Automated Search for Impossibility Theorems in Social Choice Theory: Ranking sets of Objects

Christian Geist, Ulle Endriss — 2011

pauline.baanders@student.uva.nl

October 11, 2021

Outline

1. Related Work
   Kannai-Peleg Theorem and work of Tang & Lin.
2. Preliminaries
   Interpretation of sets, orders on alternatives and their subsets.
3. Kannai-Peleg Theorem
   Seminal impossibility theorem.
4. Method
   Preservation Theorem, formulation in propositional logic and automated search.
5. Results
6. Conclusion

C. Geist and U. Endriss. Automated Search for Impossibility Theorems in Social Choice Theory: Ranking Sets of Objects. *Journal of Artif. Intell. Research*, 2011.

Related Work I

Kannai and Peleg (1984):

- First to address problem on itself, using axiomatic method.
- Two axioms: dominance and independence.
- Impossibility derived for at least six objects.

Y. Kannai and B. Peleg. A Note on the Extension of an Order on a Set to the Power Set. *Journal of Economic Theory*, 1984.

Related Work II

Tang and Lin (2009):

- Automatically re-prove of Arrow's Theorem.
- Use of lemmas to reduce theorem to special case (two agents, three alternatives).
- Reformulation in propositional logic, SAT solver to show inconsistency.
- Method extended for verification of other results.
- Starting point for the discussed paper.

P. Tang and F. Lin. Computer-Aided Proofs of Arrow's and Other Impossibility Theorems. *Artif. Intell.*, 2009.

Preliminaries

1. How to interpret sets?
   - Complete uncertainty;
   - Opportunity sets;
   - Sets as final outcomes
2. Notation
   - X: (finite) set of alternatives;
   - $\mathcal{X} := 2^X \setminus \{\emptyset\}$: nonempty subsets of X;
   - $\dot{\geq}$: linear order on X;
   - $\succeq$: weak order on $\mathcal{X}$. ($\sim$ to denote indifference.)

Kannai-Peleg Theorem

Two axioms: dominance or Gärdenfors principle (GF) and independence.
GF consists of two parts:

$$\begin{aligned} &(\texttt{GF1}) \quad \forall x \in X, \forall A \in \mathcal{X}: \quad ((\forall a \in A) x \dot{>} a) \implies A \cup \{x\} \succ A. \\ &(\texttt{GF2}) \quad \forall x \in X, \forall A \in \mathcal{X}: \quad ((\forall a \in A) x \dot{<} a) \implies A \cup \{x\} \prec A. \end{aligned}$$

For independence:

$$(\texttt{IND}) \;\; \forall A, B \in \mathcal{X}, \forall x \in X \backslash (A \cup B): \; A \succ B \Rightarrow A \cup \{x\} \succeq B \cup \{x\}.$$

Preservation Theorem

For the Preservation Theorem to hold axioms need to be formulated in specific language:

- Many-sorted logic. Sorts: elements (ε) and sets (σ).
- Example:

$$(\mathtt{IND})\ \ \forall_\sigma A \forall_\sigma B \forall_\varepsilon x[(x \notin (A \cup B) \wedge A \succ B) \rightarrow A \cup \{x\} \succeq B \cup \{x\}].$$

Further, in this language, the axioms need to satisfy a particular syntactic structure; existentially set-guarded.

Preservation Theorem

Corollary 1 (Universal Reduction Step). *Let* Γ *be a set of ESG (or ESG-equivalent) axioms and let* $n \in \mathbb{N}$ *be a natural number. If, for any linearly ordered set* Y *with* n *elements, there exists no binary relation on* $\mathcal{Y} = 2^Y \backslash \{\emptyset\}$ *satisfying* Γ*, then also for any linearly ordered set* X *with more than* n *elements there is no binary relation on* $\mathcal{X} = 2^X \backslash \{\emptyset\}$ *that satisfies* Γ.

Formulation in Propositional Logic

- Formulas must be in conjunctive normal form (CNF).
- Two kinds of propositions: $w_{A,B}$ and $l_{x,y}$.

Example, again using the independence axiom:

$$(\texttt{IND})\ \forall A, B \in \mathcal{X}, \forall x \in X \backslash (A \cup B) :\ A \succ B \Rightarrow A \cup \{x\} \succeq B \cup \{x\}$$

$$= \bigwedge_{A,B \in \mathcal{X}} \bigwedge_{x \in X \backslash (A \cup B)} [(w_{A,B} \wedge \neg w_{B,A}) \rightarrow w_{A \cup \{x\}, B \cup \{x\}}]$$

$$= \bigwedge_{A,B \in \mathcal{X}} \bigwedge_{x \in X \backslash (A \cup B)} [\neg w_{A,B} \vee w_{B,A} \vee w_{A \cup \{x\}, B \cup \{x\}}].$$

Instantiation of Axioms on a Computer

- The instantiation on the computer can be done via enumeration of the alternatives and subsets of alternatives.
- Once such an enumeration is established the operations on sets become operations on the corresponding numbers.

Note

Authors mention difficulty of transforming axioms with functions, e.g., the function to generate the singleton set $\{\cdot\} : X \to \mathcal{X}$. To me it is not clear what the difficulty is in such cases.

Automated search I

So far, we know how to construct a formula, denote it by φ, that represents an instance of an impossibility theorem, for the problem of ranking sets of objects.
Further, the authors propose an exhaustive and fully automated impossibility theorem search. Using a set of 20 axioms and up to 8 alternatives.

Automated Search II

For such an exhaustive search the order in which the instances are checked is highly relevant:

1. If a set of axioms is incompatible for n alternatives, then the set is also incompatible for more than n alternatives.
2. If a set of axioms is incompatible for n alternatives, then any superset of these axioms is also incompatible for n alternatives.
3. As the number of variables increases exponentially, the search should start checking the smaller domains.
4. Further, the authors empirically found that alternating between testing smaller sets of axioms (taking less time) and bigger sets of axioms (potentially eliminating more instances) led to the best performance.

Results

A set of 84 minimal impossibility theorems were found; some known, some straightforward, others new.
A particular interesting result is the incompatibility of a set of axioms that are shown to characterise the min-max ordering (Bossert et al., 2000), which is defined as:

$$A \succ_{mnx} B \iff [\min(A) \dot{>} \min(B) \vee (\min(A) = \min(B) \wedge \max(A) \dot{\geq} \max(B))].$$

In a paper by Arlegi (2003) it was already shown that this characterization was flawed. But the impossibility theorem of the conjunction of the axioms was not known.

Conclusion

Method for automatically finding impossibility theorems for ranking sets of objects, i.e., the problem how to extend preferences over individual objects to nonempty subsets. The method consists of three components:

1. The universal reduction step. This is presented as a corollary of the Preservation Theorem. The corollary entails that if a set of axioms is incompatible on a specific domain size n, then the set is also incompatible for any domain size larger than n. The conditions on the axioms for this result to apply are syntactical.
2. A method to translate the axioms into propositional formulas and instantiate them on a computer in such a way that they can be processed by a SAT solver.
3. A scheduling algorithm to decide how the search space, of axiom combinations for different domain sizes, should be explored.