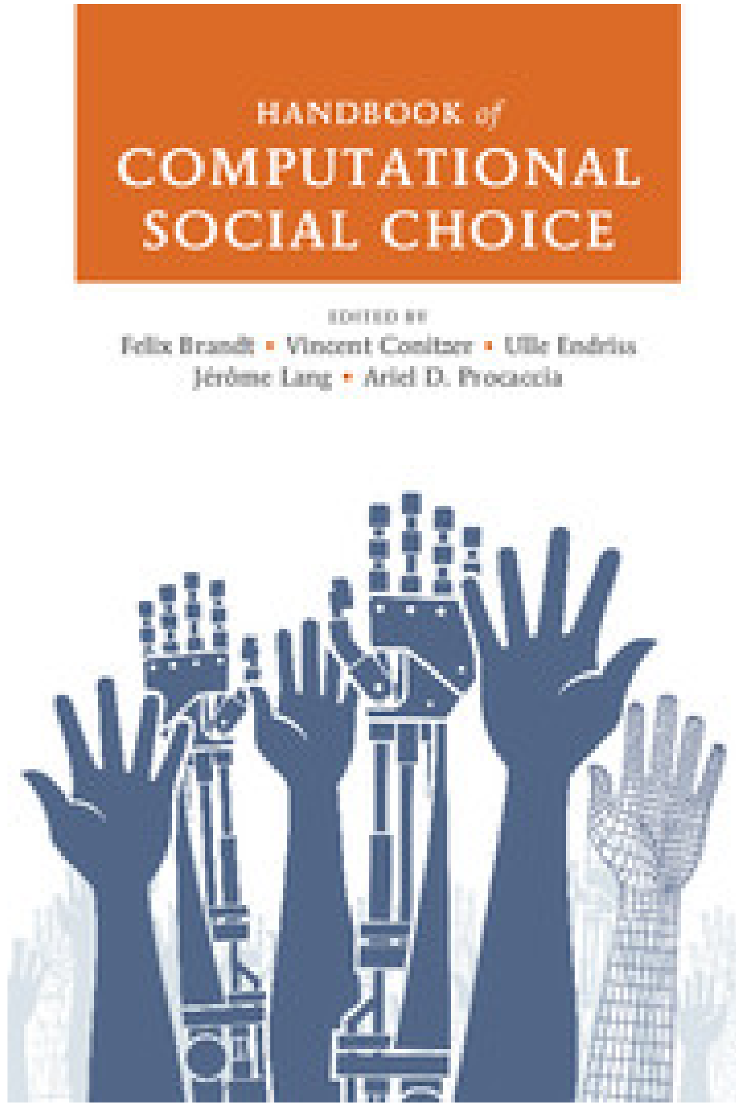
Automated Reasoning for Social Choice Theory

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

https://comsoc-community.org/



Talk Outline

The Research Agenda

- What social choice theorists do
- How computer scientists can help

The Case Study

- Scenario: designing matching markets
- Results: impossibility theorems
- Methodology: logic + algorithms

The Axiomatic Method in Economic Theory

When searching for a mechanism to transform individual preferences into societal decisions, we should start by clarifying our normative requirements (*axioms*): *fairness*, *efficiency*, *strategyproofness*, ...

Often impossible to satisfy all axioms. Famous examples:

- **Arrow's Theorem:** *For $m \geqslant 3$ alternatives, no preference aggregation rule is Paretian, independent, and nondicatorial.*
- **Gibbard-Satterthwaite Theorem:** *For $m \geqslant 3$ alternatives, no voting rule is strategyproof, onto, and nondictatorial.*
- **Roth's Theorem:** *For $n \geqslant 2$ agents on each side of the market, no matching mechanism is both stable and strategyproof.*

Such results provide crucial insights but are notoriously hard to prove!

Automated Reasoning

So establishing impossibility theorems is difficult. *Can AI help?* Yes!

Tang and Lin pioneered an exciting approach where we encode axioms as *propositional formulas* and use a *SAT solver* to prove unsatisfiability.

The approach has been used to find *new proofs* for known results, to discover *new results*, and to *uncover mistakes* in the literature.

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

SAT Solving

A *SAT solver* is a computer program to check whether a (very large) formula of *propositional logic* is satisfiable. Input typically in *CNF*.

