Recovering Single-Crossing Preferences From Approval Ballots

Andrei Constantinescu[✉][0009−0005−1708−9376] and Roger Wattenhofer[0000−0002−6339−3134]

ETH Zurich, Rämistrasse 101, 8092 Zurich, Switzerland
{aconstantine,wattenhofer}@ethz.ch

Abstract. An electorate with fully-ranked innate preferences casts approval votes over a finite set of alternatives. As a result, only partial information about the true preferences is revealed to the voting authorities. In an effort to understand the nature of the true preferences given only partial information, one might ask whether the unknown innate preferences could possibly be single-crossing. The existence of a polynomial time algorithm to determine this has been asked as an outstanding problem in the works of Elkind and Lackner [17]. We hereby give a polynomial time algorithm determining a single-crossing collection of fully-ranked preferences that could have induced the elicited approval ballots, or reporting the nonexistence thereof. Moreover, we consider the problem of identifying negative instances with a set of forbidden subballots, showing that any such characterization requires infinitely many forbidden configurations.

Keywords: Approval Voting · Single-crossing · Algorithms · Computational Complexity.

1 Introduction

One can express their opinion either in *precise* or in *simple* terms. The scientific world favors precision, but real-world decision, voting and election schemes usually adopt simplicity. Both ways have their disadvantages: simple voting is often limited in expressivity, requiring voters to make compromises ("I would really like to vote for candidate A, but candidate B has an actual chance of getting elected."). Precise voting, on the other hand, is too demanding of the voter ("Do I really have to rank all these candidates? I barely know them!"), and hence often not implemented. Indeed, the vast majority of the preference data repository PrefLib [30] consists of partially-elicited preferences. A modern solution striking a good balance between simplicity and expressive power stands in approval voting [26]. Here ballots ask voters to decide whether they approve or disapprove of each candidate; i.e., is the candidate good enough or not? Approval voting has been successfully applied to a number of settings, including among others multi-winner elections and participatory budgeting (see excellent survey in [25]).

Starting with the seminal work of Arrow [1], a long line of research has been dedicated to showing the impossibility of computing fair social outcomes.

However, these impossibility proofs typically hinge on pathological instances that are unlikely to occur in real-world elections, where political preferences can, for instance, often be explained using a left-right spectrum. Elections following such a spectrum are known as (one-dimensional) Euclidean. This model has received a number of generalizations enhancing its expressive power, including the single-peaked [6,1] and single-crossing [31,35] models, weakening the requirements of the societal axis in two distinct ways. An election is single-crossing if there is an ordering of the voters, called the single-crossing axis, such that, as we sweep through voters in order, preference between any two candidates changes at most once. We omit the definition of single-peaked preferences as it is not required for our work. While the two notions are incomparable, they share many of their attractive properties, including the existence of Condorcet winners and the polynomial-time computability of winners under election schemes for which this is hard in general, like the rules of Young, Dodgson and Kemeny [8] in the single-winner setting, and those of Chamberlin–Courant and Monroe in the multi-winner one [34,27]. The first three follow from the existence of Condorcet winners, while the latter two use dynamic programming [36,12,5,37,11]. When preferences are neither single-peaked nor single-crossing, one can still hope for the removal of a small fraction of voters and/or candidates to allow one of the models to apply.

The models discussed so far are defined for fully-elicited preferences. For approval ballots, their most natural extensions ask whether there are complete-information Euclidean, single-peaked, or single-crossing preferences refining the elicited approval preferences. If so, one calls the elicited preferences *possibly* Euclidean, single-peaked, or single-crossing (short PE, PSP, PSC). Elkind and Lackner [17] study these notions, among others, showing that PE and PSP surprisingly coincide, and PSC is more general (in fact the most general considered in their paper). However, one so far unanswered question is whether PSC collections of ballots can be recognized in polynomial time. In contrast, this is known to be possible for the other notions.

A polynomial-time algorithm for recognizing PSC and computing an associated societal axis has a few important implications. First of all, such an axis is required to apply the known polynomial-time algorithms for computing winning committees for the approval-based Chamberlin–Courant and Monroe rules under single-crossing preferences [36,12,11]. Moreover, Pierczyński and Skowron [33] show that core-stable committees always exist for approval elections under two definitions more restrictive than PSC (one of which being PE = PSP). It is likely that the same will be shown in the future for PSC, as no counterexample is known even for arbitrary approval elections,[1] and an algorithm computing the PSC axis will likely be a necessary component of an algorithm computing core-stable committees for PSC preferences. Finally, such an algorithm can be used by electoral authorities to study the political landscape emerging from an election.

