Automated Justification of Collective Decisions via Constraint Solving

Arthur Boixel
Institute for Logic, Language and Computation (ILLC)
University of Amsterdam, The Netherlands
a.boixel@uva.nl

Ulle Endriss
Institute for Logic, Language and Computation (ILLC)
University of Amsterdam, The Netherlands
u.endriss@uva.nl

ABSTRACT

Given the preferences of several agents over a set of alternatives, there may be competing views on which of the alternatives would be the "best" compromise. We propose a formal model, grounded in social choice theory, for providing a justification for a given choice in the context of a given corpus of basic normative principles (so-called *axioms*) on which to base any possible step-by-step explanation for why a given target outcome has been or should be selected in a given situation. Thus, our notion of justification has both an explanatory and a normative component. We also develop an algorithm for computing such justifications that exploits the analogy between the notion of explanation and the concept of minimal unsatisfiable subset used in constraint programming. Finally, we report on an application of a proof-of-concept implementation of our approach to run an experimental study of the explanatory power of several axioms proposed in the social choice literature.

KEYWORDS

Computational Social Choice; Explanation and Justification of Decisions; Axiomatic Method; Automated Reasoning

ACM Reference Format:
Arthur Boixel and Ulle Endriss. 2020. Automated Justification of Collective Decisions via Constraint Solving. In *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), Auckland, New Zealand, May 9–13, 2020,* IFAAMAS, 9 pages.

1 INTRODUCTION

Imagine a group of people who try to take a collective decision. They each have their own subjective preferences over the set of alternatives available to them and they will need to settle on the selection of just one of these alternatives. Given such a profile of individual preferences and any particular target alternative, we may ask: *Can we justify selecting this alternative under this profile?* Such a justification should explain (in a manner that hopefully everyone can understand) how making this particular selection is a necessary logical consequence of a number of basic normative principles (that hopefully everyone can accept). In this paper we develop a model, grounded in the axiomatic method of social choice theory [1], for making this idea formally precise. We also develop an algorithm, using techniques from constraint programming [27], to automatically search for justifications.

The notion of justification we develop has both an explanatory and a normative component. Indeed, the latter is important, because—in order for a justification to be accepted by the agents (or an outside observer)—it must use arguments that refer to societal norms on which everyone can agree. In social choice theory, such norms are encoded in the form of so-called *axioms*. These are properties of voting rules, i.e., of rules for selecting an outcome given a profile of preferences, that have some normative appeal and that can be given a precise mathematical definition [1].[1] In our model, a justification thus must be grounded in a *normative basis* that contains relevant axioms. Given the individual preferences of the agents and a target outcome, presenting a normative basis containing axioms for which no voting rule can be found that would elect a different outcome then constitutes a justification for why the target outcome should be chosen. This justification can be refined by also providing a step-by-step *explanation* for how the selection of the target outcome follows from concrete instances of the axioms in the normative basis. Thus:

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

Let us note in passing that this view is in line with Langley's take on *Explainable AI*, who writes that "[a]n intelligent system exhibits *justified agency* if it follows society's norms and explains its activities in those terms" [17].

But in what sort of situation would we want to compute a justification for a collective decision? Consider the following examples.

Example 1. A professional society has instructed a small committee to oversee the election of its new president. The statutes of the society prescribe the use of voting rule F for this purpose. However—as the members of our committee are only too keenly aware—most delegates were not involved in the decision to adopt this particular rule and some indeed view the use of a voting rule the mechanics of which they do not fully comprehend with active suspicion. For the sake of accountability and transparency, our committee decides to publish all ballots received (in anonymised form), so that anyone who wants to can verify that the published election outcome really is the correct one. But they would like to do more. So they decide to also make available a justification of the outcome in terms of fundamental normative principles they hope everyone will agree with. These principles could be the axioms characterising F, but they could also include other axioms—possibly more convincing ones—that happen to be able to explain this specific outcome for this specific collection of ballots as well. △

Example 2. A group of colleagues are working together to formulate a new business strategy for the company that employs them.

Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), B. An, N. Yorke-Smith, A. El Fallah Seghrouchni, G. Sukthankar (eds.), May 9–13, 2020, Auckland, New Zealand.

[1] Examples include the famous principles attributed to Pareto (about not choosing a dominated alternative) and Condorcet (about respecting the wishes of the majority when they are unambiguous), but also more basic principles such as the anonymity axiom, postulating equal treatment of all voters.

This process involves taking a series of decisions on which they may initially hold diverging views. They consider voting too crude a method to resolve such disagreements and instead hope to arrive at unanimous decisions after a period of deliberation. During their discussion, they every now and then conduct a straw poll to get a clearer idea of what the most promising proposals are at that point in time. After each poll, they check which proposals could possibly be justified from the preferences elicited using appealing normative principles. Initially, they will typically find that more than one proposal could be justified *somehow*, due to the number of principles they are considering when building justifications. They then take these findings—including the appeal of any specific set of axioms required to justify a specific proposal—into account as they proceed with their deliberations. △

Example 3. The faculty members of a newly established university research centre need to agree on a method for taking important policy decisions in the years ahead. They have been somewhat disheartened by their colleague, a well-known social choice theorist, who was unable to recommend a specific voting rule and instead got side-tracked into a lecture about the multitude of impossibility results marring the field. As no voting rule has all the properties our faculty members care about, they eventually agree on a method of decision making "from first principles": they rank the axioms they can think of in terms of their normative appeal and then, whenever a decision needs to be taken, they check for each possible outcome which subset of axioms (i.e., which normative basis) might justify that outcome and then choose the outcome justified by the "best" such set. In this context, set A is better than set B if the worst axiom in A is better than the worst axiom in B (and if they are the same, we look at the second-worst axiom, and so forth). △

Thus, we may use justifications to defend a decision already taken (Example 1), to support a group intent on arriving at a good decision through deliberation (Example 2), or to serve as a stand-in when no specific voting rule seems acceptable (Example 3).

Finding a justification—an explanation grounded in a suitable normative basis—is a computationally demanding task. To make it practically feasible, we exploit an analogy between the notions of (*i*) explanation for an observed inconsistency and (*ii*) the notion of *minimal unsatisfiable subset* of constraints familiar from the field of constraint programming [20]. This allows us to build on the enormous progress in solving intricate combinatorial optimisation problems using constraint programming made by the AI and OR research communities in recent years [22].

Related work. Even though the design of algorithms for generating justifications for election outcomes presents itself as a natural challenge for the field of computational social choice [6], to date there has been precious little research on this topic. The most closely related work to ours is that of Cailloux and Endriss [7], who outline a research agenda for using tools from AI to enable computer-supported deliberation between users regarding the ins and outs of different voting rules. In this context, they also provide an algorithm for justifying an election outcome in the specific case where that outcome coincides with the outcome under the Borda rule—using a normative basis that includes a corpus of axioms characterising the Borda rule [29]. In contrast, our approach is general and—in principle—can be applied to *any* corpus of axioms.

Very recently, Procaccia [26] also argued in favour of using axioms to explain election outcomes for concrete profiles of preferences encountered by people—as opposed to the traditional use of axioms as a means for motivating the design of a voting rule applicable to every conceivable profile. In the context of multi-criteria decision making (which is closely related to voting), Belahcene et al. [3] seek to justify decisions by showing that (and how) they can be derived in a noncompensatory sorting model by instantiating the parameters of that model. While this notion of justification is different from the one Cailloux and Endriss [7] and we adopt, there are certain parallels in terms of methodology, as Belahcene et al. also employ tools from combinatorial optimisation, namely SAT solvers, to operationalise their approach. In other related work, Kirsten and Cailloux [16] develop a method, grounded in bounded model checking, for automatically generating a counterexample for the claim that a given voting rule satisfies a given axiom.

