



UvA-DARE (Digital Academic Repository)

Proving Classical Theorems of Social Choice Theory in Modal Logic

Ciná, G.; Endriss, U.

DOI

[10.1007/s10458-016-9328-6](https://doi.org/10.1007/s10458-016-9328-6)

Publication date

2016

Document Version

Final published version

Published in

Autonomous Agents and Multi-Agent Systems

License

CC BY

[Link to publication](#)

Citation for published version (APA):

Ciná, G., & Endriss, U. (2016). Proving Classical Theorems of Social Choice Theory in Modal Logic. *Autonomous Agents and Multi-Agent Systems*, 30(5), 963-989.
<https://doi.org/10.1007/s10458-016-9328-6>

General rights

It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations

If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: <https://uba.uva.nl/en/contact>, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

Proving classical theorems of social choice theory in modal logic

Giovanni Ciná¹  · Ulle Endriss¹

Published online: 20 February 2016

© The Author(s) 2016. This article is published with open access at Springerlink.com

Abstract A number of seminal results in the field of social choice theory demonstrate the difficulties of aggregating the preferences of several individual agents for the purpose of making a decision together. We show how to formalise three of the most important impossibility results of this kind—Arrow’s Theorem, Sen’s Theorem, and the Muller–Satterthwaite Theorem—by using a modal logic of social choice functions. We also provide syntactic proofs of these theorems in the same logic. While prior work has been successful in applying tools from logic and automated reasoning to social choice theory, this is the first human-readable formalisation of the Arrovian framework allowing for a direct derivation of the main impossibility theorems of social choice theory. This is useful for gaining a deeper understanding of the foundations of collective decision making, both in human society and in groups of autonomous software agents.

Keywords Social choice theory · Preferences · Arrow’s Theorem · Modal logic

1 Introduction

Social choice theory is the study of mechanisms for collective decision making [35]. This includes voting rules as mechanisms to collectively make political decisions, and consequently social choice theory is chiefly associated with the disciplines of political science

This is an extended version of a paper appearing in the proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems [11]. That paper included the formalisation and proof of one of the three theorems covered here, namely Arrow’s Theorem.

✉ Giovanni Ciná
giovanni.cina88@gmail.com

Ulle Endriss
ulle.endriss@uva.nl

¹ Institute for Logic, Language and Computation, University of Amsterdam, Amsterdam, The Netherlands

and economics. But similar mechanisms can also be used to make decisions in multiagent systems, to coordinate the actions of individual agents, to resolve conflicts between them, and to bundle their information and expertise [8]. Closely related applications of social choice theory in computer science furthermore include recommender systems [29], Internet search engines [2], and crowdsourcing [21].

This widening of the scope of social choice theory has renewed interest in the formal foundations of the field. As we are designing ever more specialised social choice mechanisms for novel types of tasks, better tools to analyse the formal properties of these mechanisms are needed. Specifically, there is now a growing literature on the formal verification of social choice mechanisms by means of logical modelling and the use of techniques from automated reasoning [1, 5, 9, 12, 13, 16, 24, 34, 37, 38]. (We will review some of the contributions to this field in Sect. 6).

An obvious yardstick against which to measure different approaches to the formalisation of social choice frameworks is Arrow's Theorem [3], *the* seminal result in the field, which shows that it is impossible to design preference aggregation mechanisms for three or more alternatives that are Pareto efficient and for which the relative ranking of two alternatives is based only on the rankings for the same two alternatives submitted by the individual voters. For instance, recent work has modelled the Arrovian framework in propositional logic [34], first-order logic [16], higher-order logic [24, 38], and a tailor-made modal logic [1]. Some of this work has resulted in methods to prove Arrow's Theorem either automatically [34] or semi-automatically [24, 38], while other work has generated logical formalisations of the theorem that are easily accessible to humans and thus helpful in deepening our understanding of social choice [1, 16]. A shortcoming of the latter contributions, however, is that they have so far not resulted in a full proof of Arrow's Theorem or similar results *within* the chosen logical framework itself.¹ Rather, such work has proceeded by showing that a given logical system is complete w.r.t. an appropriate class of models of social choice theory, thereby proving that a rendering of Arrow's Theorem in the logical language in question must be a theorem of that logic. That is, such work has derived results about a given logic by means of reference to existing "semantic" proofs of Arrow's Theorem. The ultimate goal of such research, however, must be the opposite: to use the logic to derive proofs for Arrow's Theorem and similar results.

In this paper, we close this gap by providing a Hilbert-style syntactic proof of Arrow's Theorem within a simple tailor-made modal logic that is shown to be complete. We have opted for a Hilbert calculus, rather than, say, an approach based on natural deduction, because Hilbert calculi are still the systems used most widely by modal logicians and thus facilitate comparison to proof systems for other logics, and because this choice allows for a particularly compact presentation of our assumptions. Having said this, other proof systems have other advantages (e.g., in view of readability of proofs or implementability) and thus certainly also have a place in the study of social choice theory. Our logic of choice is a fragment of the *modal logic of social choice functions* proposed by Troquard et al. [37]. Troquard et al. have used their (full) logic to reason about the strategy-proofness of voting rules (but it has not previously been applied to Arrow's Theorem). This logic can be used to model a (resolute) *social choice function* (SCF), i.e., a function that maps any given profile of preference orders to a single winning alternative. While Arrow originally formulated his theorem for social welfare functions, i.e., functions that map any given profile of preference orders to a single social preference order [3], we will instead work with a standard variant of the theorem for

¹ However, in recent and as yet unpublished work, Perkov [30] has sketched a syntactic proof of Arrow's Theorem in a natural deduction calculus for the modal logic of Ágotnes et al. [1]. (See also Sect. 6.)

SCF's [35]. Arguably, SCF's (returning a top alternative rather than a full ranking of all alternatives) are relevant to a wider range of applications. In any case, known techniques to prove either version of the theorem are very similar [12,35]. Thus, our work also suggests how one might construct a similar syntactic proof of Arrow's Theorem for social welfare functions, using, for instance, a logic such as that of Ågotnes et al. [1].

Besides encoding and proving Arrow's Theorem, we also cover two further seminal impossibility results from social choice theory, namely Sen's Theorem [32] on the impossibility of a Paretian liberal and the Muller–Satterthwaite Theorem [23], thereby demonstrating the generality and flexibility of our approach. Both of these theorems have so far received only very little attention in the literature on logics for social choice, with the notable exception of the work of Tang and Lin [34]. Sen's Theorem shows that the Pareto principle, by which unanimously held preferences should be respected, and a very weak form of liberalism, by which there should be certain private issues that only concern a single agent and that therefore should be dictated by that agent, are incompatible. The Muller–Satterthwaite Theorem shows that the only SCF's that satisfy a particular—strong but intuitively appealing—form of monotonicity are the dictatorships and those SCF's that bar certain alternatives from winning, even if they are preferred by all agents. Arguably, these are three of the four most important classical impossibility results in social choice theory. The fourth, the Gibbard–Satterthwaite Theorem [15,31] on the impossibility of devising a strategy-proof SCF, is outside the scope of this paper as it requires us to model both declared preferences (as for the three theorems covered here) and actual preferences, so as to be able to distinguish truthful agents from agents engaging in strategic manipulation. The modal logic of SCF's we are working with can only model one type of preference. This is intended and appropriate for our purposes. However, the full original logic of Troquard et al. [37] can model these two layers of preferences—indeed, this is the main objective it had been designed for originally. Our work, together with the fact that the Gibbard–Satterthwaite Theorem may be considered a relatively simple corollary to the Muller–Satterthwaite Theorem requiring only a proof showing that strategy-proofness implies strong monotonicity [12], therefore strongly suggests that proving the Gibbard–Satterthwaite Theorem in the full logic of Troquard et al. using an extension of our approach is possible in principle.

Our proofs are presented as human-readable recipes for how to construct a fully formal derivation inside the modal logic of SCF's of the three impossibility theorems discussed. These recipes can be transformed into machine-readable proofs relatively easily, and it is therefore possible in principle to have the proofs verified automatically by a proof-checker for this logic. In this sense, our contribution narrows the gap between, on the one hand, work on logics for modelling social choice [1, 16,37] and, on the other, work on automated reasoning for social choice [9,13,24,34,38]. Having said this, there currently is no work on automated theorem proving for the modal logic we are working with, so while narrowed, aforementioned gap has not yet been fully closed. As a further step in this direction, we also discuss how to translate from modal logic into propositional logic. While this does result in a blow-up of the size of the representation of theorems (meaning that we lose readability for humans) it makes it possible for us to use standard tools, particularly SAT solvers, to automatically reason about these theorems. This perspective provides a close connection to the approach pioneered by Tang and Lin [34], and later refined by others [9,13], of automatically proving results in social choice theory using SAT solvers.

The remainder of this paper is organised as follows. Section 2 recalls the definition of a SCF, and then introduces our logic of SCF's and establishes completeness for it. This is followed up in Sect. 3, where we show how various concepts of interest for social choice theory can be modelled in this logic. This includes a discussion of the universal domain

assumption and encodings of desirable properties of SCF's, such as Pareto efficiency and monotonicity. The three theorems are encoded and then proved in Sect. 4. The translation into propositional logic, offering a means of implementation via a SAT solver, is presented in Sect. 5. Finally, Sect. 6 discusses related work in some detail and Sect. 7 concludes.

2 A modal logic of social choice functions

In this section, we recall the formal definition of a SCF and introduce the fragment of the logic put forward by Troquard et al. [37] required to define such a SCF, adapting some of their notation and terminology to our purposes. We then demonstrate that the known completeness theorem for the full logic extends to the fragment that is of interest to us here. Finally, we discuss the limitations of this logic in view of expressing properties of families of SCF's ranging over electorates of varying size, as well as how to overcome these limitations in practice.

2.1 Social choice functions

Let $N = \{1, \dots, n\}$ be a finite set of *agents* (or *individuals*) and let X be a finite set of *alternatives* (or *candidates*). To vote, each agent $i \in N$ expresses her preferences by supplying a linear order \succsim_i over X , i.e., a binary relation that is reflexive, antisymmetric, complete, and transitive.² Let $\mathcal{L}(X)$ denote the set of all such linear orders. We shall also refer to \succsim_i as the *ballot* provided by agent i , to stress the fact that this is the preference declared by the agent, but not necessarily her true preference. A *profile* is an n -tuple $(\succsim_1, \dots, \succsim_n) \in \mathcal{L}(X)^n$ of such ballots, one for each agent.

Definition 1 A resolute social choice function is a function $F : \mathcal{L}(X)^n \rightarrow X$ mapping any given profile of ballots to a single winning alternative.

Examples for resolute SCF's are well-known voting rules, such as the Borda rule or the plurality rule [35]—when combined with a suitable tie-breaking rule that ensures that there always is just a single winner. Under the Borda rule, for instance, an agent assigns as many points to a given alternative as she lists other alternatives below it (with the alternatives obtaining the most points winning). Ties may be broken, for instance, by using the ballot of the first agent.

2.2 Language

Troquard et al. [37] have introduced a modal logic, which they call $L^{\text{scf}}[N, X]$, to reason about resolute SCF's (mapping declared preferences to winners) as well as the agents' truthful preferences. This logic can be used to model strategic behaviour in voting. Here we are not specifically interested in this strategic component, but rather in the purely aggregative aspect of social choice, i.e., in the question of whether a given SCF fairly aggregates individual ballots into a social decision. For the purposes of the present paper, we shall refer to the relevant fragment of the logic of Troquard et al. as $L[N, X]$, the *logic of SCF's* parametrised by N and X . Next, we define the language, i.e., the set of well-formed formulas, of this logic.