---

[1] This is perhaps the main open question in the literature on proportional representation [25].

Our Contribution. We give a polynomial time algorithm that takes as input a collection of approval ballots and outputs a collection of fully-ranked ballots with the single-crossing property that could have produced the observed approval votes, reporting accordingly in case this is not possible. In the process, our approach also leads to a novel FPT algorithm for the non-betweenness problem [22]. Our problem has been posed as open by Elkind and Lackner [17].[2]

Moreover, in Appendix A, we show that, while single-crossing elections admit a characterization in terms of a set of two small forbidden subelections [9], such a characterization ceases to exist for PSC elections. We show this by exhibiting an infinite set of approval elections that are not PSC, but all their proper subelections are.

Related Work. Checking whether an election (with fully-elicited strictly ranked preferences) is single-crossing can be achieved in polynomial time [16,9], in fact even in near-linear time [18]. Checking whether an election can be embedded on a left-right spectrum (i.e., whether it is 1-Euclidean) can also be done in polynomial time [24,13,14]. In both cases, finding witnessing axes/embeddings is also polynomial. On the other hand, deciding whether an embedding exists in d dimensions becomes NP-hard as soon as $d \geq 2$ [32], and heuristic accounts support the difficulty of fitting such higher-dimensional models in practice [10].

Single-peaked elections can also be recognized in polynomial time [3,13,19,18]. A natural question related to our work is how difficult it is to recover 1-Euclidean preferences or single-peaked preferences from approval ballots. The two notions coincide (PE = PSP), and polynomial-time algorithms have been proposed [17,20] by reduction to the consecutive ones problem (which admits a linear-time algorithm, although using rather complex machinery [7]). In contrast, our algorithm for PSC proceeds directly, using non-betweenness only as a lens. An interesting model that encompasses both single-crossing and single-peaked elections is that of top-monotonic elections [2]. Such elections can be recognized in polynomial time [28] (also the paper whose techniques are closest to ours).

For the case of multi-valued approval, where voters can choose between more than two judgments when filling in the ballot, Fitzsimmons [21] provides a polynomial time recognition algorithm for possibly single-peaked preferences, working even for the case of an unbounded number of possible judgments; i.e., voters give a score between 1 and the number of candidates to each candidate (can also be thought as voters reporting rankings with ties). Their method proceeds by reducing to the consecutive ones problem. Getting similar results for the single-crossing case might seem desirable, but, as shown in [15], this is NP-hard for an unbounded number of possible judgments.

Working with approval ballots requires rather different social aggregators to those taking fully-ranked profiles as input. These aggregators have been recently

---

[2] Recently, we learned that a solution has in fact been proposed as early as 1979 in the context of the simple plant location problem, by Beresnev and Davydov [4]. This paper is only available in Russian, and Russian-speaking experts seem to believe that the paper is likely missing steps in the arguments. Beresnev and Davydov [4] is referenced in [23], but without details.

$$\mathcal{U} = \{1, 2, 3, 4, 5, 6\}$$
$$\mathcal{C} = \{(1, 2, 3), (2, 3, 4),$$
$$(2, 4, 3), (4, 2, 5),$$
$$(1, 5, 6)\}$$

