ECAI 2014
T. Schaub et al. (Eds.)

doi:10.3233/978-1-61499-419-0-183

Eliciting a Suitable Voting Rule via Examples

Olivier Cailloux and **Ulle Endriss**[1]

Abstract. We address the problem of specifying a voting rule by means of a series of examples. Each example consists of the answer to a simple question: how should the rule rank two alternatives, given the positions at which each voter ranks the two alternatives? To be able to formalise this elicitation problem, we develop a novel variant of classical social choice theory in terms of associations of alternatives with vectors of ranks rather than the common associations of voters with preference orders. We then define and study a class of voting rules suited for elicitation using such answers. Finally, we propose and experimentally evaluate several elicitation strategies for arriving at a good approximation of the target rule with a reasonable number of queries.

1 INTRODUCTION

Voting theory is concerned with the analysis of rules for conducting an election [10]. In recent years, there has been a marked interest in voting theory within AI, for two reasons: first, voting is relevant to AI-related applications such as recommender systems, search engines, and multiagent systems; and, second, techniques developed in AI and computer science more generally, such as complexity theory and knowledge representation, turned out to be useful for the analysis of voting rules [2].

In this work, we consider the problem of identifying an initially unknown rule that is suitable in a given situation. Consider a committee that wants to decide on a voting rule to use for some future decisions it will have to take. How can this committee articulate its requirements regarding the rule? The literature on voting theory provides a number of *axioms*, such as homogeneity or monotonicity, that are satisfied by some rules and not by others [10]. Following this approach, the committee could select the voting rule that satisfies the axioms it considers most important. This might however be difficult to implement. For example, the committee might choose axioms that are mutually incompatible or that do not determine a single rule. Considering the range of surprising paradoxes and impossibility theorems in social choice theory, it is also likely that they will not fully comprehend the implications of adopting a given axiom.

We propose to treat the problem of selecting a voting rule as a problem of preference elicitation. In classical voting theory, each voter provides a ranking (a linear order) of the alternatives on the table. Thus, we can identify each alternative with the vector of ranks it receives, one for each voter. We shall assume the voting rule our committee has in mind can be specified in terms of an ordering over these rank-vectors: an alternative wins if the rank-vector it is associated with is not dominated by any other rank-vector occurring in the election instance at hand (this may be considered a basic axiom our committee accepts). To determine which rule is best for our committee, we ask questions about the ideal behaviour of the rule. Each question takes the following form: we present two rank-vectors to the committee and ask which of them they want the voting rule to prefer or whether they think the rule should remain indifferent between the two. Every answer is interpreted as a constraint on the rule. For example, a committee wanting to favor "consensual" alternatives may prefer a rank-vector composed only of ranks 2 and 3 to one consisting of ranks 1 and 4.

To fully specify a voting rule requires a huge number of queries, even for moderate numbers of voters and alternatives. We therefore are interested in approximating the target rule as well as possible by means of what we call a *robust* voting rule: the rule returning the union of the sets of winners of all voting rules compatible with the constraints elicited at a given point.

In this paper, we introduce and study a class of voting rules suited for such questioning process.

Our approach is inspired by a similar idea used in multiple criteria decision aiding [5]. To obtain a model of the preferences of a decision maker [4, 8], or a group of decision makers [3, 7], looking for a preference model in some *a priori* defined class of possible models, the preference elicitation process asks for constraints given by the decision makers in the form of examples of input and related expected output of the model. Robust results are then computed by considering every model of the considered class that is compatible with the constraints given so far. The process is iterated by asking more questions and showing intermediate results until the decision makers are satisfied or some stopping criterion is met.

Preference learning [6] is another field concerned with methods for obtaining preference models about various kinds of objects, often preferences of consumers over sets of goods. Our approach, however, is about eliciting information about something more abstract, namely a preferred voting rule. Therefore, a crucial part of the problem that we explore in this paper is to develop a way of asking simple questions that can serve as examples for directing the elicitation process.

The remainder of the paper is organised as follows. Our formal framework for modelling voting rules is presented in Section 2. In this framework, we adopt an unusual perspective and describe elections in terms of mappings from alternatives to rank-vectors rather than the familiar profiles (which are mappings from voters to preference orders). In Section 3, we introduce the concept of a voting rule that is based on either a preorder or a weak order on rank-vectors. We prove several results that shed light on the structure of these classes of rules and show how they relate to the rules that are definable by the answers to the type of questions we are interested in here. In Section 4, we propose different strategies for deciding which questions to ask at what point in an elicitation process and we provide experimental results on the performance of different elicitation strategies. Section 5 concludes with a brief discussion of future directions.

[1] Institute for Logic, Language and Computation, University of Amsterdam, email: olivier.cailloux@uva.nl, ulle.endriss@uva.nl