² The strict part \succ_i of \succsim_i is a strict linear order, a relation that is irreflexive, complete, and transitive. While most work in voting theory tends to take such strict linear orders as primitive, we instead follow Troquard et al. [37] and work with non-strict linear orders. Ultimately, both approaches are equivalent: \succsim_i uniquely determines \succ_i , and vice versa.

This language is built on top of two types of atomic propositions. First, for every $i \in N$ and $x, y \in X$, $p_{x \succ_i y}^i$ is an atomic proposition (with the intuitive meaning that agent i prefers x to y). $Pref[N, X] := \{p_{x \succ_i y}^i \mid i \in N \text{ and } x, y \in X\}$ is the set of all such propositions. Second, by a slight abuse of notation, every alternative $x \in X$ is also an atomic proposition (with the intuitive meaning that x wins). Besides the usual propositional connectives, we have a modal operator \diamond_C for every coalition of agents $C \subseteq N$ (with the intuitive meaning that C can ensure the truth of a given formula, provided the others do not alter their ballots). The following definition summarises how the language is constructed.

Definition 2 The set of well-formed formulas φ in the language of $L[N, X]$ is generated by the following Backus-Naur Form (where $p \in Pref[N, X]$, $x \in X$ and $C \subseteq N$):

$$\varphi ::= p \mid x \mid \top \mid \neg\varphi \mid \varphi \vee \varphi \mid \diamond_C \varphi$$

Additional propositional connectives and a dual modal operator are defined in the usual manner: $\varphi \wedge \psi$ is short for $\neg(\neg\varphi \vee \neg\psi)$, $\varphi \rightarrow \psi$ is short for $\neg\varphi \vee \psi$, $\varphi \leftrightarrow \psi$ is short for $(\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$, \perp is short for $\neg\top$, and $\square_C \varphi$ is short for $\neg\diamond_C\neg\varphi$. For $i \in N$, we write \diamond_i as a shorthand for $\diamond_{\{i\}}$ and \square_i as a shorthand for $\square_{\{i\}}$.

The full logic of Troquard et al. [37] includes an additional pair of modal operators to speak about true preferences.

2.3 Semantics

The semantics of the logic is a standard possible-worlds semantics for modal logics, defined in terms of a set of possible worlds, a family of accessibility relations, and a valuation function [6]. We first give a short high-level description intended for readers familiar with such semantics, and then provide complete formal definitions.

First, the set of possible worlds is the set of all possible profiles—which is fully determined by N and X . The semantics of atomic propositions of the form $p_{x \succ_i y}^i$ will be defined solely in terms of this set of possible worlds: $p_{x \succ_i y}^i$ is true at a given world/profile w , if agent i prefers x to y in w . Only to model the truth of atomic propositions of the form x will we require a valuation function. Valuation functions here are SCF's: x is true at world/profile w if the SCF in question maps profile w to the winning alternative x . Finally, for every coalition $C \subseteq N$, there is an accessibility relation between worlds/profiles: w is connected to w' if they differ only w.r.t. the preferences of agents in C . These accessibility relations will be used to define the semantics of modal formulas of the form $\diamond_C \varphi$ in the usual manner.

Definition 3 A model is a triple $M = \langle N, X, F \rangle$, consisting of a finite set of agents N with $n = |N|$, a finite set of alternatives X , and a SCF $F : \mathcal{L}(X)^n \rightarrow X$.

For fixed sets N and X , we sometimes write M_F for the model $M = \langle N, X, F \rangle$ based on the SCF F . From now on we shall use the terms ‘world’ and ‘profile’ interchangeably. We are now ready to define what it means for a formula φ to be true at a world $w = (\succ_1, \dots, \succ_n)$ in a given model M .

Definition 4 Let $M = \langle N, X, F \rangle$ be a model. We write $M, w \models \varphi$ to express that the formula φ is true at the world $w = (\succ_1, \dots, \succ_n) \in \mathcal{L}(X)^n$ in M . The satisfaction relation \models is defined inductively:

- $M, w \models p_{x \succ_i y}^i$ iff $x \succ_i y$
- $M, w \models x$ iff $F(\succ_1, \dots, \succ_n) = x$

- $M, w \models \neg\varphi$ if $M, w \not\models \varphi$
- $M, w \models \varphi \vee \psi$ iff $M, w \models \varphi$ or $M, w \models \psi$
- $M, w \models \diamond_C\varphi$ iff $M, w' \models \varphi$ for some world $w' = (\succ'_1, \dots, \succ'_n) \in \mathcal{L}(X)^n$ with $\succ'_i = \succ_i$ for all agents $i \in N \setminus C$.

That is, $\diamond_C\varphi$ is true at w , if the agents in C can make φ true by changing their own ballots (assuming none of the other agents change as well). Thus, $\square_C\varphi$ is true at w if φ holds at every world that is reachable from w by only the agents in C changing their ballots.

In some sense, the truth of every formula of the form $p^i_{x \succ_y}$ is under the control of agent i . Because of this feature, this kind of logic is sometimes classified as a *logic of propositional control*. The motivation underlying such logics is essentially game-theoretic: every individual is conceived as having “control” over a set of atomic propositions. The choice of a particular truth value for these atomic propositions can be seen as an action of the individual, and therefore a valuation of all the atomic propositions of this sort corresponds to a strategy profile. For more details and motivations on logics of propositional control we refer to the work of van der Hoek and Wooldridge [19], Gerbrandy [14], Balbiani et al. [4] and Troquard et al. [37], amongst others. We also note that these logics are closely related to Pauly’s coalition logic [27], Boolean games [7, 18], and the Ceteris Paribus Logic of Grossi et al. [17].

Let φ be a formula in the language based on N and X . Then φ is called *satisfiable*, if there exist a SCF F and a world $w \in \mathcal{L}(X)^n$ such that $M_F, w \models \varphi$. It is called *true in the model* M , denoted $M \models \varphi$, if $M, w \models \varphi$ for every world $w \in \mathcal{L}(X)^n$. Finally, it is called *valid*, denoted $\models \varphi$, if $M \models \varphi$ for every model M based on N and X .

The logic of Troquard et al. [37] is known to be decidable and this result immediately extends to the fragment of their logic discussed here:

Proposition 1 *Determining whether a formula in the language of $L[N, X]$ is valid is a decidable problem.*

Proof Since N and X are fixed, we can enumerate all models and check for each of them whether our formula is true at every world in the model. □

2.4 Axiomatisation and completeness

Next, we review the axiomatisation due to Troquard et al. [37], restricted to the fragment $L[N, X]$ discussed here, and then adapt their completeness result to this fragment. The first few axioms ensure that the propositions of the form $p^i_{x \succ_y}$ really encode linear orders.

- (1) $p^i_{x \succ_x}$ (reflexivity)
- (2) $p^i_{x \succ_y} \leftrightarrow \neg p^i_{y \succ_x}$ for $x \neq y$ (antisymmetry and completeness)
- (3) $p^i_{x \succ_y} \wedge p^i_{y \succ_z} \rightarrow p^i_{x \succ_z}$ (transitivity)

Here x, y and z range over atomic propositions in X , and i ranges over agents. Before we continue with the axiomatisation, let us first introduce a couple of additional language constructs to refer to ballots and profiles within the logical language. Consider a profile $w = (\succ_1, \dots, \succ_n) \in \mathcal{L}(X)^n$. For a given agent $i \in N$, let x_1, x_2, \dots, x_m be a permutation of the elements of X such that $x_1 \succ_i x_2 \succ_i \dots \succ_i x_m$. Then $ballot_i(w)$ is defined as the following formula:

$$ballot_i(w) := p^i_{x_1 \succ x_2} \wedge p^i_{x_2 \succ x_3} \wedge \dots \wedge p^i_{x_{m-1} \succ x_m}$$

Thus, $ballot_i(w)$ is true at world w' if and only if w and w' agree as far as the ballot of agent i is concerned. Note that $ballot_i(w)$ is a purely syntactic representation of a semantic notion

(namely, agent i 's preference order \succ_i). Similarly, we define $profile(w)$ as the following formula:

$$profile(w) := ballot_1(w) \wedge ballot_2(w) \wedge \dots \wedge ballot_n(w)$$

Hence, the formula $profile(w)$ is true at world w , and only there. This shows that *nominals*, i.e., formulas uniquely identifying worlds [6], are definable within this logic. Furthermore, due to the finiteness of X and N , there can be only finitely many formulas of type $profile(w)$ that are consistent with the axioms.

Let $N_{x \succ y}^w := \{i \in N \mid x \succ_i y\}$ denote the set of agents that prefer x over y in profile $w = (\succ_1, \dots, \succ_n)$. By a slight abuse of notation, we use the same expression as a construct of our language:

$$N_{x \succ y}^w := \bigwedge \left\{ P_{x \succ y}^i \mid x \succ_i y \text{ in } w \right\}$$

We write $N_{x \succ y}^w$ to denote both the set of agents and the formula; the context will disambiguate the intended meaning. Note that $\bigwedge_{x,y \in X} N_{x \succ y}^w$ is logically equivalent to $profile(w)$: this reflects the fact that a profile can either be presented by specifying the preferences of each individual or by specifying the sets of agents preferring one alternative over another, for all pairs of alternatives.

For any two alternatives $x, y \in X$, we define $profile(w)(x, y)$ as the formula fixing the relative ordering of x and y for all agents as in profile w :

$$profile(w)(x, y) := N_{x \succ y}^w \wedge N_{y \succ x}^w$$

This formula will be used to express the fact that two profiles ‘agree’ on the preferences concerning the alternatives x and y .

We now state the remaining axioms defining the logic $L[N, X]$:

- (4) all propositional tautologies
- (5) $\Box_i(\varphi \rightarrow \psi) \rightarrow (\Box_i\varphi \rightarrow \Box_i\psi)$ (K(i))
- (6) $\Box_i\varphi \rightarrow \varphi$ (T(i))
- (7) $\varphi \rightarrow \Box_i\Diamond_i\varphi$ (B(i))
- (8) $\Diamond_i\Box_j\varphi \leftrightarrow \Box_j\Diamond_i\varphi$ (confluence)
- (9) $\Box_{C_1}\Box_{C_2}\varphi \leftrightarrow \Box_{C_1 \cup C_2}\varphi$ (union)
- (10) $\Box_{\emptyset}\varphi \leftrightarrow \varphi$ (empty coalition)
- (11) $(\Diamond_i p \wedge \Diamond_i \neg p) \rightarrow (\Box_j p \vee \Box_j \neg p)$, where $i \neq j$ (exclusiveness)
- (12) $\Diamond_i ballot_i(w)$ (ballot)
- (13) $\Diamond_{C_1}\delta_1 \wedge \Diamond_{C_2}\delta_2 \rightarrow \Diamond_{C_1 \cup C_2}(\delta_1 \wedge \delta_2)$ (cooperation)
- (14) $\bigvee_{x \in X}(x \wedge \bigwedge_{y \in X \setminus \{x\}} \neg y)$ (resoluteness)
- (15) $(profile(w) \wedge \varphi) \rightarrow \Box_N(profile(w) \rightarrow \varphi)$ (functionality)

Here φ and ψ range over arbitrary formulas, x over atomic propositions in X , i and j over agents, C_1 and C_2 over coalitions, and w over profiles. In axiom (11), p is ranging only over atomic propositions in the set $Pref[N, X]$, and in axiom (13) δ_1 and δ_2 do not contain any common atoms.