(a) Input instance.   (b) Formula graph, excl. singletons.   (c) Colorful graph.
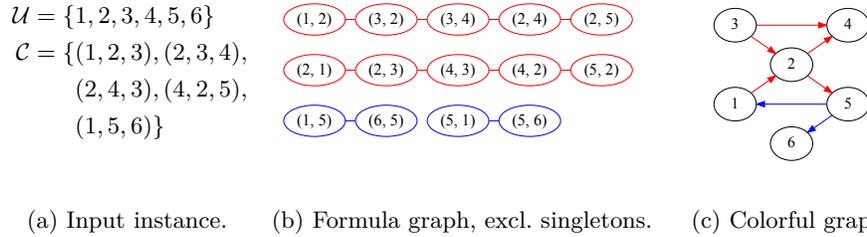
Fig. 1: Example of the construction of the formula and colorful graphs. Consider the instance of the non-betweenness problem given by $\mathcal{U}$ and $\mathcal{C}$ in Fig. 1a. The corresponding formula graph in Fig. 1b is constructed by adding edges $(a, b) - (c, b)$ and $(b, a) - (b, c)$ for each triple $(a, b, c) \in \mathcal{C}$. Isolated nodes have been omitted. Note that that no connected component of the formula graph contains both vertices (a, b) and (b, a) for any $a, b \in \mathcal{U}$. Subsequently, the connected components split into two pairs of complementary components: the two top long red chains and the two bottom short blue chains. The corresponding colorful graph in Fig. 1c is constructed by (arbitrarily) taking the top red component and the right blue component and introducing directed edges corresponding to the ordered pairs in the component. To solve the instance in Fig. 1a one needs to decide for each color whether or not to flip the direction of all edges of that color such that the resulting graph is acyclic. In our case, to avoid the cycle $1 \rightarrow 2 \rightarrow 5 \rightarrow 1$, the possible solutions are to either flip all blue edges or all red edges (but not both, as that leads to cycle $1 \rightarrow 5 \rightarrow 2 \rightarrow 1$ instead).

extensively studied in the context of multi-winner elections; see the comprehensive survey of both normative and computational results of Lackner and Skowron [25]. In an effort to translate the known models of voting to approval elections, Elkind and Lackner [17] consider twelve models catered to approval voting, the most general being the domain of possibly single-crossing elections. For all others, their paper shows polynomial-time recognition, while Terzopoulou et al. [38] show that the least restrictive of the others admit finite forbidden substructures characterizations. We resolve the open case of PSC and show that it admits no finite forbidden substructures characterization.

Finally, the problem of recognizing possibly single-crossing approval elections has a similar flavor to the recent work of Mandal et al. [29] on distortion induced by the preference elicitation mechanism, thus making it interesting and natural to study. In particular, instead of being interested in recovering voter utilities, here the objective is recovering the single-crossing axis of the innate preferences, mitigating the information lost when the electorate casts their votes through approval ballots; i.e., recovering utilities inducing a single-crossing profile.

Technical Overview. Our approach for proving the main result proceeds by first translating the input into an instance of the non-betweenness problem, which asks given a ground set $\mathcal{U}$ and a set $\mathcal{C}$ of triples (a, b, c) over $\mathcal{U}$ to compute a

linear (i.e., total, irreflexive, antisymmetric, transitive) ordering of the elements of $\mathcal{U}$ such that for none of the triples does b come between a and c. While non-betweenness is NP-hard in general [22], for the type of instances produced by our reduction the problem will turn out to be polynomial. To show this, we begin with an approach to non-betweenness based on boolean satisfiability. In particular, we introduce for every pair of distinct entries $a, b \in \mathcal{U}$ a boolean variable $x_{a,b}$ standing for whether a comes before b in the ordering. Antisymmetry can be captured by adding clauses requiring that $x_{a,b} = \overline{x_{b,a}}$, while for every triple $(a, b, c) \in \mathcal{C}$ one can show that non-betweenness is enforced by adding clauses ensuring that $x_{a,b} = x_{c,b}$. However, transitivity is substantially trickier to model without resorting to 3-literal clauses. To overcome this difficulty, taking a step back, we notice that except for transitivity all constraints are equalities. We take advantage of this fact by constructing the so-called *formula graph*, which has a vertex (a, b) for each variable $x_{a,b}$ in the boolean formula. Edges in the formula graph correspond to the equalities $x_{a,b} = x_{c,b}$ in the boolean formulation above, together with the inferred opposite equalities $x_{b,a} = x_{b,c}$. If some connected component contains both $x_{a,b}$ and $x_{b,a}$, then the instance is not satisfiable. Otherwise, we introduce yet another construct, the *colorful graph*, which is an edge-colored directed graph with vertex set G. One can show that at this point connected components in the formula graph come in complementary pairs $\{S, \overline{S}\}$, where $(a, b) \in S$ whenever $(b, a) \in \overline{S}$. For each such pair we create a new color, the edges of this color corresponding to the ordered pairs in set S. Some colors will only have one edge, which can be safely removed. Our two graph concepts are illustrated in Fig. 1. We then show that satisfying assignments of the boolean formula, this time including transitivity, correspond to acyclic orientations of the colorful graph where edges of each color are either all reversed, or all left in the original orientation. Deciding if such orientations of the colorful graph exist is still NP-hard, but exhausting over all possible orientations already shows that the non-betweenness problem is FPT with respect to the number of colors, a fact which might be of independent interest. At this point, we start studying the structure of the formula and colorful graphs induced by approval elections, showing the surprising result that, unless there is a monochromatic cycle, an acyclic orientation exists and can be computed in polynomial time, from which the election is possibly single-crossing. Proving this result constitutes the most demanding part of our work, and relies on a mixture of combinatorial and computer-aided techniques. Most notably, along the way we prove that if some orientation of the colorful graph is not acyclic, then the induced cycle is either monochromatic or of length three and not repeating colors. With further work, we then prove a strong structural result about colors taking part in three-colored triangles, allowing them to be consistently oriented to complete the proof.