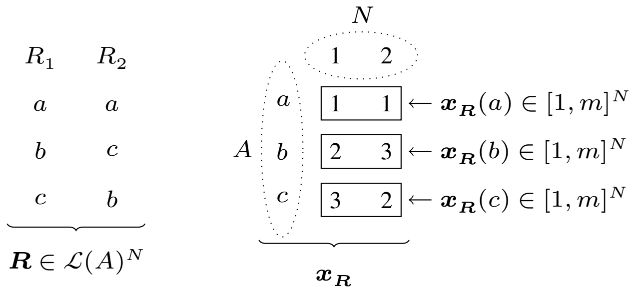


Figure 1. A Profile and the corresponding Rank-Profile

2 FORMAL FRAMEWORK

Let N be a finite set of *voters*, with $|N| = n$, and let A be a finite set of *alternatives*, with $|A| = m$. We write $\mathcal{L}(A) \subseteq \mathcal{P}(A \times A)$ for the set of linear orders on A. Recall that a linear order is a complete, transitive, and antisymmetric binary relation. We use linear orders to model *preferences* over alternatives. A *profile* is a function $\boldsymbol{R} : N \to \mathcal{L}(A)$ mapping each voter to her preference order. We write R_i rather than $\boldsymbol{R}(i)$ for the preference of voter i. The set of all possible profiles is $\mathcal{L}(A)^N$. A *voting rule* F, given a profile on A, returns a non-empty subset of A, which are the winning alternatives according to F:

$$F\colon \mathcal{L}(A)^N \to \mathcal{P}(A) \setminus \{\emptyset\}. \tag{1}$$

This is the standard model of classical voting theory familiar from the literature [10]. Let us now change perspective and consider a profile $\boldsymbol{R}$ from the viewpoint of one alternative $a \in A$. Each voter $i \in N$ has ranked a at a certain position in her own preference order. That is, we can think of a as a function mapping voters to ranks (numbers between 1 and m). When taking this perspective, we shall identify alternatives with *rank-vectors*. Formally, a rank-vector is a function $x : N \to [1, m]$ mapping each voter to a rank.[2] We write x_i for $x(i)$. The set of all possible rank-vectors is $[1, m]^N$.

Given a profile $\boldsymbol{R} \in \mathcal{L}(A)^N$ and an alternative $a \in A$, the rank-vector x associated with a by $\boldsymbol{R}$ is defined so that x_i is the rank of a according to R_i, i.e. $x_i = k + 1$ where k is the number of alternatives strictly better than a in R_i. The *rank-profile* corresponding to a profile $\boldsymbol{R} \in \mathcal{L}(A)^N$ thus is $\boldsymbol{x_R} : A \to [1, m]^N$ such that $\boldsymbol{x_R}(a)$ is the rank-vector associated with a by $\boldsymbol{R}$. The correspondence between $\boldsymbol{R}$ and $\boldsymbol{x_R}$ is illustrated in Figure 1. When the identity of the alternative to which a rank-vector corresponds is not important, we denote a rank-vector by x rather than $\boldsymbol{x_R}(a)$.

Note that not all combinations of m rank-vectors are admissible as rank-profiles. As we only deal with linear orders as basic preferences, rank-profiles featuring multiple times the same rank for a given voter are not allowed. The set of admissible rank-profiles is therefore:

$$\{\boldsymbol{x} : A \to [1, m]^N \mid \forall i \in N, a, b \in A : \boldsymbol{x}(a)_i \neq \boldsymbol{x}(b)_i\}. \tag{2}$$

Note that a rank-profile contains the same information as a profile: given an admissible rank-profile $\boldsymbol{x}$, there is a unique profile $\boldsymbol{R} \in \mathcal{L}(A)^N$ such that $\boldsymbol{x_R} = \boldsymbol{x}$, and *vice versa*. We can therefore consider a voting rule as operating on rank-profiles, rather than on profiles. Given a voting rule F, we define the corresponding rank-voting rule as the function F' that selects the winning alternatives out of an admissible rank-profile: $F'(\boldsymbol{x_R}) = F(\boldsymbol{R})$. Conversely, to each rank-voting rule corresponds a unique standard voting rule.[3]

In this paper, we will only be concerned with voting rules that are *neutral* [10], i.e. rules that treat all alternatives symmetrically. Just as in the standard framework assuming *anonymity* (symmetry w.r.t. voters) permits us to model profiles as multisets (rather than vectors) of preferences, in our model neutrality permits us to simplify notation and to model rank-profiles as sets (rather than vectors) of rank-vectors. (Observe that we can indeed work with sets rather than multisets because no rank-profile can include the same rank-vector more than once.) Thus, we can think of a voting rule as selecting a subset of rank-vectors from a given set of rank-vectors. We write X for the set of available rank-vectors in a rank-profile $\boldsymbol{x_R}$ (i.e. for $\{\, x \in [1, m]^N \mid \exists a \in A : x = \boldsymbol{x_R}(a) \,\}$), which becomes the input to our voting rule using this simplified notation. We call X a *voting instance*. Let $\mathcal{X}$ denote the set of all admissible voting instances. Having a profile $\boldsymbol{R}$, with X the corresponding voting instance, we define $\boldsymbol{x_R}(a) \in F''(X) \Leftrightarrow a \in F(\boldsymbol{R})$. There is thus a bijection between these simplified voting rules selecting subsets of rank-vectors and neutral classical voting rules. By a slight abuse of notation, we write $F(X)$ rather than $F''(X)$.

When giving examples of rank-vectors, we only use one-digit ranks. Therefore, instead of writing the rank-vector as a tuple of ranks, we write it as a string of ranks. For example, instead of writing $x = (3, 2)$ we will write $x = 32$. Furthermore, we will write $F_1 \subseteq F_2$ if $F_1(X) \subseteq F_2(X)$ for every voting instance $X \in \mathcal{X}$.

Let us now define a few classical properties and voting rules that we will need, all translated into our framework of rank-vectors. Observe that for two rank-vectors x and y, $x_i < y_i$ means that voter i prefers the alternative associated with x to the alternative associated with y. A *Condorcet winner* is rank-vector that would beat every other rank-vector in a given set of rank-vectors in a pairwise majority contest.