As we use constraint programming to operationalise our definition of the justification of an election outcome, in methodological terms our work may be seen as part of the recent trend of using tools from automated reasoning (as long studied in AI) in computational social choice. This includes in particular a strand of work—recently reviewed by Geist and Peters [11]—on using SAT solvers to automatically prove (so-called "base cases" of) impossibility theorems in social choice theory. Somewhat further afield, it also includes work on using bounded model checking to semi-automatically verify the formal correctness of concrete implementations of various voting rules [2]. While constraint programming has been mentioned as an alternative to SAT solving to prove impossibility theorems by Tang and Lin [28], it otherwise has remained largely unexplored in computational social choice to date.

Contribution. Our contribution is threefold. First, we develop a definition of the notion of *justification* of a collective decision in terms of (*i*) a normative basis of socially adequate axioms and (*ii*) an explanation, consisting of a minimal set of instances of those axioms that force the collective decision in question to be adopted. Second, we operationalise our definition as an *algorithm* by showing how to encode the problem of generating a justification as a constraint network and by proving that any such problem can be mapped to the well-understood problem of computing a minimal unsatisfiable subset of a constraint network. We also report on a *proof-of-concept implementation* of our algorithm. Third, we put our implementation to practical use and conduct an experimental study of the *explanatory power* of several of the standard axioms proposed in the social choice literature.

Paper outline. We present our model of justifications in Section 2 and our approach to automatically search for such justifications in Section 3. We then report on our study of the explanatory power of axioms in Section 4 and conclude with a brief discussion of directions for future work in Section 5.

2 JUSTIFYING COLLECTIVE DECISIONS

In this section we develop our proposal for a notion of *justification* for a collective decision taken on the basis of the declared preferences of several agents. We start by recalling a number of relevant concepts from the theory of voting [31].

2.1 Voting Theory

Let X, with $|X| = m$, be a finite set of *alternatives* and let $\mathcal{L}(X)$ denote the set of all strict linear orders on X. We use elements $\succ$ of $\mathcal{L}(X)$ to model the *preferences* of individual *agents*. Let N^*, with $|N^*| = n$, be a finite set of agents (the *universe*). A *profile* $\succ_N$ for an *electorate* $N \subseteq N^*$ declaring preferences is a function associating every agent $i \in N$ with a preference $\succ_i$ in $\mathcal{L}(X)$. By a slight abuse of notation, we use $\mathcal{L}(X)^+$ to denote the set $\bigcup_{N \in 2^{N^*} \setminus \{\emptyset\}} \mathcal{L}(X)^N$ of all possible profiles—for all nonempty electorates.

A *voting rule* is a function $F : \mathcal{L}(X)^+ \to 2^X \setminus \{\emptyset\}$, mapping any given profile to a nonempty subset of X, the *election winners* for that profile. Note that this definition accounts for the fact that most voting rules used in practice are *irresolute*, meaning that they will sometimes return a set of alternatives tied for winning the election. Examples for well-known voting rules include the plurality rule, the Borda rule, the Kemeny rule, and others [5].

Social choice theorists have put forward numerous normative principles, so-called *axioms*, that—ideally—a good voting rule should satisfy. Some of them, such as *strategyproofness*, are known to be particularly hard to satisfy [31], but others are less demanding. In this paper we will work with seven of them to illustrate our approach. They all apply to all nonempty electorates $N, N' \subseteq N^*$:

- *Anonymity* (ANO). If profile $\succ'_N$ can be obtained from $\succ_N$ by permuting the set of agents, then $F(\succ_N) = F(\succ'_N)$.
- *Neutrality* (NEU). If profile $\succ'_N$ can be obtained from profile $\succ_N$ by permuting the occurrences of the alternatives within all of $\succ_N$'s preferences, then $F(\succ'_N)$ can be obtained from $F(\succ_N)$ by performing the same kind of permutation.
- *Pareto Principle* (PAR). If all agents voting agree that $x \succ y$, then y should not win: $y \notin F(\succ_N)$ if $\{i \mid x \succ_i y\} = N$.
- *Condorcet Principle* (CON). If, for all $y \neq x^*$, a majority ranks x^* above y, then only x^* should win: $F(\succ_N) = \{x^*\}$ if we have $|\{i \mid x^* \succ_i y\}| > |N| / 2$ for all $y \in X \setminus \{x^*\}$.
- *Reinforcement* (REI). If the intersection of the winning sets for two disjoint electorates is nonempty, then that intersection should win when they all vote: $F(\succ_{N \cup N'}) = F(\succ_N) \cap F(\succ_{N'})$ if $N \cap N' = \emptyset$ and $F(\succ_N) \cap F(\succ_{N'}) \neq \emptyset$.[2]
- *Cancellation* (CAN). In case of a perfect tie on all pairwise comparisons, *all* alternatives should win: $F(\succ_N) = X$ in case $|\{i \mid x \succ_i y\}| = |\{i \mid y \succ_i x\}|$ for all $x, y \in X$.
- *Faithfulness* (FAI). If only a single agent votes, then her top alternative should be the unique winner.

We can express axioms such as these in a suitable formal language. For example, Grandi and Endriss [12] have shown how to do so in first-order logic, and later on we will do the same using a constraint modelling language. Whatever the language chosen, the *interpretation* of an axiom A, which we denote by $\mathbb{I}(A)$, is always a subset of the set of all voting rules:

$$\mathbb{I}(A) \quad \subseteq \quad \mathcal{L}(X)^+ \to 2^X \setminus \{\emptyset\}$$

Thus, $F \in \mathbb{I}(A)$ is the case if and only if the voting rule F is a rule that satisfies the axiom A. Similarly, for any given set of axioms $\mathcal{A}$, we use $\mathbb{I}(\mathcal{A})$ to denote the set of voting rules that satisfy all the axioms in $\mathcal{A}$, i.e., $\mathbb{I}(\mathcal{A}) = \bigcap_{A \in \mathcal{A}} \mathbb{I}(A)$.

Importantly, while our axioms encode reasonable requirements when considered in isolation, it is well known that no single voting rule satisfies all of them. In particular, the Condorcet Principle and (classical) reinforcement are in direct conflict with each other [30].

2.2 Justifications

Suppose we want to justify an election outcome X^* for a given profile $\succ_{N^*}$ (in which everyone voted) to a particular audience. Rather than referring to any specific voting rule F, we would like to explain why all (and only) the alternatives in X^* should win by referring only to normative principles (axioms) this audience considers adequate for this purpose.

Example 4. Consider the following profile:

$$\begin{array}{ll} \textit{Agent 1:} & a \succ_1 b \succ_1 c \\ \textit{Agent 2:} & b \succ_2 a \succ_2 c \end{array}$$

We can justify the claim that $\{a, b\}$ should win by first invoking the *Pareto Principle* to exclude c as a viable winner (because all agents rank a above c) and then use *anonymity* and *neutrality* to argue that either both or none of a and b must win. The claim then follows from the fact that the set of election winners cannot be empty. △

Note that in this example we have not really used the full power of the Pareto Principle, which applies to *all* alternatives $x, y \in X$ and profiles $\succ_N$ with $\{i \mid x \succ_i y\} = N \subseteq N^*$. Rather, we have only invoked a specific *instance* of the axiom, for the specific alternatives a and c and one specific profile. While it is intuitively clear what "being an instance of" means in the context of axioms, a formal definition of this concept would only be possible relative to a concrete formal language for encoding axioms. As we want to develop an approach that is general and can be applied to any such language, we side-step the issue of providing a definition and instead simply stipulate three requirements:

(i) Every instance A' of an axiom A must itself be an axiom.

(ii) The interpretation of an axiom is always equal to the intersection of the interpretations of all its instances. Thus, if A' is an instance of A, then we must have $\mathbb{I}(A') \supseteq \mathbb{I}(A)$.