2   Preliminaries

For a non-negative integer n, we write $[n]$ for the set $\{1, 2, \ldots, n\}$. Given a statement S, we write $[S]$ for the Iverson bracket: $[S] = 1$ if S holds, and 0 otherwise.

Given a finite set A, a *partial* order $\succ$ over A is an irreflexive, antisymmetric, transitive relation over A. Two values $a, b \in A$ are *comparable* under $\succ$ if either $a \succ b$ or $b \succ a$; otherwise, they are *incomparable*, in which case we write $a \approx_\succ b$. When any two distinct elements are comparable (i.e., the order is total), order $\succ$ is called a *linear order*. A partial order $\succ$ is called a *weak order* if $\approx_\succ$ forms an equivalence relation (i.e., it is transitive). One can think of weak orders as linear orders allowing for ties, corresponding to incomparable elements. For this reason, when $\succ$ is a weak order, whenever $a \approx_\succ b$ we say that $\succ$ is *indifferent* between a and b. We call *indifference classes* the equivalence classes induced by $a \approx_\succ b$. We call a weak order with at most k indifference classes a *k-weak order*. Note that an order $\succ$ is k-weak if and only if $\succ$ can be induced by a *scoring* function $s : A \to [k]$; i.e., $a \succ b$ if and only if $s(a) > s(b)$. Hence, 2-weak orders are precisely the approval ballots. Given partial orders $\succ$ and $\succ'$, we say that $\succ'$ *extends* $\succ$ if $a \succ' b$ whenever $a \succ b$.

In our setting, an assembly of voters $V = [n]$ expresses their preferences over a set of candidates (alternatives) $C = [m]$. The expressed preferences of voter $i \in V$ consist of a weak order $\succ_i$ over the set of candidates C. The collection of all voter preferences $\mathcal{P} = (\succ_i)_{i \in [n]}$ is known as the *preference profile*. When $\mathcal{P}$ consists solely of linear orders, we say that the electorate has *fully-ranked* preferences. When $\mathcal{P}$ consists solely of 2-weak orders, we say the electorate has *approval* preferences, in which case we will often see $\mathcal{P}$ as an $m \times n$ *approval matrix*, where $\mathcal{P}[c, v] = 1$ if voter v approves of candidate c, and 0 otherwise; i.e., column v contains the approval ballot cast by voter v. Given two profiles $\mathcal{P}$ and $\mathcal{P}'$, we say that $\mathcal{P}'$ is a *subprofile* of $\mathcal{P}$ if $\mathcal{P}'$ can be obtained from $\mathcal{P}$ by removing voters and/or candidates. For approval ballots, subprofiles correspond to removing rows and columns from $\mathcal{P}$. If $\mathcal{P}' \neq \mathcal{P}$ in the previous, then $\mathcal{P}$ is a *proper* subprofile of $\mathcal{P}$.