Definition 1 (Condorcet winner)**.** *Let $X \in \mathcal{X}$. A rank-vector $x \in X$ is a Condorcet winner if $|\{\, i \mid x_i < y_i \,\}| > \frac{n}{2}$ for all $y \in X \setminus \{x\}$.*

Definition 2 (Condorcet consistency)**.** *A voting rule F is Condorcet-consistent if x being a Condorcet winner for X implies $F(X) = \{x\}$.*

Definition 3 (PSR)**.** *A voting rule F is a positional scoring rule (PSR) if there exists a function $s : [1, m] \to \mathbb{R}$, mapping ranks to scores, such that for every voting instance $X \in \mathcal{X}$ we get:*

$$F(X) = \operatorname*{arg\,max}_{x \in X} \Big(\sum_{i \in N} s(x_i)\Big)\,. \tag{3}$$

It is common to require the scores to be non-increasing with increasing ranks. We do not impose this restriction here.

We now define the Bucklin rule. We will use it as an example of rule that is not a PSR but is included in the class of voting rules defined in Section 3.

Definition 4 (Bucklin rule)**.** *Let $X \in \mathcal{X}$. For $k \in [0, m]$ and $x \in X$, define $r_{\leq k}(x)$ as the number of ranks in x that are better (thus lower) than, or as good as, k, i.e. $r_{\leq k}(x) = |\{\, i \in N \mid x_i \leq k \,\}|$. The Bucklin threshold t given X is the smallest number such that some alternative has a majority of ranks at least as good as t, thus $t = \min\{k \in \mathbb{N} \mid \exists x \in X : r_{\leq k}(x) > \frac{n}{2}\}$. The Bucklin rule is the voting rule F which, given $X \in \mathcal{X}$, and considering t the Bucklin*

[2] Throughout the text, bracket notation such as $[1, m]$ designates intervals in the natural numbers $\mathbb{N}$, not in $\mathbb{R}$.

[3] There are some similarities with the informational approach to social choice theory using utilities rather than ordinal preferences. In that approach, it is natural to view an alternative as being associated with a set of numbers, representing the utilities given by each voter to that alternative [1, 9].

threshold given X, selects as winners the alternatives that attain the maximum score as evaluated by $r_{\leq t}$:

$$F(X) = \arg\max_{x \in X}(r_{\leq t}(x)). \quad (4)$$

Fact 1. *The Bucklin rule is not a PSR.*

Indeed, consider the voting instances $X_1 = \{13, 24, 32, 41\}$ and $X_2 = \{11, 22, 33, 44\}$, with $n = 2$ and $m = 4$. Under Bucklin, the winners for X_1 are $\{13, 32\}$ and the only winner for X_2 is 11. For Bucklin to be a PSR we would need, from the first instance, $s(1) = s(2)$, which contradicts the second instance.

3 VOTING FROM A PREORDER

In this section we study several new classes of voting rules. We first introduce two simple classes of voting rules: the preorder-based rules and the weak order-based rules. We then present two ways of defining voting rules from answers to the elicitation questions we are interested in. Our goal is to show the links between the rules that can be defined from the questioning process we propose and the classes of preorder and weak order-based rules, as well as how these compare to classical voting rules. Specifically, we will show the following. First, the class of weak order-based rules is a strict superset of the PSR's and a strict subset of the preorder-based rules. Second, the class of preorder-based rules equals the class of rules that can be defined from our questions. The last result holds for both proposed ways of interpreting the answers.

A *preorder*, denoted $\succsim$, is a transitive and reflexive binary relation. Its asymmetric part is denoted $\succnsim$, its symmetric part $\sim$. Let $\mathcal{Z}$ be the set of all preorders defined over $[1, m]^N$.

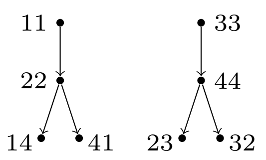
Definition 5 (Voting from a preorder)**.** *Let $\succsim$ be a preorder on $[1, m]^N$. Given $X \in \mathcal{X}$, the voting rule $F_{\succsim}$ returns as winners those rank-vectors which are maximal under $\succsim$ in X:*

$$F_{\succsim}(X) = \{\, x \in X \mid \nexists y \in X : y \succnsim x \,\}. \quad (5)$$

A voting rule F is called *preorder-based* if there exists a preorder $\succsim$ in $\mathcal{Z}$ such that $F = F_{\succsim}$.

A *weak order* is a complete preorder. We use the symbol $\succeq$ to denote a weak order over the set of rank-vectors $[1, m]^N$, its asymmetric part being denoted $\succneq$. Let $\mathcal{W}$ denote the set of weak orders defined over $[1, m]^N$. Observe that $\mathcal{W} \subseteq \mathcal{Z}$. We call a voting rule *weak order-based* if there exists a weak order $\succeq$ in $\mathcal{W}$ such that $F = F_{\succeq}$. Any voting rule that is weak order-based is also preorder-based. The following example shows that the converse is not true.