Example: The formula $\varphi = (\neg p_1 \vee p_2) \wedge (p_1) \wedge (\neg p_2)$ is unsatisfiable.

```
>>> cnf = [[-1,2], [1], [-2]]
>>> solve(cnf)
'UNSATISFIABLE'
```

Case Study: Matching Markets

Scenario: Two groups of n *agents* each. Each agent ranks all the members of the other group. *Find a good matching!*

Applications: job markets, school admissions, kidney transplants

Would like a mechanism with good normative properties (*axioms*):

- *Stability:* never beneficial for two agents to leave the market
- *Strategyproofness:* never beneficial to misrepresent preferences
- *Fairness:* (for example) no advantage for one side of the market

The classic 1962 algorithm achieves stability, but treats the 'left' side of the market better than the 'right' side (not fair) and incentivises agents on the 'right' to lie (not strategyproof). *Can we do better?*

D. Gale and L. Shapley. College Admissions and the Stability of Marriage. *The American Mathematical Monthly*, 1962.

Encoding

For a fixed number of agents, we can encode axioms in propositional logic with variables $x_{p \triangleright (i,j)}$ ("*match* i *and* j *in profile* p"). Example:

$$\bigwedge_{p} \bigwedge_{i} \bigwedge_{j} \bigwedge_{i' \prec_j i} \bigwedge_{j' \prec_i j} \left(\neg x_{p \triangleright (i,j')} \vee \neg x_{p \triangleright (i',j)}\right)$$

Exercise: *What is the name of this axiom?*

Remark: For $n = 3$ agents on each side of the market, above formula is a conjunction of $419,904$ *clauses* (big, yet manageable).

An Impossibility Theorem

Axiom: call a mechanism *left/right-fair* if swapping the two sides of the market never changes the outcome. Can encode this as well.

Let's run a *SAT solver* on what we prepared:

```
>>> setDimension(3)
>>> cnf = cnfMechanism() + cnfStable() + cnfLeftRight()
>>> solve(cnf)
'UNSATISFIABLE'
```

So we obtain a new impossibility theorem!

Impossibility Theorem: *For $n \geqslant 3$ agents on each side of the market, no matching mechanism is both stable and left/right-fair.*

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

U. Endriss. Analysis of One-to-One Matching Mechanisms via SAT Solving: Impossibilities for Universal Axioms. AAAI-2020.

Computer Proofs

We can *proof-read the script* used to generate our formulas just as we would proof-read a paper. And we can use *multiple SAT solvers* and check they agree. So we can have *confidence* in the result.

Missing Pieces

But some pieces are still missing:

- *Does the theorem really generalise to arbitrary* $n \geqslant 3$*?*

  Clear for our case. But we can do better: *Preservation Theorem* identifies simple conditions on axioms licensing this generalisation.

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

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

U. Endriss. Analysis of One-to-One Matching Mechanisms via SAT Solving: Impossibilities for Universal Axioms. AAAI-2020.

A Formal Language for Axioms

Would like to have formal language with clear semantics (i.e., a logic) to express axioms, to be able to get results for entire families of axioms.

Agents $\ell_1, \ldots, \ell_n$ and $r_1, \ldots, r_n$. First-order logic with *sorts*, one for *profiles* and one for agent *indices*, with these basic ingredients:

- $p \triangleright (i, j)$ — in profile p, agents ℓ_i and r_j will get matched
- $j \succ^{\mathrm{L}}_{p,i} j'$ — in profile p, agent ℓ_i prefers r_j to $r_{j'}$ (also for R)
- $top^{\mathrm{L}}_{p,i} = j$ — in profile p, agent ℓ_i most prefers r_j (also for R)
- $p \sim^{\mathrm{L}}_i p'$ — profiles p and p' are ℓ_i-variants (also for R)
- $p \rightleftarrows p'$ — swapping sides in profile p yields profile p'
- $\forall_{\mathrm{P}} / \exists_{\mathrm{P}}$ and $\forall_{\mathrm{N}} / \exists_{\mathrm{N}}$ — quantifiers for variables of two sorts

Recall that axioms describe properties of mechanisms. So *truth* of a *sentence* φ in our logic is defined relative to a *mechanism* μ.

Example

$$\forall_{\mathrm{P}} p.\forall_{\mathrm{P}} p'.\forall_{\mathrm{N}} i.\forall_{\mathrm{N}} j.\forall_{\mathrm{N}} j' \,.\, \left[(j \succ^{\mathrm{L}}_{p,i} j' \,\wedge\, p \sim^{\mathrm{L}}_{i} p') \rightarrow \neg(p \rhd (i,j') \,\wedge\, p' \rhd (i,j))\right]$$

<u>Exercise:</u> *What is the name of this axiom?*

The Preservation Theorem

Call a mechanism *top-stable* if it always matches all mutual favourites.
Call an axiom *universal* if it can be written in the form $\forall \vec{x}.\varphi(\vec{x})$.