(iii) The number of instances of any given axiom A is finite.

We write $A' \triangleleft A$ to denote the fact that axiom A' is an instance of axiom A. Similarly, we write $\mathcal{A}' \triangleleft \mathcal{A}$ for two sets of axioms if every $A' \in \mathcal{A}'$ is an instance of some $A \in \mathcal{A}$.

We are now ready to state the central definition of this paper:

Definition 1 (Justification). *Let $\mathbb{A}$ be a corpus of axioms for voting rules $F : \mathcal{L}(X)^+ \to 2^X \setminus \{\emptyset\}$, let $\succ_{N^*}$ be a profile, and let $X^* \subseteq X$ be a set of alternatives. Then we say that a pair $\langle \mathcal{A}^N, \mathcal{A}^E \rangle$ of sets of axioms is a **justification** (with the **normative basis** $\mathcal{A}^N$ and the **explanation** $\mathcal{A}^E$) for the set X^* winning the election under profile $\succ_{N^*}$ if and only if the following conditions are satisfied:*

(i) ***Explanatoriness.*** *$\mathcal{A}^E$ (but none of its proper subsets) can explain the desired outcome: $F(\succ_{N^*}) = X^*$ for every voting rule $F \in \mathbb{I}(\mathcal{A}^E)$, but for every set $\mathcal{A} \subset \mathcal{A}^E$ we have $F(\succ_{N^*}) \neq X^*$ for some voting rule $F \in \mathbb{I}(\mathcal{A})$.*

(ii) ***Relevance.*** *The explanation $\mathcal{A}^E$ is an instance of the normative basis $\mathcal{A}^N$: $\mathcal{A}^E \triangleleft \mathcal{A}^N$.*

[2] We note that this variant of the reinforcement axiom, which applies only to electorates $N, N' \subseteq N^*$, is subtly weaker than the classical formulation due to Young [29]. We are going to comment on this issue in more detail in Section 4.3.

(iii) ***Adequacy.*** *All axioms in the normative basis $\mathcal{A}^{\mathrm{N}}$ belong to the corpus $\mathbb{A}$ of axioms provided: $\mathcal{A}^{\mathrm{N}} \subseteq \mathbb{A}$.*

(iv) ***Nontriviality.*** *There exists at least one voting rule that satisfies all axioms in the normative basis: $\mathbb{I}(\mathcal{A}^{\mathrm{N}}) \neq \emptyset$.*

We may think of the axioms in $\mathbb{A}$, including in particular those in $\mathcal{A}^{\mathrm{N}}$, as reflecting the norms that (we might expect that) our audience subscribes to. The axioms in $\mathcal{A}^{\mathrm{E}}$ are instantiations of those general normative principles to specific profiles, agents, and alternatives that can be used to explain how those principles justify the desired election outcome.

We call a justification $\langle \mathcal{A}^{\mathrm{N}}, \mathcal{A}^{\mathrm{E}} \rangle$ *minimal* if its normative basis cannot be reduced without violating the relevance condition, i.e., if $\mathcal{A}^{\mathrm{E}} \triangleleft \mathcal{A}$ does not hold for any $\mathcal{A} \subset \mathcal{A}^{\mathrm{N}}$.

Observe that, for a given profile and outcome, we may find two different justifications, grounded in different normative bases. Thus, our approach adequately reflects the natural phenomenon that there sometimes will be more than one way to justify a decision. This is an important feature in the context of application scenarios such as the one described in Example 1: our committee may want to select the (subjectively) most convincing justification from amongst several contenders, or they may even choose to publish two or more alternative justifications for the same outcome, so as to have a better chance of convincing everyone in their audience.

Now, for a given profile, we may also find one justification for outcome X^* and another justification for Y^*. Often this is entirely unproblematic. Indeed, this is one of the strengths of our approach: different axioms can justify different outcomes for the same profile, just as different voting rules sometimes elect different winners. This is a feature we require for the application sketched in Example 2 regarding the group of colleagues using the ability to generate justifications for alternative proposals to support their deliberations aimed at identifying a high-quality decision. However, we would not want to be able to justify contradictory outcomes from *the same* normative basis. The following result shows that this indeed cannot happen, essentially thanks to the condition of nontriviality.

Theorem 1 (Coherent Justifications). *It is impossible to justify two different election outcomes for the same profile using justifications that are grounded in the same normative basis.*

Proof. Let $X^*, Y^* \subseteq X$ be sets of alternatives, $\succ_{N^*} \in \mathcal{L}(X)^+$ a profile, and $\mathcal{A}^{\mathrm{N}}$, $\mathcal{A}^{\mathrm{E}}_1$, and $\mathcal{A}^{\mathrm{E}}_2$ sets of axioms such that $\langle \mathcal{A}^{\mathrm{N}}, \mathcal{A}^{\mathrm{E}}_1 \rangle$ justifies X^* being selected under $\succ_{N^*}$ and $\langle \mathcal{A}^{\mathrm{N}}, \mathcal{A}^{\mathrm{E}}_2 \rangle$ justifies Y^* being selected under $\succ_{N^*}$. We need to show that assuming $X^* \neq Y^*$ leads to a contradiction.

From the requirement that both justifications must satisfy the condition of explanatoriness, we can infer:

$$\begin{aligned} \mathbb{I}(\mathcal{A}^{\mathrm{E}}_1) &\subseteq \{F : \mathcal{L}(X)^+ \to 2^X \setminus \{\emptyset\} \mid F(\succ_{N^*}) = X^*\} \\ \mathbb{I}(\mathcal{A}^{\mathrm{E}}_2) &\subseteq \{F : \mathcal{L}(X)^+ \to 2^X \setminus \{\emptyset\} \mid F(\succ_{N^*}) = Y^*\} \end{aligned}$$

Recall that $A' \triangleleft A$ implies $\mathbb{I}(A') \supseteq \mathbb{I}(A)$. Due to the relevance condition for $\langle \mathcal{A}^{\mathrm{N}}, \mathcal{A}^{\mathrm{E}}_1 \rangle$, we have that $\mathcal{A}^{\mathrm{E}}_1 \triangleleft \mathcal{A}^{\mathrm{N}}$. Hence, for every $A' \in \mathcal{A}^{\mathrm{E}}_1$ there exists an $A \in \mathcal{A}^{\mathrm{N}}$ with $A' \triangleleft A$ and thus $\mathbb{I}(A') \supseteq \mathbb{I}(A)$. It follows that $\mathbb{I}(\mathcal{A}^{\mathrm{N}}) \subseteq \mathbb{I}(\mathcal{A}^{\mathrm{E}}_1)$. We can use analogous reasoning for the pair $\langle \mathcal{A}^{\mathrm{E}}_2, \mathcal{A}^{\mathrm{N}} \rangle$, and thus obtain the following inclusion:

$$\mathbb{I}(\mathcal{A}^{\mathrm{N}}) \subseteq \mathbb{I}(\mathcal{A}^{\mathrm{E}}_1) \cap \mathbb{I}(\mathcal{A}^{\mathrm{E}}_2)$$

Putting everything together, we obtain:

$$\begin{aligned} \mathbb{I}(\mathcal{A}^{\mathrm{N}}) \subseteq\ & \{F : \mathcal{L}(X)^+ \to 2^X \setminus \{\emptyset\} \mid F(\succ_{N^*}) = X^*\} \cap \\ & \{F : \mathcal{L}(X)^+ \to 2^X \setminus \{\emptyset\} \mid F(\succ_{N^*}) = Y^*\} \end{aligned}$$

The intersection on the right is empty, unless $X^* = Y^*$. But the set on the left-hand side being empty would be in direct contradiction with the condition of nontriviality, so we are done. □

It is easy to adapt this proof to obtain a stronger variant of Theorem 1 that states that it is impossible to justify two different election outcomes for the same profile grounded in two normative bases $\mathcal{A}^{\mathrm{N}}_1$ and $\mathcal{A}^{\mathrm{N}}_2$ with $\mathcal{A}^{\mathrm{N}}_1 \subseteq \mathcal{A}^{\mathrm{N}}_2$.