Axioms (4)–(8) describe well-known properties of normal modal logics [6]. Axiom (9) describes the capability of a coalition to enforce a certain formula in terms of the capabilities of its sub-coalitions. Axiom (10) states that the empty coalition cannot enforce any formula. Axiom (11) enforces a division among the atomic propositions of the shape $p_{x \succ y}^i$: if an atom is controlled by an agent i , then other agents cannot change its value. Axiom (12) ensures that every agent can express every possible preference. Due to axiom (13), if two formulas δ_1 and

δ_2 do not contain a common atom and two coalitions C_1 and C_2 can each enforce one of the formulas, then the joint coalition can enforce the conjunction $\delta_1 \wedge \delta_2$. Axiom (14) expresses that any outcome associated with a profile must be a single winning alternative. Thus, this axioms encodes the resoluteness of the SCF in question. Finally, axiom (15) ensures that every profile is associated with a single outcome, i.e., it encodes the fact that the SCF being modelled must be a function.

The inference rules of the logic are *modus ponens* and *necessitation* w.r.t. all modalities of the form \Box_i [6]:

- (MP) from $\varphi \rightarrow \psi$ and φ , infer ψ
- (Nec_{*i*}) from $\vdash \varphi$, infer $\vdash \Box_i \varphi$

Here we write $\vdash \varphi$ to express that a well-formed formula φ in the language parametrised by N and X is a *theorem* of the logic $L[N, X]$, in the sense that it can be derived from axioms (1)–(15), together with the above inference rules. The $\vdash \varphi$ appearing in the second rule thus indicates that the rule can only be applied to theorems. We define a set of formulas Γ to be *consistent* if we cannot derive a contradiction from it. The theorems of $L[N, X]$ coincide with the valid formulas:

Theorem 1 (Completeness) *The logic $L[N, X]$ is sound and complete w.r.t. the class of models of SCF’s.*

Proof (sketch) Since our logic is a fragment of $L^{\text{scf}}[N, X]$, the soundness result due to Troquard et al. [37] applies directly. The same is not true for completeness. However, as we shall outline next, *mutatis mutandis*, the proof of Troquard et al. [37] for the richer logic can be adapted to our fragment.

First, we show the existence of an isomorphism between the models of Definition 3 and particular Kripke models. The latter structures are tuples $\langle W, (R_C)_{C \subseteq N} \rangle$ where W is the set of profiles and $R_C \subseteq W \times W$ are relations defined as

$$w R_C w' \text{ iff } w \upharpoonright N \setminus C = w' \upharpoonright N \setminus C,$$

where $w \upharpoonright N \setminus C$ is the profile w restricted to only the individuals outside of C . Intuitively, $w R_C w'$ holds if all the agents outside of C express the same preferences in w and w' .

Second, given a consistent formula φ , we build a maximally consistent set Γ_φ containing it using the usual Lindenbaum construction.

Define $\text{Cluster}(\Gamma_\varphi)$ to be the set of maximally consistent sets that describe the same SCF:

$$\begin{aligned} \text{Cluster}(\Gamma_\varphi) := \{ & \Gamma \mid \forall w \in \mathcal{L}(X)^n, \forall x \in X : \\ & \Diamond_N(\text{profile}(w) \wedge x) \in \Gamma \text{ iff} \\ & \Diamond_N(\text{profile}(w) \wedge x) \in \Gamma_\varphi \} \end{aligned}$$

Finally, we consider the submodel of the canonical model generated by $\text{Cluster}(\Gamma_\varphi)$. Let us call this submodel M_φ . It remains to be checked that:

- the Truth Lemma holds for M_φ ,
- there is a bijection between profiles and states of M_φ ,
- M_φ is one of the aforementioned particular Kripke models corresponding to the models of our logic.

The first item is shown in the customary way, while the other items are proven by exploiting the axioms. □

2.5 Representing families of social choice functions

To complete the outline of the expressive capabilities of $L[N, X]$, we illustrate how it is possible to encode a SCF as a formula. Given a SCF F , its representation will be:

$$\rho^F = \bigwedge \{ \text{profile}(w) \rightarrow x \mid w \in \mathcal{L}(X)^n \text{ and } F(w) = x \}$$

That is, ρ^F is simply the conjunction, over all profiles w , of implications between a formula describing w and a formula identifying the winning alternative for profile w under F . In other words, we need to have the full graph of the function, that is, the full set of input-output pairs, to be able to encode F in the language. This is indeed possible, because, strictly speaking, ρ^F represents the function only for a fixed number of alternatives and a fixed number of agents. Moreover, since we are able to encode any set of input-output pairs, we can represent any SCF in the language.

Unfortunately, for the very same reason, ρ^F cannot be taken as a proper representative of a SCF, because it only tells us what the output of the function is in a very limited case: when the alternatives are exactly those in X and when the agents are exactly those in N . In practice, however, we are interested in *families* of SCF's. If, say, F is the Borda rule and X and N both have cardinality 3, then ρ^F will only express the workings of the Borda rule for three alternatives and three agents. A full representation of the Borda rule (which formally is a family of SCF's in the sense of Definition 1), however, should contain the information necessary to compute the output from *any* given profile. It should be a conjunction of all the formulas ρ^F for all possible choices of X and N . But even assuming that we had all such sets of pairs, there are countably many ρ 's of this kind, and our logical language does not contain countable conjunctions. Given that the language is not powerful enough to encode an algorithmic specification, there is no hope that our logic, or a similar logic, will do better than using ρ^F in representing SCF's. Indeed, this restriction to specific sets of alternatives and agents is a recognised limitation of most existing logic-based approaches to modelling frameworks of social choice [12].

Interestingly, however, this problem affects the representations of the properties of SCF's only partially. Since most of the properties do not directly refer to the specific number of alternatives and agents, we can formulate the properties leaving X and N as parameters. The same can be done when proving the relative dependencies between properties. This means that, to prove that property P_1 entails P_2 , we prove that, for fixed choices of X and N , there is a proof in the logic from the formula encoding P_1 to the formula encoding P_2 (both these formulas are instantiated to X and N themselves). This is the approach we shall take here.

3 Modelling features of social choice theory

In this section, we show how to model several important concepts of social choice theory in our logic. We start by proving the Universal Domain Lemma, which demonstrates that there exists a formula in our language that expresses that for every possible preference profile there exists a world where it is realised, and that is a theorem of our logic. This simple but important result will be used throughout the paper. We then pause to introduce and encode a notion that will feature in several properties and proofs in the next (sub)sections, the concept of a decisive coalition. Finally, we formalise the main properties featuring in the classical impossibility theorems we want to prove, particularly Pareto efficiency, independence of

irrelevant alternatives, strong monotonicity, and liberalism. For each property we suggest an encoding in the logic and prove that it indeed captures the corresponding semantic notion.

Throughout, we exploit freely the finiteness of the language: we will use big conjunctions and disjunctions to quantify over individuals, alternatives, and profiles.

3.1 The Universal Domain Lemma

The following lemma states that all the possible profiles are also possible worlds in the semantics. This fact, which is implicit in our definition of a SCF, is called the *universal domain* condition in Arrow’s original work [3].

Lemma 1 (Universal Domain Lemma) *For every possible profile $w \in \mathcal{L}(X)^n$, we have that $\vdash \diamond_N \text{profile}(w)$.*

Proof Take any profile w . Then $\text{ballot}_1(w)$ encodes the preferences of the first agent. We have, by axiom (I2), that $\diamond_1 \text{ballot}_1(w)$, and similarly for the second agent we get $\diamond_2 \text{ballot}_2(w)$. Because $\text{ballot}_1(w)$ and $\text{ballot}_2(w)$ contain different atoms (the former only atoms with superscript 1, the latter only atoms with superscript 2), we can apply axiom (I3) and obtain $\diamond_{\{1,2\}}(\text{ballot}_1(w) \wedge \text{ballot}_2(w))$. We can repeat this reasoning for all the finitely many agents in N to prove $\diamond_N \text{profile}(w)$. \square

3.2 Decisive coalitions

We will call a coalition of agents $C \subseteq N$ *decisive* over a pair of alternatives $(x, y) \in X^2$ if the members of C preferring x to y is a sufficient condition for preventing y from winning. We use the following formula to encode decisiveness of C over (x, y) :

$$Cdec(x, y) := \left(\bigwedge_{i \in C} p_{x \succ y}^i \right) \rightarrow \neg y$$

If C is decisive on every pair, we will simply write $Cdec$. Along the same lines, we define a *weakly decisive* coalition C for (x, y) as a coalition that can bar y from winning if *exactly* the agents in C prefer x to y . We encode weak decisiveness of C over (x, y) as follows:

$$Cwdec(x, y) := \left(\bigwedge_{i \in C} p_{x \succ y}^i \wedge \bigwedge_{i \notin C} p_{y \succ x}^i \right) \rightarrow \neg y$$

The reader can easily check that these syntactic notions match the semantic ones; for example, in the case of decisiveness we have that $Cdec(x, y)$ is true in the model M_F if and only if the coalition C is decisive over that pair of alternatives for the corresponding SCF F .

3.3 Pareto efficiency

We now introduce several properties that one might reasonably want to require a SCF to satisfy. The first is *Pareto efficiency*, expressing the desideratum that, if all the agents rank an alternative x above another alternative y , then y should not win.

Definition 5 A SCF F is Pareto efficient if, for every profile $w \in \mathcal{L}(X)^n$ and every pair of distinct alternatives $x, y \in X$ with $N_{x \succ y}^w = N$, we obtain $F(w) \neq y$.

This is formalised as follows:

$$Par := \bigwedge_{x \in X} \bigwedge_{y \in X \setminus \{x\}} \left[\left(\bigwedge_{i \in N} P_{x \succcurlyeq y}^i \right) \rightarrow \neg y \right]$$

Observe that *Par* is equivalent to *Ndec*, i.e., to saying that the grand coalition *N* is decisive on every pair.

Lemma 2 *For every SCF F, M_F ⊨ Par if and only if F is Pareto efficient.*

Proof Straightforward. □

3.4 Independence of irrelevant alternatives

Our next property of interest is *independence of irrelevant alternatives* (IIA). It expresses the intuitively desirable property of a SCF *F* that, for every two profiles and for every two alternatives *x* and *y*, if the outcome of *F* in the first profile is *x* and the two profiles are identical as far as the preferences of the agents over *x* and *y* are concerned, then the outcome of *F* in the second profile should not be *y*. The original formulation of IIA given by Arrow [3] was applied to social welfare functions rather than SCF’s. Our definition is the most natural adaptation of Arrow’s idea to SCF’s. It has also been used by Taylor [35], amongst others.

Definition 6 A SCF *F* satisfies IIA if, for every pair of profiles *w, w’* ∈ ℒ(*X*)^{*n*} and every pair of distinct alternatives *x, y* ∈ *X* with *N*^{*w*}_{*x* ≻_≧ *y*} = *N*^{*w’*}_{*x* ≻_≧ *y*}, it is the case that *F*(*w*) = *x* implies *F*(*w’*) ≠ *y*.

We formalise this property in our logic as follows:

$$IIA := \bigwedge_{w \in \mathcal{L}(X)^n} \bigwedge_{x \in X} \bigwedge_{y \in X \setminus \{x\}} [\Diamond_N(\text{profile}(w) \wedge x) \rightarrow (\text{profile}(w)(x, y) \rightarrow \neg y)]$$

That is, if in some world (reachable via the \Diamond_N -modality) we observe profile *w* with alternative *x* winning, then in the present world, if it agrees with *w* as far as the relative ranking of *x* and *y* is concerned, *y* cannot be the winner.

The following lemma formally establishes the correspondence between the syntactic formulation of IIA and its semantic definition.