Preservation Theorem: *For every top-stable mechanism μ^+ of dimension $n > 1$ that satisfies a given set Φ of universal axioms there exists a top-stable mechanism μ of dimension $n-1$ that does the same.*

Proof idea: Construct larger profile in which extra agents most prefer each other and are least liked by everybody else.

Corollary: Enough to prove impossibility theorems for smallest n!

Proof Detail

Given an $(n-1)$-dimensional profile, construct an n-dimensional one, in which top-stability forces the extra agents ℓ_n and r_n to be matched:

$$
\begin{array}{lllllllllllll}
\ell_1 & : & \Box & \succ & \cdots & \succ & \Box & \succ & r_n & \qquad r_1 & : & \Box \succ \cdots \succ \Box \succ \ell_n \\
\ell_2 & : & \Box & \succ & \cdots & \succ & \Box & \succ & r_n & \qquad r_2 & : & \Box \succ \cdots \succ \Box \succ \ell_n \\
\vdots & & \vdots & & & & \vdots & & \vdots & \qquad \vdots & & \vdots \qquad \vdots \quad \vdots \\
\ell_{n-1} & : & \Box & \succ & \cdots & \succ & \Box & \succ & r_n & \qquad r_{n-1} & : & \Box \succ \cdots \succ \Box \succ \ell_n \\
\ell_n & : & r_n & \succ & \cdots & \succ & r_2 & \succ & r_1 & \qquad r_n & : & \ell_n \succ \cdots \succ \ell_2 \succ \ell_1
\end{array}
$$

Counterexample

Preservation Theorem might look trivial. *Doesn't this always hold?*
No: some axioms we can satisfy for large but not for small domains.

Suppose we want to design a mechanism under which at least one agent in each group gets assigned to their most preferred partner:

$$\forall_{\mathrm{P}} p . \exists_{\mathrm{N}} i . \forall_{\mathrm{N}} j . [\, (top^{\mathrm{L}}_{p,i} = j) \;\rightarrow\; (p \triangleright (i,j)) \,] \;\wedge$$

$$\forall_{\mathrm{P}} p . \exists_{\mathrm{N}} j . \forall_{\mathrm{N}} i . [\, (top^{\mathrm{R}}_{p,j} = i) \;\rightarrow\; (p \triangleright (i,j)) \,]$$

This is *not* universal! Mechanism exists for $n = 3$ but not for $n = 2$.

Exercise: *Explain why, and why not!*

Minimally Unsatisfiable Subsets

Given a (large) unsatisfiable set of formulas Φ, an *MUS* is a (small) unsatisfiable set $\Phi' \subseteq \Phi$ all proper subsets of which are satisfiable.

Intuitively, Φ' captures the essence of the unsatisfiability exhibited by Φ. If Φ' is reasonably small, one can understand *why* Φ is unsatisfiable.

MUS extraction is much harder a problem than satisfiability checking, but good tools exist nonetheless.

Another Impossibility Theorem

Recall this classic result:

Roth's Theorem: *For $n \geqslant 2$, no matching mechanism is both stable and two-way strategyproof (for incomplete preferences).*

Remark: In our model (with complete preferences) true only for $n \geqslant 3$.

We can use our approach to prove this stronger variant:

Impossibility Theorem: *For $n \geqslant 3$, no matching mechanism is both top-stable and two-way strategyproof (even in our model).*