A profile of weak orders $\mathcal{P}$ is said to be *seemingly single-crossing* (SSC) with respect to a linear order $\lhd$ over the set of voters V, called the *axis*, if there are no three voters i, j, k such that $i \lhd j \lhd k$ and candidates a, b such that $a \succ_i b, b \succ_j a$ and $a \succ_k b$. If additionally profile $\mathcal{P}$ consists solely of linear orders, then under the same conditions $\mathcal{P}$ is *single-crossing* (SC) with respect to $\lhd$. A profile $\mathcal{P}$ of weak orders is said to be *possibly single-crossing* (PSC) with respect to a linear order $\lhd$ of the voters if for all voters $i \in V$ the preference order $\succ_i$ can be extended to a linear order $\succ'_i$ such that $(\succ'_i)_{i \in [n]}$ is single-crossing with respect to $\lhd$. A profile $\mathcal{P}$ is (seemingly/possibly) single-crossing if it is (seemingly/possibly) single-crossing with respect to some linear order $\lhd$ over V. For linear orders, the three notions coincide. More surprisingly, Elkind et al. [15] show that for weak orders the notions of SSC and PSC coincide.[3] Moreover,

----

[3] For general partial orders, they show that PSC implies SSC, but not conversely.

they show that given a linear order $\lhd$ witnessing SSC for profile $\mathcal{P}$, one can compute in polynomial time for each voter $i \in V$ a linear extension $\succ'_i$ of $\succ_i$ such that $(\succ'_i)_{i \in [n]}$ is single-crossing with respect to $\lhd$. In other words, knowing a seemingly single-crossing axis is enough to compute fully-ranked ballots that could have produced the cast ballots in polynomial time. For completeness, we give a streamlined version of their argument in the Appendix B. Therefore, our focus will be on computing, given a profile $\mathcal{P}$ of weak orders, an SSC axis, or deciding the nonexistence thereof. For general weak orders, Elkind et al. [15] show that deciding axis existence is NP-hard, but they leave it open for k-weak orders for $k \geq 2$, the case $k = 2$ corresponding with the approval voting case.

Given a finite ground set $\mathcal{U}$, a *non-betweenness* (NB) constraint over $\mathcal{U}$ is a triple of distinct elements $(i, j, k) \in \mathcal{U}^3$. A linear order $\lhd$ over $\mathcal{U}$ satisfies a collection $\mathcal{C}$ of NB constraints if for every $(i, j, k) \in \mathcal{C}$ it does not hold that $i \lhd j \lhd k$ or $k \lhd j \lhd i$; i.e., j is not between i and k in $\lhd$. The problem of deciding whether a linear order $\lhd$ satisfying $\mathcal{C}$ exists is NP-hard [22].[4] Checking for the SSC property reduces to verifying whether a set of NB constraints is satisfiable: given a profile $\mathcal{P}$, construct a set of NB constraints $\mathcal{C}_\mathcal{P}$ over V such that $(i, j, k) \in \mathcal{C}_\mathcal{P}$ iff there are two alternatives $a, b \in C$ such that $a \succ_i b, b \succ_j a$ and $a \succ_k b$. Then, we have the following, which is intuitively enabled by the fact that SSC is invariant with respect to reversing the axis:

Lemma 1. *$\mathcal{P}$ is SSC with respect to a linear order $\lhd$ iff $\mathcal{C}_\mathcal{P}$ is satisfied by $\lhd$.*

3   A Boolean Encoding of Non-Betweenness Constraints

In this section, we develop a general technique for deciding the satisfiability of an arbitrary set $\mathcal{C} \subseteq V^3$ of NB constraints over V. Naturally, our approach does not lead to a polynomial time algorithm for checking satisfiability, as the problem is NP-hard, but analysis in later sections will show that when applied to NB constraint sets $\mathcal{C}_\mathcal{P}$ originating from approval ballots, it leads to a polynomial algorithm for deciding satisfiability.

First, note that a NB constraint (i, j, k) is satisfied by a linear order $\lhd$ if and only if $[i \lhd j] = [k \lhd j]$. Moreover, order $\lhd$ is uniquely determined by the values $[i \lhd j]$ for $i \neq j$. These facts suggest a reformulation of the problem of satisfying NB constraints in terms of the values $[i \lhd j]$, which we do in the following. Given $\mathcal{C}$, we construct a boolean formula $\Phi_\mathcal{C}$ with a variable $x_{i,j}$ for every ordered pair of distinct elements $(i, j) \in V^2$. The clauses of $\Phi_\mathcal{C}$ comprise of the following three constraint sets, corresponding to antisymmetry, agreement with $\mathcal{C}$ and transitivity, respectively:

$\langle 1 \rangle$ For every pair of distinct elements $(i, j) \in V^2$ add the constraint $x_{i,j} = \overline{x_{j,i}}$.
$\langle 2 \rangle$ For every triple $(i, j, k) \in \mathcal{C}$, add the constraint $x_{i,j} = x_{k,j}$.