Lemma 3 *For every SCF F, M_F ⊨ IIA if and only if F satisfies the property of independence of irrelevant alternatives.*

Proof From right to left, assume *F* satisfies IIA. We want to prove every conjunct of the formula *IIA*. So take any generic world *w’* such that *M_F, w’* ⊨ $\Diamond_N(\text{profile}(w) \wedge x)$. We want to show that *M_F, w’* ⊨ $(\text{profile}(w)(x, y) \rightarrow \neg y)$. So suppose *M_F, w’* ⊨ $\text{profile}(w)(x, y)$, which entails *N*^{*w*}_{*x* ≻_≧ *y*} = *N*^{*w’*}_{*x* ≻_≧ *y*}. By the semantics of \Diamond_N , there is a world *w’’* such that *M_F, w’’* ⊨ $\text{profile}(w) \wedge x$, which entails *N*^{*w*}_{*x* ≻_≧ *y*} = *N*^{*w’’*}_{*x* ≻_≧ *y*}. Thus, also *N*^{*w’*}_{*x* ≻_≧ *y*} = *N*^{*w’’*}_{*x* ≻_≧ *y*}. From *M_F, w’’* ⊨ *x* we can infer *F*(*w’’*) = *x*. Now we can apply IIA to *w’’* and *w’* and obtain *F*(*w’*) = *x* and thus *F*(*w’*) ≠ *y*. Again by the semantics, this is tantamount to *M_F, w’* ⊨ $\neg y$.

From left to right, assume *M_F* ⊨ *IIA*. Take any two profiles *w, w’* and two alternatives *x, y* with *N*^{*w*}_{*x* ≻_≧ *y*} = *N*^{*w’*}_{*x* ≻_≧ *y*}. Now assume *F*(*w*) = *x*. We thus have *M_F, w* ⊨ $\text{profile}(w) \wedge x$ and, by the semantics of \Diamond_N , also *M_F, w’* ⊨ $\Diamond_N(\text{profile}(w) \wedge x)$. Using modus ponens and formula *IIA*, we get *M_F, w’* ⊨ $(\text{profile}(w)(x, y) \rightarrow \neg y)$. But we assumed *N*^{*w*}_{*x* ≻_≧ *y*} = *N*^{*w’*}_{*x* ≻_≧ *y*}, hence *M_F, w’* ⊨ $\text{profile}(w)(x, y)$ and thus *M_F, w’* ⊨ $\neg y$, which by the semantics entails *F*(*w’*) ≠ *y*. □

3.5 Strong monotonicity

Next is a monotonicity property known as *Maskin monotonicity* or *strong monotonicity*. It requires that, whenever alternative x wins in a given profile and we (weakly) improve the standing of x *vis-à-vis* all other alternatives, then x should still win in the new profile—even if the relative rankings of other alternatives change in the profile as well. While its formal definition is similar to that of IIA, there are subtle differences: we are now quantifying over all other alternatives y rather than considering one specific such alternative.

Definition 7 A SCF F is strongly monotonic if, for every pair of profiles $w, w' \in \mathcal{L}(X)^n$ and every alternative $x \in X$, it is the case that $F(w) = x$ and $N_{x \succ y}^w \subseteq N_{x \succ y}^{w'}$ for all $y \in X \setminus \{x\}$ together imply $F(w') = x$.

This property can be encoded as follows:

$$SM := \bigwedge_{w \in \mathcal{L}(X)^n} \bigwedge_{x \in X} \left[\diamond_N(\text{profile}(w) \wedge x) \wedge \left(\bigwedge_{y \in X \setminus \{x\}} N_{x \succ y}^w \right) \rightarrow x \right]$$

Lemma 4 For every SCF F , $M_F \models SM$ if and only if F is strongly monotonic.

Proof From left to right, suppose $M_F \models SM$ is the case. Now, fix two profiles w and w' and an alternative x , assume that $F(w) = x$, and assume that $N_{x \succ y}^w \subseteq N_{x \succ y}^{w'}$ for all $y \in X \setminus \{x\}$. Due to $F(w) = x$, we have $M_F, w \models \text{profile}(w) \wedge x$ and, by the semantics of \diamond_N , also $M_F, w' \models \diamond_N(\text{profile}(w) \wedge x)$. By the second assumption, namely $N_{x \succ y}^w \subseteq N_{x \succ y}^{w'}$, we obtain that the second conjunct of SM , namely $\bigwedge_{y \in X \setminus \{x\}} N_{x \succ y}^w$, is also true at w' . From the validity of SM we can conclude $M_F, w' \models x$ and hence $F(w') = x$.

From right to left, say that F is strongly monotonic and fix w and x . Take a generic profile w' and assume $M_F, w' \models \diamond_N[x \wedge \text{profile}(w)] \wedge \bigwedge_{y \in X \setminus \{x\}} N_{x \succ y}^w$. Due to the first conjunct we know that $F(w) = x$, while in light of the second we can conclude that $N_{x \succ y}^w \subseteq N_{x \succ y}^{w'}$ for all $y \in X \setminus \{x\}$, because by the semantics all the supporters of x over y in w still support x over y in w' . By strong monotonicity we get $F(w') = x$ and $M_F, w' \models x$. Since w' was generic we can conclude that SM is a validity in M_F . □

3.6 Surjectivity

Our most basic property is *surjectivity*. It expresses the desideratum that every alternative should be the winner for at least one profile.

Definition 8 A SCF F is surjective if, for every alternative $x \in X$ there exists a profile $w \in \mathcal{L}(X)^n$ such that $F(w) = x$.

We can encode surjectivity as follows:

$$Sur := \bigwedge_{x \in X} \bigvee_{w \in \mathcal{L}(X)^n} \diamond_N(\text{profile}(w) \wedge x)$$

Lemma 5 For every SCF F , $M_F \models Sur$ if and only if F is surjective.

Proof Straightforward. □

3.7 Liberalism

The idea that a form of *liberalism* can be modelled as a property of SCF’s is due to Sen [32]. He postulated that every agent should have the power to determine the relative ranking of at least two alternatives x and y . For example, x might be the state of the world in which Barack Obama is president of the United States of America and you paint the walls of your bedroom in pink, and y might be the state of the world where Barack Obama is president of the United States of America and you paint the walls of your bedroom in white. Then you should have the power of excluding one of x and y from being the collectively chosen alternative (which of course does not mean that the other one of the two necessarily needs to be chosen). In this case, we say that you are (two-way) decisive on x and y .

Definition 9 A SCF F satisfies the property of liberalism if, for every individual $i \in N$ there exist two distinct alternatives $x, y \in X$ for which i is two-ways decisive.

The property of liberalism can be encoded as follows:

$$Lib := \bigwedge_{i \in N} \bigvee_{x \in X} \bigvee_{y \in X \setminus \{x\}} (\{i\}dec(x, y) \wedge \{i\}dec(y, x))$$

Lemma 6 For every SCF F , $M_F \models Lib$ if and only if F satisfies liberalism.

Proof From left to right, suppose $M_F \models Lib$. Suppose for the sake of contradiction that F does not satisfy liberalism. If there is an individual i that is not two-ways decisive on any pairs then for every pair there is a profile w such that the outcome $F(w)$ is in conflict with the preferences of i (say, $x \succsim_i y$ and $F(w) = y$). This means that $\{i\}dec(x, y) \wedge \{i\}dec(y, x)$ cannot be a validity in the model M_F , and the same holds for all the pairs, so $\bigvee_x \bigvee_{y \neq x} (\{i\}dec(x, y) \wedge \{i\}dec(y, x))$ cannot be a validity either, for our fixed i . This in turn entails that $M_F \models Lib$ is not the case, which constitutes the desired contradiction.

From right to left, say that F satisfies liberalism. Fixing an agent i , it is easy to check that, calling x, y the alternatives for which i is decisive, we must have $\{i\}dec(x, y) \wedge \{i\}dec(y, x)$ as a validity on the model M_F . Thus, also $\bigvee_x \bigvee_{y \neq x} (\{i\}dec(x, y) \wedge \{i\}dec(y, x))$ is a validity, and the same holds for every i , so we get the validity of Lib . □

3.8 Dictatorships

Finally, we will require one *undesirable* property of SCF’s. A *dictatorship* is a SCF for which one individual, the dictator, can enforce their top alternative as the outcome. Denote with top_i^w that alternative $x \in X$ for which $x \succsim_i y$ for all other alternatives $y \in X$ in profile $w = (\succsim_1, \dots, \succsim_n)$.

Definition 10 A SCF F is a dictatorship if there exists an agent $i \in N$ (the dictator) such that, for every profile $w \in \mathcal{L}(X)^n$, we obtain $F(w) = top_i^w$.

The property of being a dictatorship is encoded by the following formula:

$$Dic := \bigvee_{i \in N} \bigwedge_{x \in X} \bigwedge_{y \in X \setminus \{x\}} (p_{x \succsim y}^i \rightarrow \neg y)$$

Observe that Dic is equivalent to $\bigvee_{i \in N} \{i\}dec$, i.e., a SCF is dictatorial if and only if there exists an individual that is decisive on every pair.

Lemma 7 For every SCF F , $M_F \models Dic$ if and only if F is a dictatorship.

Proof From right to left, suppose F is a dictatorship, and call the dictator i . Take any world $w = (\succsim_1, \dots, \succsim_n)$. We want to show that the disjunct corresponding to i is true at w . Thus, for any two distinct alternative x, y we want to show that $p_{x \succsim_i y}^i \rightarrow \neg y$ is true at w . First, if $x \succsim_i y$, then $top_i^w \neq y$ and thus, due to F being a dictatorship of i , we have $F(w) \neq y$. By the semantics, this entails $M_F, w \models \neg y$ and thus $M_F, w \models p_{x \succsim_i y}^i \rightarrow \neg y$. Second, if $x \not\succeq_i y$, then $M_F, w \not\models p_{x \succsim_i y}^i$, and the implication holds vacuously.

From left to right, suppose $M_F \models Dic$. Then one of the disjuncts must be valid, say for agent i . Suppose $x = top_i^w$ under profile w . Then $M_F, w \models \bigwedge_{y \in X \setminus \{x\}} p_{x \succsim_i y}^i$. Since (the disjunct referring to i in) the condition Dic is true at w , we obtain $M_F, w \models \bigwedge_{y \in X \setminus \{x\}} \neg y$. By resoluteness, this entails $M_F, w \models x$ and thus $F(w) = x$. \square

Note that, in the presence of axiom (14), encoding resoluteness, the disjunction in the formula Dic is actually an exclusive one, i.e., not only must there be some dictator, but there must be exactly one dictator.³

4 Impossibility theorems

We are now ready to state the three major impossibility theorems we are interested in as formulas in the language of our modal logic of SCF’s. For each of them, we then demonstrate how to construct a full proof of the theorem within the axiomatic system we have seen to be complete for our logic (cf. Theorem 1). We start with Arrow’s Theorem and then prove the Muller–Satterthwaite Theorem as a corollary. The third theorem, Sen’s Theorem on the impossibility of a Paretian liberal, is mathematically much simpler and also admits a relatively short proof in our logic.

Before we begin, we need to make one important remark concerning the expressivity of our logic. Given that the language of $L[N, X]$ is parametrised by the set of individuals N and the set of alternatives X , strictly speaking the aforementioned theorems, which all apply to scenarios with arbitrary numbers of individuals and alternatives (provided those numbers are sufficiently large), cannot be stated or proven within the logic. To prove each of these impossibility theorems in their full generality we have to resort to a meta-argument, using a proof schema, to show that, for each choice of N and X , it is possible to prove a version of the theorem in the logic instantiated to those two parameters. The same *proviso* also holds for the properties of SCF’s featuring in the previous section: rather than being formulas in the logic, they are schemas of the representations of the properties in the logic.

