$ for all rules $F \in \bigcap_{A' \in \mathcal{A}^{\text{EX}}} \mathbb{I}(A')$ and this is not the case for any proper subset of $\mathcal{A}^{\text{EX}}$
- *Nontriviality:* $\bigcap_{A \in \mathcal{A}^{\text{NB}}} \mathbb{I}(A) \neq \emptyset$ (*some* rule satisfies all axioms)

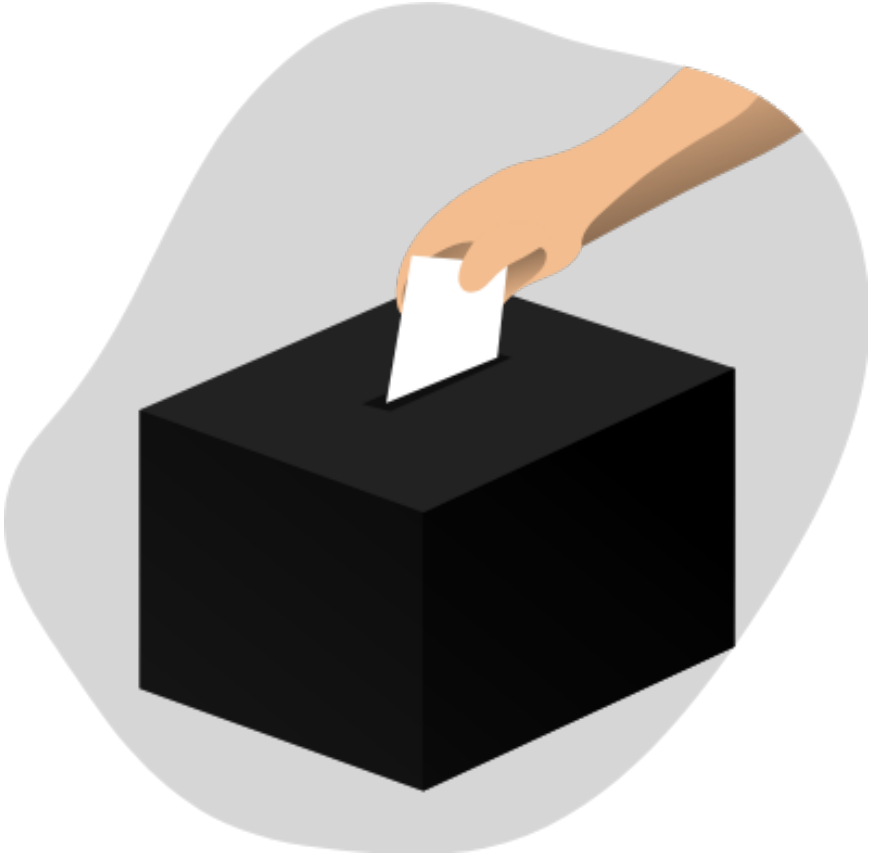
A. Boixel and U. Endriss. Automated Justification of Collective Decisions via Constraint Solving. AAMAS-2020.

Scenario 1: Confidence in Election Results

Scenario 2: Deliberation Support



Scenario 3: Justification Generation as Voting



Computing Justifications

We can encode *instances* of axioms in $\mathbb{A}$ in propositional logic just as we did earlier, and similarly for the *goal constraint* $F(\boldsymbol{R}^\star) \neq X^\star$.

Then use a *SAT solver* to check whether this set is *satisfiable:*

- If *yes*, no justification exists.
- If *no*, a justification $\langle \mathcal{A}^{\mathrm{NB}}, \mathcal{A}^{\mathrm{EX}} \rangle$ exists if these steps succeed:
  - Find MUS (*min. unsatisfiable subset*) including goal constraint. Let $\mathcal{A}^{\mathrm{EX}}$ be MUS $\setminus$ {goal constraint}.
  - Let $\mathcal{A}^{\mathrm{NB}}$ be the set of axioms in $\mathbb{A}$ with instances in $\mathcal{A}^{\mathrm{EX}}$. Check that $\mathcal{A}^{\mathrm{NB}}$ is *satisfiable* (for nontriviality).

Highly complex! But intractable tasks map to *well-studied problems* in automated reasoning. Challenge: generate only *relevant* instances.

→ online demo available: `http://bit.ly/xsoc-demo` ←

O. Nardi, A. Boixel, and U. Endriss. A Graph-Based Algorithm for the Automated Justification of Collective Decisions. AAMAS-2022.

Summary

To provide *computer support for social choice theorists*, we explored the idea of making use of tools from *automated reasoning:*

- Encoding voting rules and axioms in logic
- SAT solvers to identify unsatisfiable requirements

To approach the topic of *explainability in social choice*, we discussed the idea of *axiomatic justifications* for election outcomes:

- Scenarios: confidence building | deliberation support | voting
- Definition: justification = normative basis + explanation
- Algorithm: graph search + MUS generation + SAT solving

Exciting stuff ahead: experiments + voting on what Jérôme will teach