---

[4] This is used to prove the hardness of our problem for general weak orders, but the argument requires an unbounded number of indifference classes, so it does not work for bounded k.

⟨3⟩ For every triple of distinct $(i, j, k) \in V^3$ add the constraint $(x_{i,j} \wedge x_{j,k}) \rightarrow x_{i,k}$.

In the above we used = for the bi-implication operator $\leftrightarrow$, and we wrote $\overline{x}$ for logical negation, also commonly denoted by $\neg x$. Note that, under the presence of set ⟨1⟩, set ⟨2⟩ can be reformulated as below since $x_{j,i} = \overline{x_{i,j}} = \overline{x_{k,j}} = x_{j,k}$:

⟨2′⟩ For every triple $(i, j, k) \in \mathcal{C}$, add the constraints $x_{i,j} = x_{k,j}$ and $x_{j,i} = x_{j,k}$.

Henceforth, we will mostly use ⟨2′⟩, but ⟨2⟩ will still be more convenient in a few cases. The following lemma formally establishes the correspondence between linear orders satisfying $\mathcal{C}$ and satisfying assignments of $\Phi_{\mathcal{C}}$. The proof is essentially straightforward by following the definitions, so we leave it for the appendix.

Lemma 2. *Linear orders ⊲ satisfying the NB constraints in $\mathcal{C}$ correspond bijectively to satisfying assignments of $\Phi_{\mathcal{C}}$ by $x_{i,j} = [i \triangleleft j]$. Hence, $\mathcal{C}$ is satisfiable iff $\Phi_{\mathcal{C}}$ is satisfiable.*

Is this the end of the story? Not quite, because constraint set ⟨3⟩ encodes transitivity as $(x_{i,j} \wedge x_{j,k}) \rightarrow x_{i,k} \equiv (\overline{x_{i,j}} \vee \overline{x_{j,k}} \vee x_{i,k})$, which is a three-literal clause. However, constraint sets ⟨1⟩ and ⟨2⟩ only require two-literal clauses, so there is hope. The approach presented so far is similar in spirit to the approach of Magiera and Faliszewski [28] for recognizing top-monotonic elections. The main difficulty in their case also stems from enforcing transitivity, which similarly seems to require 3-literal clauses at a first glance. The essential observation that they make to progress is that whenever $(i, j, k) \in \mathcal{C}$, then the six entries in ⟨3⟩ enforcing transitivity for the unordered triple $\{i, j, k\}$ are no longer required, a fact which also holds true in our case. If for every unordered triple we would have at least one NB constraint involving its elements, then none of the transitivity constraints would be required, and the problem could simply be solved by any polynomial 2-CNF solver. This is the main idea of the approach taken in [28]. In our case, on the other hand, it turns out that it is surprisingly tricky to completely eliminate the transitivity constraints, so we will need more insight to deal with them gracefully.

4   Resolving Constraint Sets ⟨1⟩ and ⟨2⟩ Using Connected Components, The Formula Graph

Before tackling transitivity, we first turn our attention to constraint sets ⟨1⟩ and ⟨2⟩ in isolation, providing a precise characterization of the satisfying assignments of $\Phi_{\mathcal{C}}$ in the absence of constraint set ⟨3⟩. This characterization will be instrumental later on when reasoning about the transitivity constraints. In contrast, here Magiera and Faliszewski [28] directly employ a 2-CNF solver in order to tell whether sets ⟨1⟩ and ⟨2⟩ are satisfiable in isolation, but doing so only produces one satisfying assignment, rather than a compact representation of all of them, as will be the case for us. Our approach will hinge on the observation that

constraint set $\langle 2 \rangle$ only consists of equalities. In what follows, we will work with constraint set $\langle 2' \rangle$ instead of $\langle 2 \rangle$.