4.1 Encoding Arrow’s Theorem

First published in 1951, Arrow’s Theorem is widely regarded as *the* seminal contributions to social choice theory [3]. The original theorem concerns *social welfare functions*, i.e., functions mapping profiles of (weak) preference orders (permitting indifference between alternatives) to single collective preference orders. The version we present here is adapted for preference orders that do not permit indifferences between alternatives and to SCF’s (which return a single winning alternative rather than a collective order). We refer to Taylor [35] for an extensive discussion of this variant of the theorem. From a mathematical point of view,

³ The reader can prove this using the Universal Domain Lemma, formula Dic , and axiom (14). The gist of the proof is to take a profile where two dictators disagree and to show that this leads to a contradiction.

both variants are essentially equivalent and can be proven using the same methods [12,35]. We focus on linear orders (not permitting indifferences), because most standard voting rules impose this requirement on ballots [35]. We furthermore focus on SCF's, because the problem of choosing a single best alternative is more pervasive in applications than that of choosing a full ranking over alternatives.

Arrow showed that, rather surprisingly, any SCF for three or more alternatives that is Pareto efficient and that satisfies the property of independence of irrelevant alternatives must be dictatorial.

Theorem 2 (Arrow’s Theorem) *Any SCF for at least three alternatives that satisfies IIA and the Pareto condition is a dictatorship.*

We now proceed to code a proof of Arrow’s Theorem in our logic. We will use a familiar technique, based on the concept of decisive coalitions (as defined in Sect. 3.2), to guide our search for a proof [12,33]. What is novel about our approach is that we show that this technique can be fully embedded into a formal derivation of the axiomatic system for $L[N, X]$ presented earlier. We offer an outline on the main steps of the proof, from which a complete formal derivation can be recovered.

The proof is based on two main lemmas. The first lemma shows that, under certain conditions, a coalition being weakly decisive over a specific pair of alternatives implies that the same coalition is (not only weakly) decisive over all pairs.

Lemma 8 *Consider a language parametrised by X such that $|X| \geq 3$. Then for any coalition $C \subseteq N$ and any two distinct alternatives $x, y \in X$, we have that:*

$$\vdash Par \wedge IIA \wedge Cwdec(x, y) \rightarrow Cdec$$

Proof Suppose x, y, x' and y' are distinct alternatives.⁴ To prove $Cdec$ we need to prove each of the conjuncts in the following formula:

$$\bigwedge_{x \in X} \bigwedge_{y \in X \setminus \{x\}} \left[\left(\bigwedge_{i \in C} P_{x \succ y}^i \right) \rightarrow \neg y \right]$$

Denote by C' one of the possible subsets of $N \setminus C$ preferring x' over y' . Now consider the following derivation:

- (1) $(\bigwedge_{i \in C} P_{x' \succ y'}^i) \rightarrow [(\bigwedge_{i \in C} P_{x' \succ y'}^i) \wedge \bigvee_{C' \subseteq N \setminus C} ((\bigwedge_{i \in C'} P_{x' \succ y'}^i) \wedge (\bigwedge_{i \notin C' \cup C} P_{y' \succ x'}^i))]$
 By finiteness of agents and alternatives and the theorems $P_{x' \succ y'}^i \vee P_{y' \succ x'}^i$ for all $i \in N$ we can, rearranging conjunctions and disjunctions, prove the second line of the formula; the implication follows.
- (2) $(\bigwedge_{i \in C} P_{x' \succ y'}^i) \rightarrow \bigvee_{C' \subseteq N \setminus C} [(\bigwedge_{i \in C} P_{x' \succ y'}^i) \wedge (\bigwedge_{i \in C'} P_{x' \succ y'}^i) \wedge (\bigwedge_{i \notin C' \cup C} P_{y' \succ x'}^i)]$
 by distributivity from (1)
- (3) This part of the proof contains the derivation of the following formula, for every coalition $C' \subseteq N \setminus C$:
 $Par \wedge IIA \wedge Cwdec(x, y) \rightarrow$
 $[(\bigwedge_{i \in C} P_{x' \succ y'}^i) \wedge (\bigwedge_{i \in C'} P_{x' \succ y'}^i) \wedge (\bigwedge_{i \notin C' \cup C} P_{y' \succ x'}^i) \rightarrow \neg y']$
 We will present the derivation for any such C' below.

⁴ With three alternatives the argument is analogous but simplified, since two of the alternatives coincide.

- (4) $Par \wedge IIA \wedge Cwdec(x, y) \rightarrow \bigvee_{C' \subseteq N \setminus C} [(\bigwedge_{i \in C} P_{x' \succ y'}^i) \wedge (\bigwedge_{i \in C'} P_{x' \succ y'}^i) \wedge (\bigwedge_{i \notin C' \cup C} P_{y' \succ x'}^i) \rightarrow \neg y']$
by propositional reasoning from all the instances of (3)
- (5) $Par \wedge IIA \wedge Cwdec(x, y) \rightarrow [(\bigwedge_{i \in C} P_{x' \succ y'}^i) \rightarrow \neg y']$
by propositional reasoning from (2) and (4)

We still need to show (all the finitely many instances of) step (3). We prove each of them in the following way. Consider a specific profile $w = (\succ_1, \dots, \succ_n)$ for which we can rearrange the conjuncts in the formula $profile(w)$ as follows:

$$profile(w) = \left(\bigwedge_{i \in C} P_{x \succ y}^i \right) \wedge \left(\bigwedge_{i \in N} \left(P_{x' \succ x}^i \wedge P_{y \succ y'}^i \right) \right) \wedge \left(\bigwedge_{i \in C \cup C'} P_{x' \succ y'}^i \right) \wedge \left(\bigwedge_{i \notin C} P_{y \succ x}^i \right) \wedge \left(\bigwedge_{i \notin C \cup C'} P_{y' \succ x'}^i \right) \wedge \alpha$$

Here α is the formula expressing the fact that all the other alternatives (if any) are ranked by all agents below x, y, x', y' . We are now ready to present a derivation for a specific coalition C' :

- (a) For any $z \in X \setminus \{x, y, x', y'\}$:
 $Par \wedge profile(w) \rightarrow \neg x \wedge \neg y' \wedge \neg z$
from formula Par , the second part of $profile(w)$, and α
- (b) $Cwdec(x, y) \wedge profile(w) \rightarrow \neg y$
by definition of $Cwdec(x, y)$
- (c) $Par \wedge Cwdec(x, y) \rightarrow (profile(w) \rightarrow x')$
by axiom (14), encoding resoluteness, with (a) and (b)
- (d) $\diamond_N profile(w)$
by the Universal Domain Lemma
- (e) $Par \wedge Cwdec(x, y) \rightarrow \diamond_N (profile(w) \wedge x')$
by standard modal reasoning from (c) and (d)
- (f) $Par \wedge IIA \wedge Cwdec(x, y) \rightarrow \diamond_N (profile(w) \wedge x')$
by propositional reasoning from (e)
- (g) $Par \wedge IIA \wedge Cwdec(x, y) \rightarrow [(profile(w)(x', y') \rightarrow \neg y')]$
from (f) and formula IIA w.r.t. x' and y'

But $profile(w)(x', y')$ consists of the following conjuncts:

$$\left(\bigwedge_{i \in C} P_{x' \succ y'}^i \right) \wedge \left(\bigwedge_{i \in C'} P_{x' \succ y'}^i \right) \wedge \left(\bigwedge_{i \notin C' \cup C} P_{y' \succ x'}^i \right)$$

Hence, we may infer that this latter formula entails $\neg y'$. This shows step (3) and concludes the proof. □

The next lemma establishes a syntactic counterpart of what is known as the *Contraction Lemma* in the literature [33]. It says that, under certain conditions, for any way of splitting a decisive coalition of two or more agents into two sub-coalitions, one of those sub-coalitions must also be decisive.

Lemma 9 (Contraction lemma) *Consider a language parametrised by X such that $|X| \geq 3$. Then for any coalition $C \subseteq N$ and any two coalitions C_1 and C_2 that form a partition of C , we have that:*

$$\vdash Par \wedge IIA \wedge Cdec \rightarrow (C_1dec \vee C_2dec)$$

Proof Consider C , C_1 and C_2 as in the statement of the lemma (i.e., $C = C_1 \cup C_2$ and $C_1 \cap C_2 = \emptyset$) and let x, y, z be three distinct alternatives. Now consider any profile w for which $profile(w)$ has the following form:

$$profile(w) = \left(\bigwedge_{i \notin C_2} p_{x \succ y}^i \right) \wedge \left(\bigwedge_{i \in C_1} p_{x \succ z}^i \right) \wedge \left(\bigwedge_{i \in C_1 \cup C_2} p_{y \succ z}^i \right) \\ \left(\bigwedge_{i \in C_2} p_{y \succ x}^i \right) \wedge \left(\bigwedge_{i \notin C_1} p_{z \succ x}^i \right) \wedge \left(\bigwedge_{i \notin C_1 \cup C_2} p_{z \succ y}^i \right) \wedge \alpha$$

Here α encodes the fact that all other alternatives (if any) are ranked by all agents below x, y, z . By propositional reasoning and the fact that in profile w all agents in C prefer y over z we can derive:

$$Cdec \rightarrow (profile(w) \rightarrow \neg z) \tag{1}$$

For any other alternative k different from x or y , we can derive:

$$Par \rightarrow (profile(w) \rightarrow \neg k) \tag{2}$$

This is so because α in $profile(w)$ encodes the fact that all other alternatives are ranked by all agents below x, y, z . Formulas (1) and (2), together with axiom (14), encoding resoluteness, enforce that x or y must be the outcome:

$$Par \wedge Cdec \rightarrow ((profile(w) \rightarrow x) \vee (profile(w) \rightarrow y)) \tag{3}$$

As an aside, we note that we know (again from resoluteness) that this disjunction must be exclusive. By the Universal Domain Lemma, we have that $\diamond_N profile(w)$ is a theorem, and thus, using standard modal reasoning on formula (3), we obtain:

$$Par \wedge Cdec \rightarrow (\diamond_N(profile(w) \wedge x) \vee \diamond_N(profile(w) \wedge y)) \tag{4}$$

Now propositional reasoning together with IIA, first w.r.t. the pair (x, z) and then w.r.t. the pair (y, x) , allows us to derive from formula (4) the following formula:

$$Par \wedge IIA \wedge Cdec \rightarrow ((profile(w)(x, z) \rightarrow \neg z) \vee (profile(w)(y, x) \rightarrow \neg x))$$

Recall that in $profile(w)$ the agents in C_1 are the only ones supporting x over z . Hence, $(profile(w)(x, z) \rightarrow \neg z)$ means that C_1 is weakly decisive for the pair (x, z) . Likewise, the agents in C_2 are the only ones supporting y over x ; thus $(profile(w)(y, x) \rightarrow \neg x)$ means that C_2 is weakly decisive for the pair (y, x) . In this fashion we can conclude that:

$$Par \wedge IIA \wedge Cdec \rightarrow (C_1wdec(x, z) \vee C_2wdec(y, x)) \tag{5}$$

We can now use Lemma 8 and propositional reasoning on formula (5) to derive:

$$Par \wedge IIA \wedge Cdec \rightarrow (C_1dec \vee C_2dec)$$

We have thus shown that $Par \wedge IIA \wedge Cdec \rightarrow (C_1dec \vee C_2dec)$ must be a theorem of the logic. Note that the disjunction is still exclusive. □

We can now state and prove a syntactic counterpart of Arrow’s Theorem:

Theorem 3 (Arrow’s Theorem) *Consider a logic $L[N, X]$ with a language parametrised by X such that $|X| \geq 3$. Then we have:*

$$\vdash Par \wedge IIA \rightarrow Dic$$

Proof As mentioned earlier, *Par* is equivalent to *Ndec*. Exploiting the formula *IIA*, we can apply the Contraction Lemma and prove that one of two disjoint subsets of *N* is decisive. Repeating the process finitely many times (we have finitely many agents), we can show that one of the singletons that form *N* is decisive. But this is tantamount to saying that there exist a decisive agent, i.e., a dictator, so the formula $\bigvee_{i \in N} \{i\}dec$, which is equivalent to *Dic*. Hence, the formula $Par \wedge IIA \rightarrow Dic$ can be derived as a theorem of the logic $L[N, X]$ for any set *X* with $|X| \geq 3$ as claimed. \square

Note that throughout the proof we have made implicit use of the condition $|X| \geq 3$ when assuming the availability of three distinct alternatives (in fact, in the proof of Lemma 8 we have only gone through the most interesting case, requiring at least four alternatives).