Example 1. *Consider the voting instances X_1 and X_2 as well as the preorder $\succsim$ shown below, with $n = 2$, $m = 4$. A down-arrow represents $\succnsim$, the transitive closure is left implicit, arrows implied by reflexivity are omitted and isolated rank-vectors are not shown.*



X_1	X_2
$\boxed{11}$	14
22	$\boxed{22}$
$\boxed{33}$	$\boxed{33}$
44	41

Let F be the preorder-based rule based on $\succsim$; let us show that it is not weak order-based. When given the voting instances X_1 and X_2, F elects the boxed rank-vectors. For any rule $F_{\succeq}$, with $\succeq$ a weak order, satisfying the instances X_1 and X_2, it must be the case that $\succeq$ is indifferent between 11 and 33, and also between 22 and 33. By transitivity of indifference, $\succeq$ thus must be indifferent between 11 and 22, but this is impossible while also ensuring 22 is not a winner for X_1.

3.1 Relationship to classical voting rules

The class of preorder-based voting rules, including in particular rules based on weak orders, is certainly an intuitively appealing class to consider. We will now see that it is a generalisation of the PSR's, but not one that is so general as to encompass all voting rules.

Proposition 2. *Every PSR is weak order-based.*

Proof. Take any PSR F, defined by scoring function s. Define the weak order $\succeq$ such that $x \succeq y$ if and only if $\sum_i s(x_i) \geq \sum_i s(y_i)$. Then $F = F_{\succeq}$ by construction. □

Our next result shows that there are weak order-based voting rules that are not PSR's (recall that Bucklin is not a PSR by Fact 1).

Proposition 3. *The Bucklin rule is weak order-based.*

Proof. Given a rank $k \in [1, m]$ and a number of voters α with $\frac{n}{2} < \alpha \leq n$, define $X_{k=\alpha} \subseteq [1, m]^N$ as the set of rank-vectors which do not have a majority of ranks lower than k and have exactly α ranks lower than or equal to k. Thus $X_{k=\alpha}$ is:

$$\{\, x \in [1, m]^N \mid r_{\leq k-1}(x) \leq \tfrac{n}{2} \text{ and } r_{\leq k}(x) = \alpha \,\}. \quad (6)$$

Observe that the sets $X_{k=\alpha}$ form a partition (a complete and disjoint covering) of $[1, m]^N$. Now define a weak order $\succeq$ on $[1, m]^N$. The sets $X_{k=\alpha}$ define the equivalence classes of $\succeq$, and $\succeq$ orders these equivalence classes as follows: $X_{k=\alpha} \succneq X_{\ell=\beta}$ if and only if $k < \ell$ or both $k = \ell$ and $\alpha > \beta$.

Now let t be the Bucklin threshold for a given voting instance X and define $u = \max_{x \in X} r_{\leq t}(x)$. Then x is a Bucklin winner if and only if $x \in X_{t=u}$, which is the case if and only if $x \in F_{\succeq}(X)$. Hence, $F_{\succeq}$ is the Bucklin rule. □

Proposition 4. *For $n = 3$ and $m = 4$, no Condorcet-consistent voting rule is preorder-based.*

Proof. Take any voting rule F that is Condorcet-consistent. Now consider the following three voting instances (the boxed rank-vectors are the Condorcet winners).

X_1	X_2	X_3
231	312	123
342	423	234
$\boxed{123}$	$\boxed{231}$	$\boxed{312}$
414	144	441

For the sake of contradiction, assume there exists a preorder $\succsim$ in $\mathcal{Z}$ such that $F_{\succsim} = F$. F must elect the Condorcet winner 123 in X_1. To have $F_{\succsim}(X_1) = \{123\}$, we must have $123 \succnsim 231$. Similarly, from the instances X_2 and X_3 we obtain that $231 \succnsim 312$ and $312 \succnsim 123$. Hence, we get a cycle and $\succsim$ is not a preorder. □

Observe that if $m > 4$, we can construct a similar example: simply suppose that every voter ranks the ith alternative (for $i > 4$) always in the ith position. Also, if $n > 3$ and n is divisible by 3, we can produce a variant of the above example with three groups of voters of equal size voting exactly like the three individual voters above.

3.2 Constraints and robust voting rules

We now want to approach the problem of specifying a weak order-based voting rule by means of a series of examples provided to us by a committee that needs to identify a rule they want to employ. Each example amounts to imposing a constraint on the voting rule, by fixing the relative ordering of two rank-vectors. Given two rank-vectors x and y, we may say that we want to place x above y, that we want to place x below y, or that we want to place them both in the same indifference class. Formally, we do this by defining two binary relations, $>^C$ and $\sim^C$, on the set $[1,m]^N$ of rank-vectors. Given two rank-vectors x and y, $x >^C y$ says that x must be strictly better than y, while $x \sim^C y$ says that x must be equivalent to y.

Given constraints $C = (>^C, \sim^C)$, we say that a preorder $\succsim \in \mathcal{Z}$ satisfies C if $>^C \subseteq \succnsim$ and $\sim^C \subseteq \sim$. We define $\mathcal{Z}_C$ as the set of preorders satisfying C, and we say that C is consistent if $\mathcal{Z}_C \neq \emptyset$. Similarly, $\mathcal{W}_C$ denotes the set of weak orders satisfying C.

Definition 6 (Robust voting rule)**.** *For any nonempty set of preorders $S \subseteq \mathcal{Z}$, the robust voting rule F_S returns as winners all those rank-vectors that win under some rule associated with a preorder in S:*

$$F_S(X) = \bigcup_{\succsim \in S} F_{\succsim}(X). \tag{7}$$

Such a rule is called *robust*, because we will use it to make sure that we do not exclude a potential winner, facing incomplete preference information from the committee about which preorder should be used. It is thus robust against this kind of information incompleteness.