To begin, we define the *formula graph* of $\mathcal{C}$, denoted by $G_{\mathcal{C}}$. The vertex set $V(G_{\mathcal{C}})$ of $G_{\mathcal{C}}$ consists of all ordered pairs $(i, j) \in V^2$ with $i \neq j$; i.e., one vertex per variable in the formula. For a vertex (i, j) in $G_{\mathcal{C}}$ we define the *complementary* vertex $\overline{(i, j)}$ to be (j, i). Note that this is a syntactic notation, and is not to be confused with $\overline{x_{i,j}}$, which signifies the logical negation of a variable. For every constraint $x_u = x_v$ in $\langle 2' \rangle$, where u and v are pairs of distinct elements in V, we add an undirected edge $u - v$ to $G_{\mathcal{C}}$. Intuitively, each edge in $G_{\mathcal{C}}$ signifies an equality that needs to hold in the satisfying assignments of $\langle 1 \rangle$ and $\langle 2' \rangle$. We begin with two preliminary lemmas concerning the connected components of the formula graph.

Lemma 3. *An assignment $(x_u)_{u \in V(G_{\mathcal{C}})}$ satisfies constraint set $\langle 2' \rangle$ if and only if for each connected component S of $G_{\mathcal{C}}$ it holds that $x_u = x_v$ for all $u, v \in S$.*

Lemma 4. *Let S be a connected component in $G_{\mathcal{C}}$, then $\overline{S} := \{\overline{s} : s \in S\}$ is also a connected component in $G_{\mathcal{C}}$. Note moreover that $S = \overline{S}$ or $S \cap \overline{S} = \varnothing$.*

Hence, in light of Lemma 3, if for any connected component S it holds that $S = \overline{S}$, then $\langle 1 \rangle$ and $\langle 2' \rangle$ cannot be simultaneously satisfied. Otherwise, the connected components of $G_{\mathcal{C}}$ can be paired up into "complementary" pairs of distinct connected components $\{S, \overline{S}\}$. In particular, $V(G_{\mathcal{C}}) = \cup_{i=1}^{k} \left(S_i \cup \overline{S_i} \right)$ is a *complementary pairs partition* of the graph, where $(S_i)_{i \in [k]}$ are connected components of $G_{\mathcal{C}}$, one per complementary pair. Note that this partition is unique up to changing the roles of S_i and $\overline{S_i}$ for any i, so we will for simplicity refer to it as "the" complementary pairs partition. With this in mind, one can notice that solutions to sets $\langle 1 \rangle$ and $\langle 2' \rangle$ correspond to choosing for each complementary pair $\{S, \overline{S}\}$ one of the two components and setting to true variables corresponding to it and to false variables corresponding to the other component. This is formalized in the following proposition.

Proposition 5. *If $S = \overline{S}$ for any connected component of $G_{\mathcal{C}}$, then there are no assignments satisfying $\langle 1 \rangle$ and $\langle 2' \rangle$. Otherwise, let $V(G_{\mathcal{C}}) = \cup_{i=1}^{k} \left(S_i \cup \overline{S_i} \right)$ be the complementary pairs partition. Then, an assignment satisfies $\langle 1 \rangle$ and $\langle 2' \rangle$ iff it is obtained from a tuple $(s_1, \ldots, s_k) \in \{0, 1\}^k$ by setting all variables x_u to s_i for $u \in S_i$ and to $\overline{s_i}$ for $u \in \overline{S_i}$.*

Note that the condition $S \neq \overline{S}$ for all connected components S is equivalent to checking that for all $u \in V(G_{\mathcal{C}})$ vertices u and $\overline{u}$ are in different connected components. Hence, checking whether solutions to $\langle 1 \rangle$ and $\langle 2' \rangle$ exist is equivalent to checking for the latter, which the reader might recognize as the condition used for checking the satisfiability of general 2-CNFs.[5] However, for general 2-CNFs no characterization of the solution space is known since even counting solutions is #P-complete (while in our case we know that there are exactly 2^k solutions).

---

[5] If we replace our undirected graph by the directed implications graph and the word "connected" by "strongly-connected."

Henceforth, we will assume that $V(G_\mathcal{C}) = \cup_{i=1}^k \left(S_i \cup \overline{S_i} \right)$ is the complementary pairs partition (otherwise, just report that constraints in $\mathcal{C}$ cannot be satisfied). Then, Proposition 5 gives that the assignments satisfying $\langle 1 \rangle$ and $\langle 2' \rangle$ are precisely those for which a single value x_{S_i} is assigned to each connected component S_i and, similarly, for the complementary component $x_{\overline{S_i}} = \overline{x_{S_i}}$ holds. Note that this already shows that checking the satisfiability of a set of NB constraints is FPT with respect to the number $2k$ of connected components, since one could just try out all 2^k satisfying assignments for $\langle 1 \rangle$ and $\langle 2' \rangle$ and check for transitivity in each case.