As we already mentioned, the proof provided here is not, strictly speaking, a full syntactic proof of Arrow’s Theorem *within* the logic, because the language is parametric in the set of agents *N* and the set of alternatives *X*. Nevertheless, apart from the *proviso* on the number of alternatives stated in Theorem 3, our proof is independent of the choice of *N* and *X*; that is to say, this proof can be used as a *template* to prove the appropriate instance of Arrow’s Theorem in *any* logic $L[N, X]$ for *N* and *X* such that $|X| \geq 3$.

Due to Theorem 1 establishing completeness of the logic and Lemmas 2, 3, and 7 establishing the correctness of our representation of the Arrovian conditions within the logic, Theorem 3 is equivalent to the usual, semantic, rendering of Arrow’s Theorem for SCF’s stated as Theorem 2. Thus, our purely syntactic proof constitutes an independent proof of the theorem. This shows that the logic $L[N, X]$ is a useful tool for reasoning about nontrivial concepts in social choice. In the remainder of this section we offer further support for this assertion, by proving two additional results.

4.2 Encoding the Muller–Satterthwaite Theorem

The Muller–Satterthwaite Theorem [23] establishes that, when there are at least three alternatives, the only SCF’s that are strongly monotonic—and that do not rule out some of the alternatives as potential winners to begin with (by failing surjectivity)—are the dictatorships. Like Arrow’s Theorem, this result shows that certain intuitively appealing properties of SCF’s simply cannot be realised in general. We directly give a syntactic formulation of this important result in our logic.

Theorem 4 (*Muller–Satterthwaite Theorem*) *Consider a logic $L[N, X]$ with a language parametrised by *X* such that $|X| \geq 3$. Then we have:*

$$\vdash SM \wedge Sur \rightarrow Dic$$

Proof It is enough to show that $\vdash SM \wedge Sur \rightarrow Par \wedge IIA$. Then, by the syntactic derivation of Arrow’s Theorem given earlier, we obtain *Dic*. We begin by showing that the two premises entail *IIA*. If we can show that

$$SM \rightarrow [\Diamond_N(profile(w) \wedge x) \rightarrow (profile(w)(x, y) \rightarrow \neg y)]$$

for any *w* and any distinct *x, y* then we have that the two premises prove the conjunction of all such consequents, which is *IIA*.

The general strategy is the following: first we construct a profile *w''* which ranks the alternatives *x, y* above all others and preserves the ordering of *w* encoded in $profile(w)(x, y)$; second, by *SM*, we conclude that *x* must be the outcome in this profile *w''*; third we show that, for any profile *w'* that agrees with $profile(w)(x, y)$, if the outcome at *w'* is *y* then, again

by SM , the outcome at w'' is also y ; this last passage contradicts the fact that x is the outcome at w'' , hence y cannot be the outcome at any such w' . We proceed to encode this reasoning.

First, construct a formula representing w'' :

$$profile(w'') := profile(w)(x, y) \wedge \bigwedge_{i \in N} \bigwedge_{z \neq x, y} (P_{x \succ z}^i \wedge P_{y \succ z}^i)$$

By construction, we clearly have that:

$$profile(w'') \rightarrow \bigwedge_{k \in X \setminus \{x\}} N_{x \succ k}^w$$

Together with SM , this latter formula readily entails the following implication:

$$SM \wedge \diamond_N(profile(w) \wedge x) \wedge profile(w'') \rightarrow x$$

By the Universal Domain Lemma, we know that we have $\diamond_N profile(w'')$. Thus:

$$SM \wedge \diamond_N(profile(w) \wedge x) \rightarrow \diamond_N(profile(w'') \wedge x) \tag{6}$$

This concludes the first two parts, showing that x must be the outcome for the profile w'' . We now reason by contradiction, assuming

$$SM \wedge \diamond_N(profile(w) \wedge x) \wedge profile(w)(x, y) \wedge y$$

and deriving $\diamond_N(profile(w'') \wedge y)$, in contradiction with formula (6), thereby forcing us to conclude that the following holds:

$$SM \wedge \diamond_N(profile(w) \wedge x) \wedge profile(w)(x, y) \rightarrow \neg y$$

Clearly, we can derive from the axioms that:

$$profile(w)(x, y) \wedge y \rightarrow \bigvee_{w'} [profile(w') \wedge profile(w)(x, y) \wedge y]$$

That is to say, there is a profile w' containing the preferences in $profile(w)(x, y)$ for which the outcome is y . By the Universal Domain Lemma, we can put a diamond in front of the profile formula. Hence, after some rearrangement we obtain:

$$profile(w)(x, y) \wedge y \rightarrow \bigvee_{w'} \diamond_N [profile(w') \wedge y] \tag{7}$$

Notice now that the part inside the disjunction looks like the first formula in the antecedent of SM , formulated for variable y . Upon inspection we can also check that for all such w'' , we get:

$$profile(w'') \rightarrow \bigwedge_{k \neq x} N_{x \succ k}^{w'}$$

Thus, we know that by applying SM we obtain:

$$SM \wedge \diamond_N [profile(w') \wedge y] \wedge profile(w'') \rightarrow y$$

Now we can push SM inside the disjunction in formula (7), use the Universal Domain Lemma to get $\diamond_N profile(w'')$, and apply the latter formula to conclude that each of the disjuncts entails $\diamond_N(profile(w'') \wedge y)$. But then the whole disjunction entails it and we can derive:

$$profile(w)(x, y) \wedge y \rightarrow \diamond_N (profile(w'') \wedge y)$$

This contradicts formula (6), since only one alternative can be the outcome and $x \neq y$. Hence, we have derived *IIA*.

Now for the derivation of *Par*. It is enough to show that $SM \wedge Sur$ entails each conjunct of the following form:

$$\left(\bigwedge_{i \in N} p_{x \succ_i y}^i \right) \rightarrow \neg y \tag{8}$$

From *Sur* we know that $\bigvee_w \diamond_N(profile(w) \wedge x)$. Construct the profile w'' which is the same as w but with the difference that x has been ranked over y by all agents:

$$profile(w'') := \bigwedge_{i \in N} p_{x \succ_i y}^i \wedge \bigwedge_{z, k \neq x} profile(w)(z, k) \wedge \bigwedge_{y >_i z} p_{x \succ_i z}^i \wedge \bigwedge_{z >_i y} profile(w)(x, z)$$

where $y >_i z$ in the subscript is just notation to mean that y is ranked over z by i in w and similarly for $z >_i y$. Clearly by this formula we have that if $z >_i y$ in w then in w'' their ranking is unchanged, while for $z = y$ and $y >_i z$ now x is ranked above z ; thus

$$profile(w) \rightarrow \bigwedge_{y \in X \setminus \{x\}} N_{x \succ y}^w$$

Hence by *SM* we have that $\diamond_N(profile(w) \wedge x) \wedge profile(w'') \rightarrow x$, that is, x is still the outcome in w'' . Hence, every disjunct in *Sur* entails $\diamond_N(profile(w'') \wedge x)$. Now notice that the antecedent in the formula (8), namely $(\bigwedge_{i \in N} p_{x \succ_i y}^i)$, by construction is just $profile(w'')(x, y)$. So pushing the latter into the disjunction we obtain that each disjunct entails:

$$\diamond_N(profile(w'') \wedge x) \wedge profile(w'')(x, y)$$

But this is the antecedent of *IIA*, hence each disjunct actually entails $\neg y$. Therefore, the whole disjunction entails $\neg y$, and we have proved the desired implication (8). □

4.3 Encoding Sen’s approach to rights

Sen’s Theorem [32] shows that it is impossible to satisfy both the property of Pareto efficiency and the property of liberalism. Unlike the other impossibility theorems discussed, this result does not depend on any assumptions regarding the number of alternatives. We again give directly a syntactic formulation using our logic.

Theorem 5 (Sen’s Theorem) *Consider any logic $L[N, X]$. Then we have:*

$$\vdash \neg(Par \wedge Lib)$$

Proof Our derivation will mirror the standard proof of the theorem [12, 32]. It is sufficient to show that $(Par \wedge Lib)$ entails a contradiction. To make the notation lighter we will use the following abbreviation, meaning that an agent i is two-way decisive over the pair (x, y) :

$$Lib^i(x, y) := \{i\}dec(x, y) \wedge \{i\}dec(y, x)$$

Consider only two of the conjuncts of *Lib*, say for agents i_1 and i_2 . If we can prove that these two conjuncts together with *Par* entail a contradiction then we are done. Begin by rearranging the conjunction of disjunctions in the definition of *Lib* into a disjunction of conjunctions. For two agents this will look like this:

$$\bigvee_{x_1, x_2, y_1, y_2} \left(Lib^{i_1}(x_1, y_1) \wedge Lib^{i_2}(x_2, y_2) \right) \tag{9}$$

This formula essentially says that there are two pairs of elements on which the two agents are respectively two-way decisive. If we can prove that each of the disjuncts entails a contradiction, then by the laws of disjunction we can infer that the whole formula entails a contradiction. Note that we can push *Par* inside such a conjunction. Therefore, the task at hand is to show that formulas of the following shape entail a contradiction for every choice of the four alternatives:

$$Par \wedge \left(Lib^{i_1}(x_1, y_1) \wedge Lib^{i_2}(x_2, y_2) \right)$$

We focus on the cases, i.e., the disjuncts, in which these are all distinct alternatives; the cases of two or three alternatives follow via a similar argument with some alternatives being merged into one. For each choice of $x_1, x_2, y_1, y_2 \in X$ build the profile $w_{x_1,2,y_1,2}$ with the following properties:

- Individual i_1 ranks x_1 above y_1 .
- Individual i_2 ranks x_2 above y_2 .
- All individuals rank y_1 above x_2 and also y_2 above x_1 .
- All individuals rank x_1, x_2, y_1, y_2 above all other alternatives.