This gives two ways of defining a robust voting rule, given constraints C: the rule $F_{\mathcal{Z}_C}$, considering all preorders satisfying C, and the rule $F_{\mathcal{W}_C}$, considering only the compatible weak orders. We can think of these rules as an approximation of the voting rule the committee wants to communicate to us. We now study the relationships between the preorder-based rules and such robust rules. We first state without proof some important and useful facts as a lemma. The proofs follow from the relevant definitions.

Lemma 5. *The following facts hold.*

- (i) *$\succnsim \subseteq \succnsim' \in \mathcal{Z}$ implies $F_{\succsim'} \subseteq F_{\succsim}$.*
- (ii) *$\emptyset \subset S \subseteq S' \subseteq \mathcal{Z}$ implies $F_S \subseteq F_{S'}$.*
- (iii) *$F_{\{\succsim\}} = F_{\succsim}$ for all preorders $\succsim \in \mathcal{Z}$.*

Let $tr(R)$ denote the transitive closure of a binary relation R and let R^{-1} denote the inverse of R. Let $\Delta_{[1,m]^N}$ be the identity relation on $[1,m]^N$. For every consistent set of constraints $C = (>^C, \sim^C)$, define $\succsim^C \in \mathcal{Z}$ as the following preorder:

$$\succsim^C = tr(>^C \cup \sim^C \cup \sim^{C^{-1}}) \cup \Delta_{[1,m]^N}. \tag{8}$$

Fact 6. *For any consistent set of constraints C, $\succsim^C$ is the smallest preorder satisfying C, meaning that $\succsim^C \in \mathcal{Z}_C$ and for all $\succsim \in \mathcal{Z}_C$: $\succsim^C \subseteq \succsim$ and $\succnsim^C \subseteq \succnsim$.*

We first show that a robust voting rule, when considering preorders, necessarily corresponds to some preorder-based rule.

Proposition 7. *Let C be a set of consistent constraints. Then the robust voting rule induced by C is equal to the voting rule based on the minimal preorder associated with C:*

$$F_{\mathcal{Z}_C} = F_{\succsim^C}. \tag{9}$$

Proof. As $\succsim^C \in \mathcal{Z}_C$, $F_{\succsim^C} \subseteq F_{\mathcal{Z}_C}$ follows from Lemma 5, parts (ii) and (iii). For the other direction, from the definition of a robust rule we get $F_{\mathcal{Z}_C}(X) = \bigcup_{\succsim \in \mathcal{Z}_C} F_{\succsim}(X)$ for all voting instances X. For each of these $\succsim$, by Fact 6, we have $\succnsim^C \subseteq \succnsim$; and thus we get $F_{\succsim} \subseteq F_{\succsim^C}$ from Lemma 5, part (i). Hence, $F_{\mathcal{Z}_C}(X) = \bigcup_{\succsim \in \mathcal{Z}_C} F_{\succsim}(X) \subseteq F_{\succsim^C}(X)$ for all X. □

Conversely, any preorder-based rules can be defined using some constraints.

Proposition 8. *Let $\succsim \in \mathcal{Z}$ be a preorder and let $C = (\succnsim, \sim)$ be the corresponding constraints. Then C is consistent and the robust rule induced by C is equal to the rule based on $\succsim$:*

$$F_{\mathcal{Z}_C} = F_{\succsim}. \tag{10}$$

Proof. C is consistent as $\succsim$ satisfies it. And as $\succsim = \succsim^C$, the preorder induced by C, the result follows from Proposition 7. □

The following proposition shows that our earlier results still hold if we consider only weak orders instead of all preorders.

Proposition 9. *Let C be a set of consistent constraints. Then the robust voting rule induced by C together with completeness is equal to the voting rule based on the minimal preorder associated with C:*

$$F_{\mathcal{W}_C} = F_{\succsim^C}. \tag{11}$$

Proof. We have $F_{\succsim^C} = F_{\mathcal{Z}_C}$ from Proposition 7, and as $\mathcal{W}_C \subseteq \mathcal{Z}_C$, $F_{\mathcal{W}_C} \subseteq F_{\succsim^C}$ follows from Lemma 5, part (ii).

To obtain $F_{\succsim^C} \subseteq F_{\mathcal{W}_C}$, we take $X \in \mathcal{X}$ and $x \in F_{\succsim^C}(X)$, and show that $x \in F_{\mathcal{W}_C}(X)$. We know that no rank-vector y among those in X is better than x according to $\succsim^C$. Therefore, a weak order can be defined over $[1,m]^N$, by completing $\succsim^C$, that satisfies C and has x as a maximal element among X. That weak order being a member of $\mathcal{W}_C$, we obtain $x \in F_{\mathcal{W}_C}(X)$. □

Denoting the set of consistent constraints by $\mathcal{C} = \{\, C \mid \mathcal{Z}_C \neq \emptyset \,\}$, Propositions 7, 8, 9 show the equality of the following three classes of voting rules: the robust rules using preorders, $\{\, F_{\mathcal{Z}_C}, C \in \mathcal{C} \,\}$; the robust rules using weak orders, $\{\, F_{\mathcal{W}_C}, C \in \mathcal{C} \,\}$; and the preorder-based voting rules, $\{\, F_{\succsim}, \succsim \in \mathcal{Z} \,\}$. Furthermore, Propositions 7 and 9 provide us with a convenient way to compute winners of a robust rule, given some constraints C.