Observe how Theorem 1 enables the kind of application envisaged in Example 3. If we have a manner of strictly ranking all possible normative bases (say, in view of the level of convincingness of its axioms), then for a given profile we can go through all normative bases, from best to worst, and for each basis try whether there is some outcome it can justify. By Theorem 1, we will never encounter a situation in which the best normative basis allowing us to justify *some* outcome will recommend *two* competing outcomes. Thus, provided we have at least one normative basis that fully characterises a voting rule (and thus will be able to generate a justification for some outcome for every possible profile), we can use this approach *in lieu* of a voting rule.

3 AUTOMATED SEARCH FOR JUSTIFICATIONS

Now that we have defined a notion of justification in abstract terms, we want to search for and find such justifications in practice. To this end, in this section we show how to translate the problem of finding a justification into a constraint satisfaction problem.[3]

3.1 Constraint Networks

Constraint programming is a collection of techniques for solving large combinatorial problems [27]. To be able to apply these techniques, we need to specify our problem in the form of a *constraint network* [4], which is a triple $\mathcal{N} = \langle \mathcal{V}, \mathcal{D}, \mathcal{C} \rangle$ such that:

- $\mathcal{V} = (V_1, \ldots, V_\ell)$ is a sequence of *variables*.
- $\mathcal{D} = D_1 \times \cdots \times D_\ell$ is an associated finite combinatorial *domain* (i.e., V_i takes values from the finite set D_i).
- $\mathcal{C}$ is a finite set of *constraints*.

Using a suitable formal language, each $C \in \mathcal{C}$ refers to some of the variables in $\mathcal{V}$ and specifies which combinations of values for these variables are allowed. Whatever the language used, the *interpretation* of a constraint C, which we denote by $\mathbb{I}(C)$, is always a subset of the domain: $\mathbb{I}(C) \subseteq \mathcal{D}$.

Example 5. If $\mathcal{V} = (x, y, z)$ and $\mathcal{D} = \{0, 1\}^3$, then the constraint "$x \neq y$", involving only two of the three variables, is interpreted as $\mathbb{I}(x \neq y) = \{(0, 1, \_), (1, 0, \_)\} = \{(0, 1, 0), (0, 1, 1), (1, 0, 0), (1, 0, 1)\}$. △

[3] In principle, everything we do in this paper using constraint programming can also be achieved using SAT solving techniques, or—after a slight modification of the model—other techniques used to solve complex combinatorial problems, such as integer programming. But constraint programming has some practical advantages, as it encodes information more compactly and in a more readable form than SAT.

Assigning each variable V_i to a value $v_i \in D_i$ gives rise to a tuple $(v_1, \ldots, v_\ell)$. Such an assignment *satisfies* a constraint C if it is an element of $\mathbb{I}(C)$; otherwise it *violates* C. A *solution* to a constraint satisfaction problem expressed in the form of a network $\mathcal{N} = \langle \mathcal{V}, \mathcal{D}, C \rangle$ then is an assignment that satisfies all of the constraints in C. $\mathcal{N}$ is called *unsatisfiable* if it does not have any solutions.

Suppose $\mathcal{N} = \langle \mathcal{V}, \mathcal{D}, C \rangle$ is an unsatisfiable constraint network. Then a set $C^* \subseteq C$ is called a *minimal unsatisfiable subset* (MUS) of (the set of constraints of) $\mathcal{N}$ if and only if (*i*) the network $\mathcal{N}^* = \langle \mathcal{V}, \mathcal{D}, C^* \rangle$ is unsatisfiable, yet (*ii*) for every subset $C' \subset C^*$ the corresponding network $\mathcal{N}' = \langle \mathcal{V}, \mathcal{D}, C' \rangle$ is satisfiable.

3.2 Encoding Justification Problems

Now suppose we want to justify the outcome $X^* \subseteq X$ for a given profile $\succ_{N^*}$ using axioms from a corpus $\mathbb{A}$. Let us call this a *justification problem* $\langle \succ_{N^*}, X^*, \mathbb{A} \rangle$. We approach this task by defining, in Definition 2, a constraint network $\mathcal{N} = \langle \mathcal{V}, \mathcal{D}, C \rangle$, with variables corresponding to profiles, each variable ranging over the domain of all possible outcomes, and constraints encoding axioms.

How many variables do we need? Recall that $n = |N^*|$ and $m = |X|$. So the number of profiles involving only (some of the) agents in N^* (and all alternatives in X) is $\ell_{n,m} = \sum_{k=1}^{n} \binom{n}{k} m!^k$. Let p be a function that assigns to each profile a unique number between 1 and $\ell_{n,m}$. Then fix $\mathcal{V} = (V_1, \ldots, V_{\ell_{n,m}})$ and $\mathcal{D} = D_1 \times \cdots \times D_{\ell_{n,m}}$ with $D_i = 2^X \setminus \{\emptyset\}$ for every profile index $i \leq \ell_{n,m}$. Variable V_i taking value X' is intended to represent the fact that exactly the alternatives in X' win the election under the unique profile $\succ_{N'}$ with $i = p(\succ_{N'})$. The set C of constraints includes one constraint for every instance of every axiom in $\mathbb{A}$.

Example 6. Let $X = \{a, b, c\}$. There are $6 + 6 + 36 = 48$ possible profiles involving agents from $N^* = \{1, 2\}$:

$\frac{1}{a}$	$\frac{1}{a}$		$\frac{1}{c}$	$\frac{2}{a}$	$\frac{2}{a}$		$\frac{2}{c}$	$\frac{1\,2}{a\,a}$	$\frac{1\,2}{a\,a}$		$\frac{1\,2}{c\,c}$
b	c	$\cdots$	b	b	c	$\cdots$	b	$b\,b$	$b\,c$	$\cdots$	$b\,b$
c	b		a	c	b		a	$c\,c$	$c\,b$		$a\,a$
1	2	$\cdots$	6	7	8	$\cdots$	12	13	14	$\cdots$	48

Thus, in profile 1 only agent 1 is participating in the election and she reports $a \succ_1 b \succ_1 c$, and so forth. One of the many instances of the *reinforcement axiom* is this constraint:

$$(V_1 \cap V_8 \neq \emptyset) \rightarrow (V_{14} = V_1 \cap V_8)$$

It expresses that, if the intersection of the outcomes chosen for profiles 1 and 8 is not empty, then in profile 14 (which under the above enumeration is the concatenation of profiles 1 and 8), we must elect that very intersection. Thus, for instance, if $V_1 = \{a, b\}$ and $V_8 = \{a, c\}$, then $V_{14} = \{a\}$. For comparison, one of the instances of the *faithfulness axiom* is the very simple constraint $V_{12} = \{c\}$. △

We stress that the entire construction presented here can be carried out using any constraint modelling language $\mathbb{L}$ that is able to encode the axioms in $\mathbb{A}$ and that comes with a well-defined notion of instance. Table 1 provides an example for how to encode axioms in the widely-used constraint modelling language MiniZinc [25].