These properties correspond to the following formulas:

- $P_{x_1 \succ y_1}^{i_1}$
- $P_{x_2 \succ y_2}^{i_2}$
- $\bigwedge_{i \in N} (P_{y_1 \succ x_2}^i \wedge P_{y_2 \succ x_1}^i)$
- $\bigwedge_{i \in N} (P_{x_1 \succ z}^i \wedge P_{x_2 \succ z}^i \wedge P_{y_1 \succ z}^i \wedge P_{y_2 \succ z}^i)$ for all other alternatives $z \in X$

Therefore, they will be part of a big conjunction forming $profile(w_{x_1,2,y_1,2})$. Clearly, by combining the latter two of the above formulas with the formula representing the Pareto condition, we can derive the following two formulas:

- $\left(Par \wedge \bigwedge_{i \in N} \left(P_{y_1 \succ x_2}^i \wedge P_{y_2 \succ x_1}^i \right) \right) \rightarrow (\neg x_2 \wedge \neg x_1)$
- $\left(Par \wedge \bigwedge_{i \in N} \left(P_{y_1 \succ z}^i \wedge P_{y_2 \succ z}^i \wedge P_{x_1 \succ z}^i \wedge P_{x_2 \succ z}^i \right) \right) \rightarrow \neg z$
for all other alternatives $z \in X$

Thus, we can derive:

$$(Par \wedge profile(w_{x_1,2,y_1,2})) \rightarrow (\neg x_2 \wedge \neg x_1 \wedge \neg z)$$

It is also easy to prove that the following two formulas hold:

- $Lib^{i_1}(x_1, y_1) \wedge P_{x_1 \succ y_1}^{i_1} \rightarrow \neg y_1$
- $Lib^{i_2}(x_2, y_2) \wedge P_{x_2 \succ y_2}^{i_2} \rightarrow \neg y_2$

Recall that the formulas $P_{x_1 \succ y_1}^{i_1}$ and $P_{x_2 \succ y_2}^{i_2}$ are also contained in $profile(w_{x_1,2,y_1,2})$. Hence, summing up what we have seen so far, we obtain:

$$\left[Par \wedge Lib^{i_1}(x_1, y_1) \wedge Lib^{i_2}(x_2, y_2) \wedge profile(w_{x_1,2,y_1,2}) \right] \rightarrow [\neg x_1 \wedge \neg x_2 \wedge \neg y_1 \wedge \neg y_2 \wedge \bar{Z}]$$

In the above formula, we use \bar{Z} as a shorthand for the conjunction $\bigwedge_{z \in X \setminus \{x_1, x_2, y_1, y_2\}} \neg z$. The consequent of the implication above is a negation of all the alternatives in X , a formula that is inconsistent with the first part of axiom (14), the axiom encoding resoluteness of the SCF. Hence, we obtain:

$$\left[Par \wedge Lib^{i_1}(x_1, y_1) \wedge Lib^{i_2}(x_2, y_2) \wedge profile(w_{x_1,2,y_1,2}) \right] \rightarrow \perp \tag{10}$$

Due to the Universal Domain Lemma, we know that the formula $\Diamond_N profile(w_{x_1,2,y_1,2})$ is a theorem of the logic. So if we are given $Par \wedge Lib^{i_1}(x_1, y_1) \wedge Lib^{i_2}(x_2, y_2)$, we can certainly deduce:

$$Par \wedge Lib^{i_1}(x_1, y_1) \wedge Lib^{i_2}(x_2, y_2) \wedge \Diamond_N profile(w_{x_1,2,y_1,2})$$

By this formula, formula (10), and modal reasoning we can conclude:

$$\left[Par \wedge Lib^{i_1}(x_1, y_1) \wedge Lib^{i_2}(x_2, y_2) \wedge \Diamond_N profile(w_{x_1,2,y_1,2}) \right] \rightarrow \Diamond_N \perp$$

Since $\Diamond_N \perp \rightarrow \perp$ is a theorem of normal modal logic we get:

$$\left[Par \wedge Lib^{i_1}(x_1, y_1) \wedge Lib^{i_2}(x_2, y_2) \right] \rightarrow \perp$$

Thus, we have shown that one of the disjuncts of formula (9) implies a contradiction. Repeating the same proof for every permutation of the four alternatives, we can thus prove that the whole disjunction entails a contradiction. Thus, also $(Par \wedge Lib)$ entails a contradiction and we are done. □

5 Implementing the logic

In this section we expand on the possibility of implementing the logic. As we will see, it is possible to translate the language of $L[N, X]$ into classical propositional logic, and more specifically into the propositional language used by Tang and Lin [34]. This paves the way for the application of SAT solvers to check the validity of formulas in our logic, thereby allowing for a fully automated check of the validity of the theorems formulated in this paper.

The language for modelling social choice functions used by Tang and Lin consists of two predicates: $p(i, x, y, w)$, expressing that in profile w agent i prefers x over y , and $s(x, w)$, expressing that alternative x is the winner in profile w . In full generality, these predicates belong to a multi-sorted first order logic with variables for agents, alternatives, and profiles. However, when the number of agents and alternatives is fixed, we can translate the quantified formulas into propositional formulas substituting for the variables all the finitely many constants; this is how Tang and Lin obtain a propositional language that can be fed into a SAT solver. Formulas in the resulting propositional language are also evaluated on the models of Definition 4:

- $M \models p(i, x, y, w)$ iff $x \succ_i y$ in profile w
- $M \models s(x, w)$ iff $F(w) = x$

We show here how to adapt the so-called Standard Translation [6] from modal logic into first-order logic to a translation from our modal language into the multi-sorted first-order logic with predicates $p(i, x, y, w)$ and $s(x, w)$. Once this is done, the formulas of the latter language can be turned into propositional clauses and checked following the approach of Tang and Lin [34]. Consider the following translation of the language of $L[N, X]$ into the language with predicates $p(i, x, y, w)$ and $s(x, w)$. The translation is parametric in w , a variable ranging over profiles:

$$\begin{aligned}
 t_w(p_{x \succcurlyeq y}^i) &\mapsto p(i, x, y, w) \\
 t_w(\neg\varphi) &\mapsto \neg t_w(\varphi) \\
 t_w(\varphi \wedge \psi) &\mapsto t_w(\varphi) \wedge t_w(\psi) \\
 t_w(x) &\mapsto s(x, w) \\
 t_w(\diamond_C \varphi) &\mapsto \exists w' \left(\bigwedge_{i \in N \setminus C} \bigwedge_{x \neq y \in X} [t_w(p_{x \succcurlyeq y}^i) \leftrightarrow t_{w'}(p_{x \succcurlyeq y}^i)] \wedge t_{w'}(\varphi) \right)
 \end{aligned}$$

The other propositional connectives are handled accordingly. The following lemma establishes the connection between the satisfiability problem for $L[N, X]$ and the satisfiability problem for the propositional language associated with the same parameters.

Proposition 2 *For every formula φ in the language of $L[N, X]$, profile w , and SCF F , φ is satisfiable at M_F , w if and only if $t_w(\varphi)$ is satisfiable at M_F .*

Proof The proof proceeds by induction on the complexity of φ . The base cases are immediate by the translation and the semantics; we expand only on the case of the modality.

First, suppose $\diamond_C \varphi$ is satisfiable. Then there are a SCF F and a profile w such that $M_F, w \models \diamond_C \varphi$, which in turn entails that there is another profile w' with $\succcurlyeq_i = \succcurlyeq'_i$ for all $i \in N \setminus C$ such that $M_F, w' \models \varphi$. By the induction hypothesis, $t_{w'}(\varphi)$ is satisfiable at M_F . Since $\succcurlyeq_i = \succcurlyeq'_i$ is the case, we will have that $\bigwedge_{i \in N \setminus C} \bigwedge_{x, y \in X} [t_w(p_{x \succcurlyeq y}^i) \leftrightarrow t_{w'}(p_{x \succcurlyeq y}^i)]$ is true at M_F . So we can conclude that $t_w(\varphi)$ is satisfiable at M_F when w' is the witness of the existential quantifier.

For the other direction, suppose $t_w(\diamond_C \varphi)$ is satisfiable at M_F . Then there exists a profile w' such that $\bigwedge_{i \in N \setminus C} \bigwedge_{x, y \in X} [t_w(p_{x \succcurlyeq y}^i) \leftrightarrow t_{w'}(p_{x \succcurlyeq y}^i)]$ and $t_{w'}(\varphi)$ are true. From the second formula and induction hypothesis we get that $M_F, w' \models \varphi$, while from the first we can conclude that $\succcurlyeq_i = \succcurlyeq'_i$ for all $i \in N \setminus C$. Thus, $M_F, w \models \diamond_C \varphi$. □

The reader may now wonder: why are we using modal logic at all, if we can collapse everything to propositional logic? The key here is *size*: the readability of the formulas of $L[N, X]$, and therefore its usefulness as a tool for formalisation, is lost in the translation into propositional logic.

To make this point precise, we inductively define a function s assigning a size to each formula in a modal propositional language: the size of propositional atoms is 1, and the size of any other formula is the sum of the sizes of its immediate subformulas plus 1. For example, the size of $p \wedge \neg q$ is $1 + 1 + (1 + 1) = 4$. It is easy to see that the size contributed by the propositional atoms and the boolean connectives remains constant during the translation: $s(t_w(p_{x \succcurlyeq y}^i)) = s(p(i, x, y, w)) = 1 = s(p_{x \succcurlyeq y}^i)$, and similarly for the other cases. For the modality, however, we have a significant difference. The formula $\diamond_C \varphi$ has size $s(\varphi) + 1$, while its translation $t_w(\diamond_C \varphi)$ has size

$$|\mathcal{L}(X)|^n \times [s(t_w(\varphi)) + 1 + (|N \setminus C| \times (|X|^2 - |X|) \times 4) - 1]$$

This formula comes from the definition of the translation. First of all, after X and N have been fixed, we have to transform the existential quantifier into a big disjunction over all possible profiles; this explains the multiplication with the factor $|\mathcal{L}(X)|^n$. Within the square brackets, we have to add the size of the translation of φ to the size of the formula $\bigwedge_{i \in N \setminus C} \bigwedge_{x, y \in X} [t_w(p_{x \succcurlyeq y}^i) \leftrightarrow t_{w'}(p_{x \succcurlyeq y}^i)]$, plus 1 because of the conjunction. Now let us look at the latter formula. If we take the bi-implication between atomic propositions as primitive,

the inner formula has size 3 (otherwise it would be even greater). This needs to be multiplied with the size of the complement of C and the size of $|X^2|$ minus all the pairs in the diagonal (we consider $x \neq y$). Counting the conjunction associated with each of the instances of the bi-implication and subtracting 1 for the additional conjunction that we are considering, we arrive at the formula above.

The reader can get a feel of the blow-up by considering the following example. Let us analyse the simple case in which there are three alternatives and two agents, and where C is a singleton. We take $\varphi = x$, an atomic proposition, so that $s(\varphi) = 1$ and $s(\diamond_C \varphi) = 2$. On the other hand, the size of the translation into propositional logic is $s(t_k(\diamond_C \varphi)) = (3!)^2 \times [1 + 1 \times (9 - 3) \times 4] = 36 \times 25 = 900$. Clearly, formulas of such size are unwieldy for humans; their best use is for automated reasoning.

Thus, the logic $L[N, X]$ can fulfill two roles in the study of social choice theory. First, as demonstrated in the main part of this paper it is a convenient formalism in which to cast proofs of theorems regarding the characterisation of SCF's in terms of basic properties. Second, as demonstrated in this section, it can serve as a convenient interface between social choice theory and propositional logic, with $L[N, X]$ ensuring readability and the propositional counterpart allowing for the use of standard computational tools, particularly SAT solvers, to automatically reason about the SCF's.

6 Related work

The idea of using formal methods to subject social procedures to the same kind of formal analysis routinely applied to algorithms and software systems can be traced back to, at least, the work of Parikh [25, 26]. The two main arguments motivating this kind of enterprise are obvious and well known: formal analysis will deepen our understanding of social procedures; and formal analysis can increase our confidence in the correctness of social procedures. Pauly [28] has suggested a third argument that is specific to the use of logic in social choice theory: the expressive power of a logical language required to express a choice-theoretic property (such as IIA) is a relevant criterion in judging the interestingness of a characterisation result making use of such a property. A fourth argument fueling this line of research is that it has the potential to uncover entirely new characterisation and impossibility results [9, 13, 34]—results that are of independent interest to economists [10].

Successful applications of logic and automated reasoning to social choice theory have included the automated verification of the correctness of practical algorithms for implementing voting rules [5] and the automated search for new impossibility theorems in the domain of ranking sets of objects [13]. However, most work to date has focussed on the Arrovian framework of preference aggregation and the challenges of representing Arrow's Theorem in a variety of logical frameworks [1, 16], of verifying the correctness of existing proofs for the theorem [24, 38], and of finding new such proofs [34]. Indeed, Arrow's Theorem is arguably the best yardstick against which to measure new formal methods for reasoning about problems of social choice. The work of Lange et al. [20] on the use of automated reasoning in different areas of economic theory, such as auctions and cooperative games, demonstrates that the basic concepts and techniques developed for the seemingly narrow domain of Arrovian preference aggregation can have a ripple-on effect on the use of formal methods in economics more widely.