4 ELICITING VOTING RULES

Suppose we have been asked to implement a voting rule for the use of a committee and we need to elicit the views of that committee regarding the rule to be implemented. We shall assume that our committee has a weak order $\geq$ over the set of rank-vectors $[1,m]^N$ in mind, so that their preferred voting rule is $F_{\geq}$. We call $F_{\geq}$ the *target rule*. We want to define a rule F, as resolute as possible (i.e., returning as few tied winners as possible), such that $F_{\geq} \subseteq F$.

Besides being weak order-based, we shall make two further assumptions regarding the target rule. First, we assume the committee will respect the Pareto principle. Define Pareto dominance over rank-vectors as $x \blacktriangleright y$ iff $[\forall i \in N : x_i \leq y_i] \wedge [x \neq y]$. We assume that $\geq$ is an extension of the Pareto dominance relation (thus $\blacktriangleright \subseteq \gneq$). Second, we assume that $\geq$ is indifferent to a permutation of the ranks in a rank-vector. Writing $p(x)$ for the rank-vector resulting from a permutation p of the ranks of a rank-vector

x, we have thus that $\forall x, y \in [1, m]^N$, $\forall$permutations $p, q : x \succeq y \Rightarrow p(x) \succeq q(y)$. We thus start out with a set of constraints C_0 representing these two assumptions: $C_0 = (\blacktriangleright, B)$ where $B = \{ (x, p(x)), \forall x \in [1, m]^N, \forall \text{ permutation } p \}$.

We then ask questions to the committee to elicit the target rule. A question is an unordered pair of rank-vectors (x, y). They answer each question according to their weak order: $x \succsim y$, $y \succsim x$ or $(x \succeq y) \wedge (y \succeq x)$. Starting from constraints $C_k = (>^{C_k}, \sim^{C_k})$, obtained after k answers, we can build C_{k+1} as follows. If the answer is $x \succsim y$, $C_{k+1} = (>^{C_k} \cup \{ (x, y) \}, \sim^{C_k})$. If the answer is that x and y are equivalent, $C_{k+1} = (>^{C_k}, \sim^{C_k} \cup \{ (x, y) \})$.

Having elicited constraints C_k, we can define a robust voting rule selecting the potential winners according to the preferential information known so far. This is by definition $F_{\mathcal{W}_{C_k}}$, the rule selecting as winners all alternatives that win in at least one weak order satisfying C_k. This process leads to a sequence of embedded voting rules that get more and more refined, approaching the target rule:

$$F_{\succeq} \subseteq F_{\mathcal{W}_{C_{k+1}}} \subseteq F_{\mathcal{W}_{C_k}} \subseteq \cdots \subseteq F_{\mathcal{W}_{C_0}}.$$

We now want to find a good way of asking questions (i.e. of choosing unordered pairs of rank-vectors) such that the rule $F_{\mathcal{W}_C}$ obtained at the end of the questioning process is as "close" as possible to $F_{\succeq}$.

4.1 Elicitation strategies

To determine which question should be asked at a given step (with C the current set of constraints at that step and $\succsim^C$ the preorder induced by C), we define a fitness measure $\mathit{fit}(x, y, C) \in \mathbb{R}_+$, a heuristic that indicates how good we expect a question (x, y) to be. A fitness measure is defined for all pairs of rank-vectors x, y that are incomparable in $\succsim^C$. Pairs for which status is already known in $\succsim^C$ are assigned a fitness of zero. An elicitation strategy then simply picks one of the maximally fit pairs (ties are broken lexicographically). Here are four strategies, defined in terms of their respective fitness functions.

Optimistic This strategy takes the fitness to be proportional to the number of rank-vectors dominated by x or y, but not both. Define $\succnsim^C(x)$ as the set of rank-rectors dominated by x according to the strict version of $\succsim^C$. Then, $\mathit{fit}_{\text{o}}(x, y, C) = \left|\succnsim^C(x) \setminus \succnsim^C(y)\right| + \left|\succnsim^C(y) \setminus \succnsim^C(x)\right|$.

Pessimistic This is a variant of the previous strategy, which makes use of the min operator rather than the sum: $\mathit{fit}_{\text{p}}(x, y, C) = \min \left\{ \left|\succnsim^C(x) \setminus \succnsim^C(y)\right|, \left|\succnsim^C(y) \setminus \succnsim^C(x)\right| \right\}$.

Likelihood The fitness used by this elicitation strategy is proportional to the likelihood of a profile occurring where both x and y are possible winners as determined by the current approximation: with p being a probability distribution over $\mathcal{X}$, $\mathit{fit}_{\text{l}}(x, y, C) = \sum_{\{X \in \mathcal{X} \mid x, y \in F_{\mathcal{W}_C}(X)\}} p(X)$.

Random This elicitation strategy (used as a basis for comparison) selects randomly a pair (x, y) among incomparable pairs in $\succsim^C$, using a uniform distribution, using one instance of each class of permutation-indifferent rank-vectors.