Suppose we have fixed a specific language $\mathbb{L}$ for encoding axioms (such as MiniZinc). Then, for any set $\mathcal{A}$ of axioms, we denote with $C_{\mathcal{A}}$ the set of constraints corresponding to the instances of the axioms in $\mathcal{A}$. Thus, the set C will include all of $C_{\mathbb{A}}$. The set C also includes one further constraint, the *goal constraint*, which expresses that the outcome should *not* be equal to X^*:

$$C_{Goal} \ : \ V_i \neq X^* \ \text{ for } \ i = p(\succ_{N^*})$$

This is useful, because whenever adding C_{Goal} to a satisfiable constraint network makes that network unsatisfiable, then that means that the constraints in that network force X^* to win under profile $\succ_{N^*}$. Let us summarise the encoding we have just described:

Definition 2 (Encoding Justification Problems). *For any corpus $\mathbb{A}$ of axioms for voting rules $F : \mathcal{L}(X)^+ \rightarrow 2^X \setminus \{\emptyset\}$, constraint modelling language $\mathbb{L}$ able to express the axioms in $\mathbb{A}$, profile $\succ_{N^*}$, and set $X^* \subseteq X$ of alternatives, we say that a constraint network $\mathcal{N} = \langle \mathcal{V}, \mathcal{D}, C \rangle$ is an **encoding** of the **justification problem** $\langle \succ_{N^*}, X^*, \mathbb{A} \rangle$ in $\mathbb{L}$ if and only if $\mathcal{N}$ has the following components:*

- $\mathcal{V} = (V_1, \ldots, V_{\ell_{n,m}})$ *for* $n = |N^*|$ *and* $m = |X|$
- $\mathcal{D} = D_1 \times \cdots \times D_{\ell_{n,m}}$ *with* $D_i = 2^X \setminus \{\emptyset\}$ *for all* $i \leq \ell_{n,m}$
- $C = C_{\mathbb{A}} \cup \{C_{Goal}\}$

3.3 Minimal Unsatisfiable Subsets

Suppose we have constructed the network $\mathcal{N} = \langle \mathcal{V}, \mathcal{D}, C \rangle$ to encode—in some constraint modelling language $\mathbb{L}$—the justification problem $\langle \succ_{N^*}, X^*, \mathbb{A} \rangle$, and suppose further that $\mathcal{N}$ turns out to be unsatisfiable. Then this means that satisfying the axioms in $\mathbb{A}$, while also selecting an outcome different from X^* for profile $\succ_{N^*}$, is impossible. Understanding where this impossibility originates from exactly will allow us to justify why X^* should (according to some axioms in $\mathbb{A}$) be chosen as the outcome in profile $\succ_{N^*}$.

Explaining the unsatisfiability of a constraint network is a well-studied problem [19, 20]: by computing an MUS for the network we can pinpoint a specific reason for its unsatisfiability. The following theorem formalises the idea that finding a justification essentially boils down to computing an MUS.

Theorem 2 (Correctness). *Given a justification problem $\mathcal{P} = \langle \succ_{N^*}, X^*, \mathbb{A} \rangle$ and a constraint modelling language $\mathbb{L}$ for $\mathbb{A}$, let $\mathcal{N} = \langle \mathcal{V}, \mathcal{D}, C \rangle$ be an encoding of $\mathcal{P}$ in $\mathbb{L}$. Then a pair $\langle \mathcal{A}^N, \mathcal{A}^E \rangle$ of sets of axioms expressible in this language $\mathbb{L}$ is a justification for X^* winning in profile $\succ_{N^*}$ if and only if the following conditions are satisfied:*

- (*i*) *The set $C^* = C_{\mathcal{A}^E} \cup \{C_{Goal}\}$ is an MUS of $\mathcal{N}$.*
- (*ii*) *$\mathcal{A}^N$ is such that both $\mathcal{A}^N \subseteq \mathbb{A}$ and $\mathcal{A}^E \triangleleft \mathcal{A}^N$ hold.*
- (*iii*) *The constraint network $\mathcal{N}^N = \langle \mathcal{V}, \mathcal{D}, C_{\mathcal{A}^N} \rangle$ is satisfiable.*

Before turning to the proof of Theorem 2, let us first explain its practical significance. The three conditions in Theorem 2 outline an *algorithm for computing justifications* (and the theorem establishes the correctness of this algorithm):

(1) Construct $\mathcal{N} = \langle \mathcal{V}, \mathcal{D}, C \rangle$ according to Definition 2.
In doing so, for each constraint $C \in C_{\mathbb{A}}$ keep a record of which axiom $A \in \mathbb{A}$ has given rise to the generation of C.

(2) Check whether $\mathcal{N}$ is satisfiable. If it is, *stop* and announce that justifying the desired outcome X^* is impossible.
Otherwise, repeat the steps below until either a justification has been found or there is no further MUS to explore:

(2*a*) Compute a (new) MUS C^* of the network $\mathcal{N}$ such that $C_{Goal} \in C^*$. Define $\mathcal{A}^E$ as the set of axiom instances corresponding to $C^* \setminus \{C_{Goal}\}$.

```
constraint :: "Reinforcement"
forall( e  in 1..nbElectorates  where card(electorates[e]) > 1,
        p  in assocProfiles[e],
        e1 in 1..nbElectorates  where electorates[e1] ⊂ electorates[e],
        p1 in assocProfiles[e1] where restrictedTo(e1, p, p1),
        e2 in 1..nbElectorates  where electorates[e2] = electorates[e] \ electorates[e1],
        p2 in assocProfiles[e2] where restrictedTo(e2, p, p2) )
  ( F[p1] ∩ F[p2] != {} -> F[p] = (F[p1] ∩ F[p2]) );
```

Table 1: Encoding REINFORCEMENT in MINIZINC, with set-theoretic notation instead of basic MINIZINC constructs for better readability. The array `electorates` stores all possible electorates and `assocProfiles` stores the profiles for each electorate.

(2*b*) Define $\mathcal{A}^{\mathrm{N}}$ as the set of those axioms in $\mathbb{A}$ that have given rise to the axiom instances in $\mathcal{A}^{\mathrm{E}}$.

(2*c*) Check whether sub-network $\mathcal{N}^{\mathrm{N}} = \langle \mathcal{V}, \mathcal{D}, C_{\mathcal{A}^{\mathrm{N}}} \rangle$ is satisfiable. If it is, *stop* and return the justification $\langle \mathcal{A}^{\mathrm{N}}, \mathcal{A}^{\mathrm{E}} \rangle$.

If no justification has been found, *stop* and announce that justifying the desired outcome X^* is impossible.

Of course, solving a justification problem is a highly complex task. But we stress that all computationally demanding steps in our algorithm (namely checking satisfiability of a constraint network and computing an MUS) can be relegated to state-of-the-art constraint solvers for these generic problems.

EXAMPLE 7. Consider the following profile $\succ_{N^*}$, which has also been discussed by Cailloux and Endriss [7]:

$$\begin{array}{ll} \textit{Agent 1:} & a \succ_1 b \succ_1 c \\ \textit{Agent 2:} & a \succ_2 b \succ_2 c \\ \textit{Agent 3:} & c \succ_3 b \succ_3 a \end{array}$$

Suppose we want to justify the outcome $X^* = \{a\}$ (rather than, say, $\{b\}$, which also would not be unreasonable) using a corpus $\mathbb{A}$ including FAITHFULNESS, CANCELLATION, and REINFORCEMENT. The goal constraint is $V_{p(\succ_{N^*})} \neq \{a\}$.

One MUS found by our algorithm includes $V_{p([\succ_1])} = \{a\}$, an instance of FAITHFULNESS. It says that in the subprofile of just agent 1, the set $\{a\}$ must win. It also includes $V_{p([\succ_2,\succ_3])} = \{a, b, c\}$, an instance of CANCELLATION, and this instance of REINFORCEMENT:

$$\left[V_{p([\succ_1])} \cap V_{p([\succ_2,\succ_3])} \neq \emptyset\right] \rightarrow \left[V_{p(\succ_{N^*})} = V_{p([\succ_1])} \cap V_{p([\succ_2,\succ_3])}\right]$$

And indeed, together these three axiom instances constitute an explanation for the desired outcome, with the corresponding axioms serving as the normative basis. △

Let us now prove our theorem.