Regarding Arrow's Theorem, starting at the top as far as the expressive power of the logical systems employed is concerned, Nipkow [24] and Wiedijk [38] have shown how to verify existing proofs for the theorem in higher-order logic proof assistants. Grandi and Endriss

[16] have shown that classical first-order logic is sufficiently expressive to model all aspects of Arrow's Theorem, with the sole exception being the requirement that the set of agents be finite (the theorem is not valid for infinite electorates; cf. the use of induction in the proof of Theorem 3). In particular, modelling IIA does not require second-order quantification. At the most extreme end of the spectrum, Tang and Lin [34] have shown that the theorem can be embedded into classical propositional logic, albeit only for a fixed set of agents and a fixed set of alternatives. This embedding itself ceases to be useful for deepening our understanding of social choice (as it involves thousands of clauses, even for the simplest case of $|N| = 2$ and $|X| = 3$). Instead, the great significance of the work of Tang and Lin derives from the fact that they have been able to provide a fully automated proof of the theorem based on this embedding. The work of Ågotnes et al. [1], like our own work, is orthogonal to these other contributions, in that they design a new tailor-made logic for social choice theory, rather than encoding those concepts into already existing logics. Note that Troquard et al. [37], the originators of the logic $\mathcal{A}^{\text{scf}}[N, X]$ we have used here, have themselves not attempted to model Arrow's Theorem.

Examples for work in this vein addressing results other than Arrow's Theorem are still rare. Tang and Lin [34] have extended their approach to proving Arrow's Theorem also to the Muller–Satterthwaite Theorem and to Sen's Theorem. Nipkow [24], besides treating Arrow's Theorem, also has verified a proof of the Gibbard–Satterthwaite Theorem using a higher-order logic proof assistant. Grandi and Endriss [16] also formalise Sen's Theorem.

To date, the approaches to modelling Arrow's Theorem in logical frameworks in Hilbert-style calculi, namely the contributions of Ågotnes et al. [1] and of Grandi and Endriss [16], have not yet yielded a complete proof of the theorem *within* that same logical framework, although Ågotnes et al. [1] do succeed in providing a syntactic proof of a relevant lemma. In as yet unpublished work, Perkov [30] has outlined a natural deduction proof of Arrow's Theorem using the language of Ågotnes et al. [1]. There currently are no results of this kind available for either the Muller–Satterthwaite Theorem or Sen's Theorem.

A recent survey on logic and social choice theory [12] has identified three critical points in existing work on logics for modelling concepts in social choice: (1) whether the approach does not require us to fix the sets of agents and alternatives upfront, (2) whether the universal domain assumption can be expressed in an elegant manner, and (3) whether the approach facilitates automation. Regarding point (1), as discussed in Sect. 2.5, our logic is indeed subject to the common limitation of requiring us to fix the cardinalities of N and X before even the notion of a well-formed formula can be defined, but we have also demonstrated that in practice this limitation can be overcome by working with schemas parametrised by N and X . Point (2) is convincingly taken care of by Lemma 1, the Universal Domain Lemma. Point (3), finally, is addressed in Sect. 5, where we show how to reduce the satisfiability problem of the logic $L[N, X]$ to the satisfiability problem for propositional logic. Of course, to directly develop automated reasoning tools for $L[N, X]$, thereby foregoing the need for translation and the associated blow-up in the problem size, is still of some interest. Evidence for the claim that also this direction is feasible and promising is given by Troquard [36], who has initiated a study of algorithms for model checking for the full logic $\mathcal{A}^{\text{scf}}[N, X]$, including a prototype implementation.

7 Conclusion

We have shown how to obtain a syntactic proofs of Arrow's Theorem [3], the Muller–Satterthwaite Theorem [23], and Sen's Theorem [32] within a simple modal logic for speaking

about basic concepts of preference aggregation. The logic in question is a fragment of a logic introduced by Troquard et al. [37], which we have shown to be complete by adapting their original completeness proof. While prior work has been successful in applying tools from logic and automated reasoning to social choice theory, this is the first human-readable formalisation of the framework of preference aggregation that allows for a direct derivation of these important impossibility theorems. Inspired by the work of Tang and Lin [34], we furthermore have suggested a pragmatic approach to implementing automated reasoning tools for the logic via a translation into propositional logic.

Because of the central role of Arrow's Theorem in particular, not only in social choice theory at large, but also more specifically in the emerging literature on logics for social choice, where it has served as a yardstick for assessing the suitability of a variety of approaches to logical modelling, we believe that our work constitutes a useful step towards the longterm aim of the field. This aim is to offer tangible computer-aided support for reasoning about methods for collective decision making, be it in the context of political decision making, economic interaction, or multiagent systems.

Our results suggest two important directions for future work. First, it certainly is possible, at least in principle, to encode most of the commonly studied desiderata for voting rules, similar to those modelled in Sect. 3, in the logic considered here. To what extent this might allow us to verify whether a given voting rule satisfies a given property or to re-prove other classical results in social choice theory, such as May's Theorem on the characterisation of the simple majority rule [22], are intriguing open questions. Second, our demonstration of the usefulness of modal logics of social choice underlines the importance of further developing the reasoning machinery for such logics, including optimised implementations, also beyond the translation-based approach advocated in Sect. 5.

Acknowledgments We thank the anonymous reviewers of both AAMAS-2015 and the *Journal of Autonomous Agents and Multiagent Systems* for their valuable comments and suggestions regarding improvements in presentation, additional pointers to the literature, and possible directions for future work.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

1. Ågotnes, T., van der Hoek, W., & Wooldridge, M. (2011). On the logic of preference and judgment aggregation. *Journal of Autonomous Agents and Multiagent Systems*, 22(1), 4–30.
2. Altman, A., & Tennenholtz, M. (2008). Axiomatic foundations for ranking systems. *Journal of Artificial Intelligence Research*, 31, 473–495.
3. Arrow, K.J. (1963). *Social choice and individual values* (2nd ed.). New York: Wiley. First edition published in 1951.
4. Balbiani, P., Herzog, A., & Troquard, N. (2013). Dynamic logic of propositional assignments: A well-behaved variant of PDL. In *Proceedings of the 28th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS-2013)*. IEEE Computer Society.
5. Beckert, B., Goré, R., Schürmann, C., Borner, T., & Wang, J. (2014). Verifying voting schemes. *Journal of Information Security and Applications*, 19(2), 115–129.
6. Blackburn, P., de Rijke, M., & Venema, Y. (2001). *Modal Logic*. New York: Cambridge University Press.
7. Bonzon, E., Lagasque-Schiek, M.C., Lang, J., & Zanuttini, B. (2006). Boolean games revisited. In *Proceedings of the 17th European Conference on Artificial Intelligence (ECAI-2006)*.
8. Brandt, F., Conitzer, V., & Endriss, U. (2013). Computational social choice. In G. Weiss (Ed.), *Multiagent systems* (pp. 213–283). London: MIT Press.

9. Brandt, F., & Geist, C. (2014). Finding strategyproof social choice functions via SAT solving. In *Proceedings of 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2014)*.
10. Chatterjee, S., & Sen, A. (2014). Automated reasoning in social choice theory: Some remarks. *Mathematics in Computer Science*, 8(1), 5–10.
11. Ciná, G., & Endriss, U. (2015). A syntactic proof of Arrow's theorem in a modal logic of social choice functions. In *Proceedings of 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2015)*.
12. Endriss, U. (2011). Logic and social choice theory. In A. Gupta & J. van Benthem (Eds.), *Logic and Philosophy Today* (Vol. 2, pp. 333–377). London: College Publications.
13. Geist, C., & Endriss, U. (2011). Automated search for impossibility theorems in social choice theory: Ranking sets of objects. *Journal of Artificial Intelligence Research*, 40, 143–174.
14. Gerbrandy, J. (2006). Logics of propositional control. In *Proceedings of 5th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2006)*.
15. Gibbard, A. (1973). Manipulation of voting schemes: A general result. *Econometrica*, 41(4), 587–601.
16. Grandi, U., & Endriss, U. (2013). First-order logic formalisation of impossibility theorems in preference aggregation. *Journal of Philosophical Logic*, 42(4), 595–618.
17. Grossi, D., & Schwarzentruber, F. (2015). The ceteris paribus structure of logics of game forms. *Journal of Artificial Intelligence Research*, 53, 91–126.
18. Harrenstein, P., van der Hoek, W., Meyer, J. J., & Witteveen, C. (2001). Boolean games. In *Proceedings of 8th Conference on Theoretical Aspects of Rationality and Knowledge (TARK-2001)*.
19. van der Hoek, W., & Wooldridge, M. (2005). On the logic of cooperation and propositional control. *Artificial Intelligence*, 164(1), 81–119.
20. Lange, C., Rowat, C., & Kerber, M. (2013). The ForMaRE Project: Formal mathematical reasoning in economics. *Intelligent Computer Mathematics* (pp. 330–334). New York: Springer.
21. Mao, A., Procaccia, A. D., & Chen, Y. (2013). Better human computation through principled voting. In *Proceedings of 27th AAAI Conference on Artificial Intelligence (AAAI-2013)*.
22. May, K. O. (1952). A set of independent necessary and sufficient conditions for simple majority decisions. *Econometrica*, 20(4), 680–684.
23. Muller, E., & Satterthwaite, M. A. (1977). The equivalence of strong positive association and strategy-proofness. *Journal of Economic Theory*, 14(2), 412–418.
24. Nipkow, T. (2009). Social choice theory in HOL: Arrow and Gibbard-Satterthwaite. *Journal of Automated Reasoning*, 43(3), 289–304.
25. Parikh, R. (1985). The logic of games and its applications. *Topics in the Theory of Computation, Vol. 24 of Annals of Discrete Mathematics*. New York: North-Holland.
26. Parikh, R. (2002). Social software. *Synthese*, 132(3), 187–211.
27. Pauly, M. (2002). A modal logic for coalitional power in games. *Journal of Logic and Computation*, 12(1), 149–166.
28. Pauly, M. (2008). On the role of language in social choice theory. *Synthese*, 163(2), 227–243.
29. Pennock, D. M., Horvitz, E., Giles, C. L. (2000). Social choice theory and recommender systems: Analysis of the axiomatic foundations of collaborative filtering. In *Proceedings of 17th National Conference on Artificial Intelligence (AAAI-2000)*.
30. Perkov, T. (2014). Natural deduction for a fragment of modal logic of social choice. *Presented at ESSLLI-2014 Workshop on Information Dynamics in Artificial Societies*.
31. Satterthwaite, M. A. (1975). Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10(2), 187–217.
32. Sen, A. K. (1970). The impossibility of a Paretian liberal. *The Journal of Political Economics*, 78(1), 152–157.
33. Sen, A. K. (1986). Social choice theory. In K. J. Arrow & M. D. Intriligator (Eds.), *Handbook of Mathematical Economics* (Vol. 3). New York: North-Holland.
34. Tang, P., & Lin, F. (2009). Computer-aided proofs of Arrow's and other impossibility theorems. *Artificial Intelligence*, 173(11), 1041–1053.
35. Taylor, A. D. (2005). *Social Choice and the Mathematics of Manipulation*. Cambridge: Cambridge University Press.
36. Troquard, N. (2011). Logics of social choice and perspectives on their software implementation. *Presented at Sagstuhl Seminar 11101 on Reasoning about Interaction: From Game Theory to Logic and Back*.
37. Troquard, N., van der Hoek, W., & Wooldridge, M. (2011). Reasoning about social choice functions. *Journal of Philosophical Logic*, 40(4), 473–498.
38. Wiedijk, F. (2007). Arrow's Impossibility Theorem. *Formalized Mathematics*, 15(4), 171–174.