The optimistic elicitation strategy tries to optimise the number of pairs that become comparable in $\succsim^{C_{k+1}}$ as compared to $\succsim^{C_k}$, thus after the answer is given. If the answer to the question (x, y) is that $x \succsim y$, then $\succsim^{C_{k+1}}$ gains at least one pair (x, z) for every z such that $[y \succnsim^{C_k} z] \wedge \neg[x \succnsim^{C_k} z]$, thus $z \in \{ \succnsim(y) \setminus \succnsim(x) \}$. (It also gains new pairs stemming from rank-vectors that dominate x, but the strategy does not consider those.) It implicitly makes the assumption that, when considering a pair x, y, the probability of an answer being $x \succsim y$ equals the probability that the answer is $y \succsim x$. The pessimistic strategy aims at optimising the number of pairs that become comparable in the case the answer is the least favorable.

The likelihood strategy considers that we do not only want to augment the number of pairs we know how to compare in $\succsim^{C_k}$, we also want to be able to compare specifically those pairs that often appear in voting instances and might be incorrectly considered as both winning in the current approximation. To estimate the probability distribution p of encountering a particular rank-profile, we use the *impartial culture assumption*, under which every voting instance is equally likely. It is well known that real elections do not conform to this assumption, but it is a useful simplification for our estimations.

Note that when implementing these strategies using the assumptions discussed here, it is only necessary to deal with one representation of each class of permutations of rank-vectors. This is so because all permutations of a rank-vector play the same role. Fix an arbitrary ordering $<_N$ on the voters N. Then define the set of increasing rank-vectors $I \subseteq [1, m]^N$ as the set of rank-vectors whose representation as a sequence of ranks following that arbitrary ordering is non-decreasing: $I = \{ x \in [1, m]^N \mid \forall i < j \in N : x_i \leq x_j \}$.

4.2 Experimental results

We now want to run an experiment in order to compare these elicitation strategies and see how "close" an approximation we can get depending on the number of questions asked. Recall that $F_{\succeq} \subseteq F_{\mathcal{W}_C}$. Thus, $F_{\mathcal{W}_C}(X)$ contains all the target winners (those given by $F_{\succeq}(X)$), but may also contain supplementary winners, denoted $S_X = F_{\mathcal{W}_C}(X) \setminus F_{\succeq}(X)$. To measure the quality of the approximation, we count how many supplementary winners the approximation gives, and we measure how bad these supplementary winners are compared to the target winners. We also make use of the impartial culture assumption in these definitions.

Ratio of number of winners The badness is $\frac{1}{|\mathcal{X}|} \sum_{X \in \mathcal{X}} \frac{\left|F_{\mathcal{W}_C}(X)\right|}{\left|F_{\succeq}(X)\right|}$.

Average WO error on a supplementary winner The second badness measure we use indicates how many equivalence classes below the target winners an average supplementary winner is. Define the weak order score $wo(x) \in \mathbb{N}$ of a rank-vector x as the number of equivalence classes that x dominates in the target weak order $\succeq$. If there are k equivalence classes in $\succeq$, $wo(x) \in [0, k-1]$. Define $WO(X)$, with X a non-empty set of rank-vectors, as the average weak order score over this set, thus $WO(X) = \frac{\sum_{x \in X} wo(x)}{|X|}$. Observe that for a voting instance $X \in \mathcal{X}$, the target winners all have the same wo score; denote that score by $t_X = WO(F_{\succsim}(X))$. The badness is $\frac{\sum_{X \in \mathcal{X}} \sum_{x \in S_X} t_X - wo(x)}{\sum_{X \in \mathcal{X}} |S_X|}$.

We approximate these badness measures by sampling 1000 randomly chosen voting instances. We also approximate the fitness given by the likelihood strategy by sampling randomly chosen voting instances.

As target rule, we used the Borda rule and randomly generated rules. To obtain a random weak order-based rule $F_{\succeq}$, we generate a weak order $\succeq$ on the set of increasing rank-vectors I, as follows. We start with a preorder $\succeq_0 = \blacktriangleright$, the Pareto dominance relation. At step k, we pick at random, using the uniform distribution over I, a pair of rank-vectors (x, y) that is incomparable in $\succeq_k$. We determine how this pair compares ($x \succsim y$; $y \succsim x$; or equivalence) with equiprobability, one chance in three for each possibility. We add this comparison to the preorder as well as the comparisons resulting from transitivity, obtaining $\succeq_{k+1}$. We iterate until all pairs are comparable.

Using our implementation, finding the next question to ask using any of these elicitation strategies only takes a few seconds on a normal desktop computer, for the problem sizes we tried.

Table 1 shows the performance of different elicitation strategies on some representative problem sizes. The two first columns indicate the problem size; the column "q" indicates the number of questions the elicitation strategy asked before computing the quality of the approximation; the column "fit" indicates which elicitation strategy that line is about (o is optimistic, r is random, p is pessimistic, l is likelihood with a sample size of 1000 and l+ is likelihood with a sample size of 10 000). The next two pairs of columns indicate the quality of the approximation according to the ratio of number of winners (nb w.) and according to the average WO error on a supplementary winner (wo su.). The first two columns of numbers relate to experiments eliciting the Borda rule; the second pair of columns of numbers indicate the quality of approximation reached when eliciting a randomly generated rule (as described above). Those results are averaged over ten runs. For each problem size, the first line gives an indication of the difficulty of the elicitation problem, as it indicates the badness of the robust rule for zero questions.