PROOF OF THEOREM 2. Let $\mathcal{P} = \langle \succ_{N^*}, X^*, \mathbb{A} \rangle$ be a justification problem, $\mathbb{L}$ a constraint modelling language that can express all axioms in $\mathbb{A}$, and $\mathcal{N} = \langle \mathcal{V}, \mathcal{D}, C \rangle$ an encoding of $\mathcal{P}$ in $\mathbb{L}$. Consider any pair $\langle \mathcal{A}^{\mathrm{N}}, \mathcal{A}^{\mathrm{E}} \rangle$ of sets of axioms expressible in $\mathbb{L}$ and define $C^* = C_{\mathcal{A}^{\mathrm{E}}} \cup \{C_{Goal}\}$ and $\mathcal{N}^{\mathrm{N}} = \langle \mathcal{V}, \mathcal{D}, C_{\mathcal{A}^{\mathrm{N}}} \rangle$.

($\Leftarrow$) Assume conditions (*i*)–(*iii*) in the statement of the theorem are satisfied. We need to show that $\langle \mathcal{A}^{\mathrm{N}}, \mathcal{A}^{\mathrm{E}} \rangle$ is a justification for $\mathcal{P}$ (in the sense of Definition 1).

First, let us verify the explanatoriness requirement. Indeed, as C^* is unsatisfiable there can be no voting rule $F \in \mathbb{I}(\mathcal{A}^{\mathrm{E}})$ with $F(\succ_{N^*}) \neq X^*$; and as every proper subset of C^* is satisfiable—including every such subset that includes C_{Goal}—no proper subset of $\mathcal{A}^{\mathrm{E}}$ is sufficiently strong to enforce the same restriction.

Furthermore, relevance and adequacy of the normative basis $\mathcal{A}^{\mathrm{N}}$ follow immediately from condition (*ii*).

Finally, we need to check that the normative basis $\mathcal{A}^{\mathrm{N}}$ is nontrivial. By condition (*iii*), we have that $\mathcal{N}^{\mathrm{N}}$, induced by $\mathcal{A}^{\mathrm{N}}$, is a satisfiable constraint network. Thus, there exists at least one voting rule that satisfies all the axioms in $\mathcal{A}^{\mathrm{N}}$.

($\Rightarrow$) Now assume that $\langle \mathcal{A}^{\mathrm{N}}, \mathcal{A}^{\mathrm{E}} \rangle$ is a justification for $\mathcal{P}$. We need to check that conditions (*i*)–(*iii*) hold.

Condition (*ii*) follows immediately from the adequacy and relevance of $\langle \mathcal{A}^{\mathrm{N}}, \mathcal{A}^{\mathrm{E}} \rangle$. Condition (*iii*) follows from the nontriviality of the normative basis $\mathcal{A}^{\mathrm{N}}$.

It remains to check condition (*i*), i.e., to show that C^* is an MUS of $\mathcal{N}$. This follows from the explanatoriness of $\mathcal{A}^{\mathrm{E}}$: As $F(\succ_{N^*}) = X^*$ for all $F \in \mathbb{I}(\mathcal{A}^{\mathrm{E}})$, adding the goal constraint to $C_{\mathcal{A}^{\mathrm{E}}}$ indeed must result in an unsatisfiable set of constraints. The requirement of no proper subset of $\mathcal{A}^{\mathrm{E}}$ being able to explain the outcome essentially corresponds to C^* being minimal. We only need to check that the special subset $C_{\mathcal{A}^{\mathrm{E}}}$ of C^* is not unsatisfiable. But this clearly cannot be the case, as then $C_{\mathcal{A}^{\mathrm{N}}}$ would be unsatisfiable as well, which is a possibility we have excluded already. □

We stress that justifications computed by our algorithm need not be minimal. If $\mathcal{A}^{\mathrm{E}} = \{A'_1, A'_2\}$, then our algorithm might return $\mathcal{A}^{\mathrm{N}} = \{A_1, A_2\}$ in case $A'_1 \triangleleft A_1$, $A'_1 \triangleleft A_2$, and $A'_2 \triangleleft A_2$, even though $\mathcal{A}^{\mathrm{N}} = \{A_2\}$ would also be sufficient.

In case no two axioms in $\mathbb{A}$ have overlapping sets of instances, this problem cannot occur and our algorithm is guaranteed to always return a minimal justification (if one exists). Related to this point, Theorem 2 in fact establishes the correctness of an entire *family* of algorithms (of which the one sketched earlier is a representative): replacing step (2*b*) by any other procedure for computing a set $\mathcal{A}^{\mathrm{N}} \subseteq \mathbb{A}$ with $\mathcal{A}^{\mathrm{E}} \triangleleft \mathcal{A}^{\mathrm{N}}$ also works. Thus, even if some axioms share some axiom instances, we can refine our algorithm to ensure that $\langle \mathcal{A}^{\mathrm{N}}, \mathcal{A}^{\mathrm{E}} \rangle$ is minimal. Note that this requires a search over all subsets of $\mathbb{A}$, which is exponential in $|\mathbb{A}|$. For small corpora of axioms (such as the corpus of seven axioms considered in this paper) this is relatively unproblematic in practice.

3.4 Proof-of-Concept Implementation

We have implemented our algorithm in Java, encoding axioms in MINIZINC, which has high expressivity and can be used with

several solvers. Our implementation uses GECODE [10] to check satisfiability and FINDMUS [18] for MUS computation.

Using this proof-of-concept implementation, we are able to find justifications for moderately-sized problems. For example, finding all minimal justifications for a given target outcome and profile with three agents and three alternatives (using any of our seven axioms) takes between 5 and 30 minutes on a machine equipped with an AMD Ryzen 5 2600 processor cadenced as 3.4 GHz and 16 GB of memory. The worst-case scenarios are those for which no justification exists. While this is too slow for deployment in a system users can interact with directly, these results nevertheless show the feasibility of our approach and suggest that further algorithmic improvements will make building such a system feasible in the not-too-distant future. As we are going to see next, already now we can use our implementation to run interesting experiments offline.

4 THE EXPLANATORY POWER OF AXIOMS

In this section we present the results of an experiment we carried out to analyse the *explanatory power* of axioms. While this experiment is restricted to small scenarios with three agents and three alternatives, it still provides interesting insights into the usefulness of different axioms in the context of finding justifications.

We shall assume the reader is familiar with the *Borda rule*, the *plurality rule*, and *Condorcet winners* [31].

4.1 Experimental Setup

We restrict attention to scenarios with three agents and three alternatives. For each of the $(3!)^3 = 216$ possible profiles $\succ_{N^*}$ and each of the $2^3 - 1 = 7$ possible outcomes X^*, we want to check whether (and how) X^* can be justified for $\succ_{N^*}$. As our corpus $\mathbb{A}$ of axioms we use the seven axioms defined in Section 2.1. We stress that our variant of REINFORCEMENT is subtly weaker than its classical formulation, because we require all electorates involved to be subsets of $N^* = \{1, 2, 3\}$. Thus, we are not permitted to refer to a hypothetical profile involving agents outside of N^* when justifying a target outcome for a profile in $\mathcal{L}(X)^{N^*}$.

Interestingly, contrary to what one might expect given the impossibility result of Young and Levenglick [30] recalled in Section 2.1, every normative basis we can build from $\mathbb{A}$ is nontrivial for this setting. For example, *Black's rule* (elect the Condorcet winner whenever it exists, otherwise use Borda) satisfies all seven axioms (for $n = 3$ and $m = 3$), even though it violates the classical formulation of REINFORCEMENT. Therefore, in practice we were able to skip step (2*c*) of our algorithm for the experiments reported here. To ensure all (and only) minimal justifications were computed, we used the refinement of our algorithm sketched earlier, in which we explicitly search over all subsets of $\mathbb{A}$.