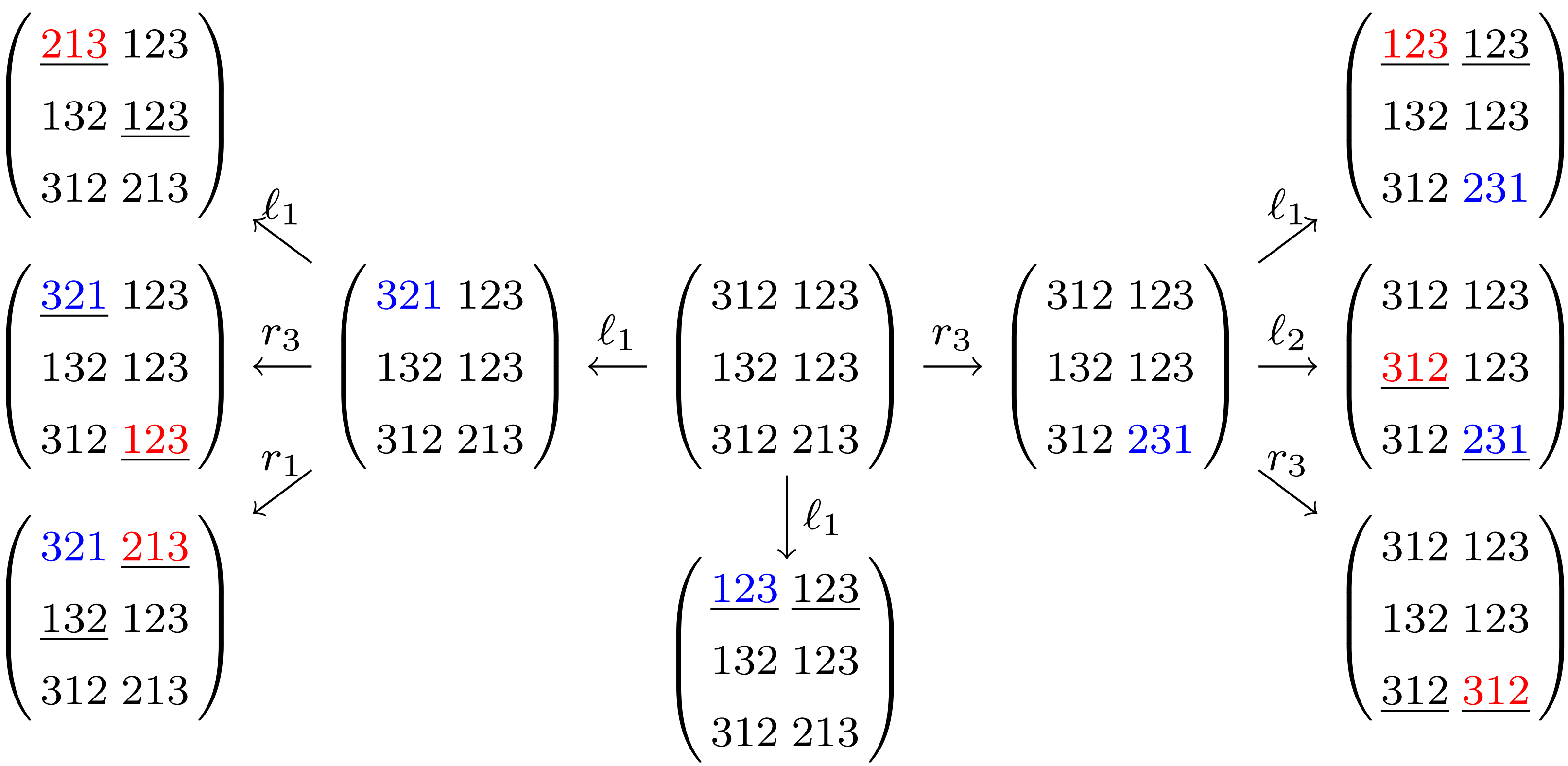
By the Preservation Theorem, we are done if the claim holds for $n = 3$.

Propositional formula has $4,805,568$ *clauses*. SAT solver says `UNSAT`.
Luckily, MUS has just 23 *clauses*. Can turn this into readable proof!

A.E. Roth. The Economics of Matching: Stability and Incentives. *Mathematics of Operations Research*, 7:617–628, 1982.

Human-Readable Proof of Base Case

Found MUS of 23 clauses, referencing 10 profiles. Proof visualisation:



Top-stability forces underlined matches. Colours indicate manipulation opportunities to be ruled out by SP. No matching left for centre profile.

Last Slide

By the *Preservation Theorem*, for top-stable mechanisms and universal axioms, proving impossibilities can be automated. Specific results:

- Impossible to get *stability* and *left/right-fairness*.
- Impossible to get *top-stability* and *two-way strategyproofness*.

Instance of a *broader research agenda* to use automated reasoning to support research in economic theory, also *beyond impossibilities:* axiom independence, designing mechanisms, outcome justification, ...

U. Endriss. Analysis of One-to-One Matching Mechanisms via SAT Solving: Impossibilities for Universal Axioms. AAAI-2020.

U. Endriss. Automated Reasoning for Social Choice Theory. Hands-on tutorial taught at AAMAS-2023. Slides and code available at bit.ly/tut7aamas.