Table 1. Results of the experiment

				Borda		Random	
n	m	q	fit	nb w.	wo su.	nb w.	wo su.
10	4	0		1.5	2.4	1.7	27.3
		25	o	1.5	2.4	1.7	26.8
			r	1.5	2.1	1.6	23.4
			p	1.3	1.7	1.4	19.6
			l	1.1	2.1	1.2	19.6
			l+	1.1	2.1	1.2	21.0
		99	o	1.5	2.4	1.7	26.9
			r	1.3	1.7	1.4	17.0
			p	1.1	1.2	1.3	12.7
			l	1.0	0.8	1.0	11.2
			l+	1.0	0.2	1.0	15.2
6	6	0		1.9	3.1	2.2	52.4
		25	o	1.9	3.0	2.2	52.8
			r	1.8	2.7	2.0	44.4
			p	1.8	2.6	2.0	45.9
			l	1.5	2.1	1.6	33.4
			l+	1.3	2.0	1.7	38.2
		99	o	1.9	3.1	2.2	51.4
			r	1.7	2.2	1.8	32.0
			p	1.6	2.0	1.7	31.5
			l	1.1	1.4	1.3	22.5
			l+	1.0	1.5	1.3	28.4
4	10	0		2.3	3.8	2.6	69.9
		25	o	2.3	3.7	2.6	68.9
			r	2.3	3.4	2.4	61.6
			p	2.0	3.0	2.2	53.5
			l	2.0	2.9	2.0	51.7
			l+	1.8	2.8	2.1	56.7
		99	o	2.3	3.7	2.5	70.2
			r	2.0	2.8	2.1	50.3
			p	1.8	2.4	2.0	44.1
			l	1.4	1.9	1.6	40.1
			l+	1.3	2.2	1.6	43.4

Observe that the approximation using simply Pareto dominance and indifference to the permutation of rank-vectors ($q = 0$) already gives results that are surprisingly good, for the problem sizes considered here. For instance, for elections involving 10 voters and 4 alternatives, out of random elections, the approximation gives only a factor of 1.5 times the number of true winners. Furthermore, asking 25 questions using the l+ elicitation strategy already achieves significant improvement. Asking 99 questions suffices in most of these (small but realistic) cases to achieve near perfect approximation.

We see that the optimistic heuristic is surprisingly bad, as it performs worse than choosing questions at random. This can be understood as a consequence of its assumption that every answer is equally likely. Indeed, the pessimistic strategy performs much better that the optimistic one. The likelihood strategy is the clear winner among the elicitation strategies considered. Interestingly, its performance does not strongly benefit from increasing the sampling size to 10 000.

As a side note, it is also interesting to observe that the way used here to generate random rules yields rules that have many more equivalence classes than the Borda rule, as can be observed in the columns "wo su." after zero questions.

5 CONCLUSION

Viewing an election in terms of a set of rank-vectors instead of a set of linear orders raises many interesting theoretical and practical challenges. This perspective is suitable for elicitation by example, as they can be naturally expressed in terms of preferences over rank-vectors. However, finding good elicitation strategies is challenging. Theoretical research, and more experiments, should be conducted in order to direct the definition and evaluation of new elicitation strategies.

We assumed that the committee has a weak order over rank-vectors in mind and answers all questions accurately. This could be relaxed. First, the committee could have a preorder over rank-vectors in mind, thus it could be the case that they do not know, or do not care, about the relative positioning of some rank-vectors. Second, the committee could sometimes give wrong answers to the questions asked. Similarly, the committee could give different types of answers, such as saying that one rank-vector should not be ranked below another one but can be ranked above or be considered equally good. Supplementary theoretical results would have to be developed, in the spirit of the ones presented in Section 3, in order to determine whether the rules that can be defined using that type of constraint represent the same class as the class of robust preorder-based rules.

REFERENCES

[1] C. Blackorby, D. Donaldson, and J. A. Weymark, 'Social choice with interpersonal utility comparisons: A diagrammatic introduction', *International Economic Review*, **25**(2), 327–356, (1984).

[2] F. Brandt, V. Conitzer, and U. Endriss, 'Computational social choice', in *Multiagent Systems*, ed., G. Weiss, 213–283, MIT Press, (2013).

[3] O. Cailloux, P. Meyer, and V. Mousseau, 'Eliciting ELECTRE TRI category limits for a group of decision makers', *European Journal of Operational Research*, **223**(1), 133–140, (2012).

[4] L.C. Dias, V. Mousseau, J. Figueira, and J. N. Clímaco, 'An aggregation/disaggregation approach to obtain robust conclusions with ELECTRE TRI', *European Journal of Operational Research*, **138**(2), 332–348, (2002).

[5] J. Figueira, S. Greco, and M. Ehrgott, *Multiple Criteria Decision Analysis: State of the Art Surveys*, Springer Verlag, 2005.

[6] *Preference Learning*, eds., J. Fürnkranz and E. Hüllermeier, Springer, 2010.

[7] S. Greco, M. Kadziński, V. Mousseau, and R. Słowiński, 'Robust ordinal regression for multiple criteria group decision: UTAGMS-GROUP and UTADISGMS-GROUP', *Decision Support Systems*, **52**(3), 549–561, (2012).

[8] S. Greco, R. Słowiński, J. R. Figueira, and V. Mousseau, 'Robust ordinal regression', in *Trends in Multiple Criteria Decision Analysis*, eds., M. Ehrgott, J. R. Figueira, and S. Greco, 241–283, Springer US, (2010).

[9] A. Sen, 'On weights and measures: Informational constraints in social welfare analysis', *Econometrica*, **45**(7), 1539–1572, (1977).

[10] A. D. Taylor, *Social Choice and the Mathematics of Manipulation*, Cambridge University Press, 2005.