At this point one *caveat* is in order. Due to the proof-of-concept nature of our system, we were not able to derive all of the results reported here in a fully automated fashion. In particular, the notion of instance used by the MINIZINC compiler does not fully match the theoretical notion best suited to our problem. For this reason, in some cases we required manual post-processing when analysing our experimental results.

Observe that for $n = 3$ and $m = 3$ every profile either has a Condorcet winner (204 profiles) or is an instance of the classical Condorcet Paradox (12 profiles), in which each of the three alternatives occurs exactly once in each of the three positions in an individual preference ordering. The profile ($a \succ_1 b \succ_1 c$, $b \succ_2 c \succ_2 a$, $c \succ_3 a \succ_3 b$) is one such profile. Finally, note that, due to symmetries within the space of all profiles, it is sufficient to only run our experiment on a small subset of the set of all 216 profiles. For example, any justification for outcome $\{a\}$ in profile ($a \succ_1 b \succ_1 c$, $a \succ_2 b \succ_2 c$, $c \succ_3 b \succ_3 a$) can be translated into a justification for $\{b\}$ in profile ($b \succ_1 a \succ_1 c$, $c \succ_2 a \succ_2 b$, $b \succ_3 a \succ_3 c$), given that the latter profile is the result of swapping alternatives a and b as well as agents 2 and 3 in the former profile.[4] We can group the 216 profiles into 10 classes of such "permutation-equivalent" profiles and we have run our system on one representative of each of these classes to obtain the results reported in the sequel.[5]

Classes	Profiles	Axioms in $\mathcal{A}^{\mathrm{N}}$	$\lvert\mathcal{A}^{\mathrm{E}}\rvert$
2	24	{PAR}	2 + 1
		{REI, FAI}	5
3	90	{REI, FAI, CAN}	3
		{REI, PAR, CAN}	4 + 1
2	54	{ANO, NEU, REI, PAR}	6 + 2

Table 2: Justifications for electing the Condorcet winner not using the Condorcet Principle (out of 204 relevant profiles).

4.2 Justifying a Unique Winner

We first consider justifications of a *unique* winner (rather than a set of tied winners). For the 12 paradox-profiles, all belonging to the same equivalence class, our system is unable to justify *any* unique-winner outcome. Upon reflection, this is exactly what we should expect and want to happen: these are perfectly "symmetric" profiles in which no alternative has any special role. Therefore, no combination of normative principles of any kind of philosophical appeal should ever allow us to justify a unique winner.

Next, we turn to the remaining 204 profiles, each of which has a Condorcet winner. Of course, for each such profile our system can easily justify the Condorcet winner as the unique winner using the normative basis {CONDORCET} and an explanation of size 1.

Although short and simple, some may find this an unsatisfactory solution. First, the Condorcet Principle has been criticised in the literature for sometimes leading to unconvincing outcomes—albeit only for larger profiles than we consider here [9]. Second, arguably CONDORCET encodes a rather complex philosophical argument and is maybe better interpreted as a family of voting rules rather than a fundamental normative principle. Therefore, we may also be interested in alternative justifications of the Condorcet winner that do not rely on CONDORCET. Indeed, short justifications are not necessarily better than longer ones.

Using the remaining six axioms, our system is able to derive justifications of the Condorcet winner as the unique winner for 168 of our 204 profiles. No further justification was found for the 36

[4] We stress that this is possible independently of whether or not we are interested in justifications that involve either the anonymity or the neutrality axiom.

[5] These 10 classes are known as the *anonymous and neutral equivalence classes* of preference profiles [15] and have been studied in the context of efficiently generating distributions of profiles from the impartial anonymous and neutral culture [8].

leftover profiles, taken from two equivalence classes. These results are summarised in Table 2. Here "$+k$" in the specification of $|\mathcal{A}^{\mathrm{E}}|$ indicates that the explanation involves k appeals to the "axiom" that the set of winners must be nonempty (as in Example 4). The first 24 cases are profiles in which all three agents agree on the top alternative. We can then use either PARETO twice to exclude the other two alternative, or we can use FAITHFULNESS to establish the obvious winners for the three singleton-profiles and then use REINFORCEMENT twice.[6] Then there are 90 profiles that are similar to the one of Example 7 (with two agents reporting "cancelling" preferences and the third agent thus determining the winner), leading to similar justifications. The remaining 54 cases are all variants of the following example.

Example 8. Consider the following profile:

$$\begin{array}{ll} \textit{Agent 1:} & a \succ_1 b \succ_1 c \\ \textit{Agent 2:} & a \succ_2 c \succ_2 b \\ \textit{Agent 3:} & b \succ_3 a \succ_3 c \end{array}$$

Using PARETO, ANONYMITY, and NEUTRALITY, we can justify $\{a, b\}$ as the winning outcome for subprofile $(\succ_1, \succ_3)$ as in Example 4. A further application of PARETO on $(\succ_2)$, followed by REINFORCEMENT, then yields a as the overall winner. △

Finally, we note that for all of the 204 profiles with a Condorcet winner we were unable to justify any other unique winner. Indeed, inspection of typical such profiles suggests that both other alternatives are "obviously bad" choices. For example, for the profile $(a \succ_1 b \succ_1 c,\ b \succ_2 a \succ_2 c,\ c \succ_3 a \succ_3 b)$, with Condorcet winner a, it seems impossible to come up with any convincing argument for why b or c should win.

4.3 Justifying Multiple Winners

As we have seen, it is not always possible to justify the election of a unique winner. This is related to the fact that most reasonable voting rules are not resolute and may return multiple tied winners. When using our system to try and justify (non-singleton) *sets of winners*, we found that (*i*) we can justify a three-way tie in the case of the 12 paradox-profiles using ANONYMITY and NEUTRALITY, and (*ii*) we cannot justify any non-singleton outcome for any of the other 204 profiles.

The first of these findings is unsurprising and exactly what we should expect to observe. Indeed, the fact that any voting rule that is anonymous and neutral must declare a three-way tie on this kind of paradox-profile is well-known and essentially an instance of an impossibility theorem regarding resolute voting rules that are anonymous and neutral due to Moulin [24].

But what about our second finding? One may feel that for some profiles returning a *pair* of winners seems reasonable.

Example 9. Consider the following profile:

$$\begin{array}{ll} \textit{Agent 1:} & a \succ_1 b \succ_1 c \\ \textit{Agent 2:} & a \succ_2 b \succ_2 c \\ \textit{Agent 3:} & b \succ_3 c \succ_3 a \end{array}$$

Here, a is the Condorcet winner, but the Borda rule will return $\{a, b\}$, as both alternatives receive 4 points. △

There are 18 such profiles (one complete equivalence class) on which the Borda rule disagrees with the Condorcet Principle. By a seminal result due to Young [29], the Borda rule is uniquely characterised by the axioms of NEUTRALITY, REINFORCEMENT, FAITHFULNESS, and CANCELLATION. Therefore, one might expect that our system should be able to justify the Borda outcome for these 18 profiles. Remarkably, this intuition turns out to be wrong. This is due to the fact that Young's definition of REINFORCEMENT is subtly stronger than ours. Our axiom does not allow us to refer to hypothetical profiles with additional agents—a restriction we consider reasonable in the context of providing explanations to human users. If we were to drop this restriction, then it would be possible to derive a justification (albeit a justification of very limited intuitive appeal). The trick consists in inspecting several (much) larger profiles to show that *not* ruling out certain undesired outcomes for the target profile would lead to contradictory conclusions at the level of these larger profiles. This kind of construction is also used in the alternative proof of Young's result given by Hansson and Sahlquist [13].

In a similar vain, one may feel that for some profiles with a Condorcet winner we should return a three-way tie. For example, for $(a \succ_1 b \succ_1 c,\ b \succ_2 a \succ_2 c,\ c \succ_3 a \succ_3 b)$, the plurality rule would declare such a tie. But our corpus of axioms is not sufficiently rich to characterise the plurality rule and thus to provide such a justification [21]. This would require adding further axioms to $\mathbb{A}$.

Finally, we stress that our results, particularly on multiple-winner justification, of course are specific to scenarios with $n = 3$ and $m = 3$ and may not directly generalise to larger scenarios.

5 CONCLUSION

We have put forward a formalisation of the notion of *justification* of a collective decision based on the reported preferences of a group of decision makers, we have outlined a number of application scenarios where this notion plays a central role, and we have presented an algorithm for computing such justifications. Our framework opens up new opportunities for computational social choice and ties in with the recent surge of (renewed) interest in *Explainable AI* [17, 23]. As our experimental study of the explanatory power of axioms demonstrates, our framework can also offer a novel perspective on some of the axioms studied in social choice theory.

We see plenty of exciting directions for future work on this topic. For example, it would be very interesting to let people rate the convincingness of alternative justifications and use the insights thus gained to guide the search for "good" justifications, by building on work on *preferred explanations* in constraint programming [14]. Another important direction for future work relates to improving the efficiency of our approach. While defining what constitutes a justification is language-independent, searching for one is not and some languages might be better suited than others. A rigorous analysis of the computational complexity of the problem of finding justifications should provide insight into the best way to follow towards practical feasibility. Some variants of the problem—such as searching for a specific type of justification—might also be easier to solve than others. The insights thus gained may serve as a basis for the design of tailor-made algorithms and heuristics.

[6] Observe that PARETO implies FAITHFULNESS, so any justification involving the latter can always be rewritten as a justification using the former instead. This does not render FAITHFULNESS redundant. Indeed, a justification relying on FAITHFULNESS rather than the more presumptuous PARETO might be preferable to some audiences.

REFERENCES

[1] Kenneth J. Arrow, Amartya K. Sen, and Kotaro Suzumura (Eds.). 2002. *Handbook of Social Choice and Welfare*. Vol. 1. Elsevier.

[2] Bernhard Beckert, Thorsten Bormer, Rajeev Goré, Michael Kirsten, and Carsten Schürmann. 2017. An Introduction to Voting Rule Verification. In *Trends in Computational Social Choice*, Ulle Endriss (Ed.). AI Access.

[3] Khaled Belahcene, Yann Chevaleyre, Christophe Labreuche, Nicolas Maudet, Vincent Mousseau, and Wassila Ouerdane. 2018. Accountable Approval Sorting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI-2018)*.

[4] Christain Bessiere. 2006. Constraint Propagation. In *Handbook of Constraint Programming*, Francesca Rossi, Peter van Beek, and Toby Walsh (Eds.). Elsevier.

[5] Steven J. Brams and Peter C. Fishburn. 2002. Voting Procedures. In *Handbook of Social Choice and Welfare*, Kenneth J. Arrow, Amartya K. Sen, and Kotaro Suzumura (Eds.). Vol. 1. Elsevier.

[6] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia (Eds.). 2016. *Handbook of Computational Social Choice*. Cambridge University Press.

[7] Olivier Cailloux and Ulle Endriss. 2016. Arguing about Voting Rules. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2016)*. IFAAMAS.

[8] Ömer Eğecioğlu. 2009. Uniform Generation of Anonymous and Neutral Preference Profiles for Social Choice Rules. *Monte Carlo Methods and Applications* 15, 3 (2009), 241–255.

[9] Peter C Fishburn. 1974. Paradoxes of Voting. *American Political Science Review* 68, 2 (1974), 537–546.

[10] Gecode Team. 2006. Gecode: Generic Constraint Development Environment. http://www.gecode.org. (2006).

[11] Christian Geist and Dominik Peters. 2017. Computer-Aided Methods for Social Choice Theory. In *Trends in Computational Social Choice*, Ulle Endriss (Ed.). AI Access.

[12] Umberto Grandi and Ulle Endriss. 2013. First-Order Logic Formalisation of Impossibility Theorems in Preference Aggregation. *Journal of Philosophical Logic* 42, 4 (2013), 595–618.

[13] Bengt Hansson and Henrik Sahlquist. 1976. A Proof Technique for Social Choice with Variable Electorate. *Journal of Economic Theory* 13, 2 (1976), 193–200.

[14] Ulrich Junker. 2004. QuickXplain: Preferred Explanations and Relaxations for Over-constrained Problems. In *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI-2004)*. AAAI Press.

[15] Alexander Karpov. 2018. *An Informational Basis for Voting Rules*. Research Paper WP BRP 188. Higher School of Economics.

[16] Michael Kirsten and Olivier Cailloux. 2018. Towards Automatic Argumentation about Voting Rules. In *Actes de la 4ème Conférence Nationale sur les Applications Pratiques de l'Intelligence Artificielle (APIA-2018)*.

[17] Pat Langley. 2019. Explainable, Normative, and Justified Agency. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI-2019)*. AAAI Press.

[18] Kevin Leo and Guido Tack. 2017. Debugging Unsatisfiable Constraint Models. In *Proceedings of the 14th International Conference on AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (CPAIOR-2017)*. Springer.

[19] Mark H. Liffiton, Alessandro Previti, Ammar Malik, and João Marques-Silva. 2016. Fast, Flexible MUS Enumeration. *Constraints* 21, 2 (2016), 223–250.

[20] Mark H. Liffiton and Karem A. Sakallah. 2008. Algorithms for Computing Minimal Unsatisfiable Subsets of Constraints. *Journal of Automated Reasoning* 40, 1 (2008), 1–33.

[21] Vincent Merlin. 2003. The Axiomatic Characterizations of Majority Voting and Scoring Rules. *Mathématiques & Sciences Humaines* 41, 161 (2003), 87–109.

[22] Michela Milano. 2018. Twenty Years of Constraint Programming (CP) Research. *Constraints* 23, 2 (2018), 155–157.

[23] Tim Miller. 2018. Explanation in Artificial Intelligence: Insights from the Social Sciences. *Artificial Intelligence* 267 (2018), 1–38.

[24] Hervé Moulin. 1983. *The Strategy of Social Choice*. North-Holland.

[25] Nicholas Nethercote, Peter J. Stuckey, Ralph Becket, Sebastian Brand, Gregory J. Duck, and Guido Tack. 2007. MiniZinc: Towards a Standard CP Modelling Language. In *Proceedings of the 13th International Conference on Principles and Practice of Constraint Programming (CP-2007)*. Springer.

[26] Ariel D. Procaccia. 2019. Axioms Should Explain Solutions. In *The Future of Economic Design*, Jean-François Laslier, Hervé Moulin, M. Remzi Sanver, and William S. Zwicker (Eds.). Springer.

[27] Francesca Rossi, Peter van Beek, and Toby Walsh (Eds.). 2006. *Handbook of Constraint Programming*. Elsevier.

[28] Pingzhong Tang and Fangzhen Lin. 2009. Computer-aided Proofs of Arrow's and other Impossibility Theorems. *Artificial Intelligence* 173, 11 (2009), 1041–1053.

[29] H. Peyton Young. 1974. An Axiomatization of Borda's Rule. *Journal of Economic Theory* 9, 1 (1974), 43–52.

[30] H. Peyton Young and Arthur Levenglick. 1978. A Consistent Extension of Condorcet's Election Principle. *SIAM Journal on Applied Mathematics* 35, 2 (1978), 285–300.

[31] William S. Zwicker. 2016. Introduction to the Theory of Voting. In *Handbook of Computational Social Choice*, Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia (Eds.). Cambridge University Press.