5 Transitivity by Acyclic Orientations, the Colorful Graph

In this section, we provide an equivalent view of the satisfying assignments of sets $\langle 1 \rangle$ and $\langle 2' \rangle$ as edge-orientations of a certain edge-colored graph. In particular, all edges have a base orientation, and the orientations satisfying $\langle 1 \rangle$ and $\langle 2' \rangle$ correspond to selecting for each color whether to leave all edges as they are, or reverse the direction of all edges of that color. Edge orientations additionally satisfying constraint set $\langle 3 \rangle$ will be those inducing a directed acyclic graph.

To begin, recall that $V(G_\mathcal{C}) = \cup_{i=1}^k \left(S_i \cup \overline{S_i} \right)$ is the complementary pairs partition. Consider some connected component S_i. This component consists of a set of ordered pairs $S_i = \{(i_1, j_1), (i_2, j_2), \ldots, (i_\ell, j_\ell)\}$. To have a satisfying assignment of $\langle 1 \rangle$ and $\langle 2' \rangle$, it has to be the case that either $x_{i_1,j_1} = \ldots = x_{i_\ell,j_\ell} = 1$, or $x_{i_1,j_1} = \ldots = x_{i_\ell,j_\ell} = 0$. Both choices lead to satisfying assignments, independently of the choices made for the other components. By Lemma 2, the previous gives that either $[i_1 \triangleleft j_1] = \ldots = [i_\ell \triangleleft j_\ell] = 1$, or $[i_1 \triangleleft j_1] = \ldots = [i_\ell \triangleleft j_\ell] = 0$, and the choice is independent across S_i's.

Any linear order $\triangleleft$ over V can be seen as a directed tournament graph with vertex set V, where we draw a directed edge $i \to j$ if $i \triangleleft j$ and a directed edge $j \to i$ if $j \triangleleft i$. Consequently, the constraint $[i_1 \triangleleft j_1] = \ldots = [i_\ell \triangleleft j_\ell]$ induced by some connected component S_i of the formula graph is equivalent in the tournament of $\triangleleft$ to the fact that we either have edges $i_1 \to j_1, \ldots, i_\ell \to j_\ell$, or edges $j_1 \to i_1, \ldots, j_\ell \to i_\ell$. In other words, every set S_i is a directed set of edges that in the tournament of $\triangleleft$ either has to have the given orientation, or exactly the reverse orientation. We call these two options for S_i its two possible *orientations*. We call a choice between the two options for each of the sets $S_1, \ldots, S_k$ an *orientation* of all edge sets. The following shows that out of those orientations, the ones that additionally satisfy set $\langle 3 \rangle$ are those where the resulting directed tournament graph is acyclic. For brevity, we call such orientations *acyclic*.

Proposition 6. *A linear order $\triangleleft$ over V satisfies the NB constraints in $\mathcal{C}$ if and only if it is given by an acyclic orientation of the edge sets $S_1, \ldots, S_k$.*

Some edge sets (connected components in the formula graph) S_i will be singletons; i.e., $|S_i| = 1$. Such sets are not relevant for deciding whether an acyclic orientation exists, because singletons can just be oriented freely after

everything else has been oriented. More formally, without loss of generality, assume that edge sets $S_1, \ldots, S_\ell$ are non-singletons; i.e., $|S_i| > 1$ for $i \in [\ell]$; and that edge sets $S_{\ell+1}, \ldots, S_k$ are singletons; i.e., $|S_i| = 1$ for $\ell < i \leq k$. Then, we have the following:

Lemma 7. *An acyclic orientation of edge sets $S_1, \ldots, S_k$ exists if and only if an acyclic orientation of edge sets $S_1, \ldots, S_\ell$ exists.*

Armed as such, we now introduce the *colorful graph* corresponding to the NB constraint set $\mathcal{C}$. The colorful graph is a directed graph with vertex set V and edges colored in ℓ colors, identified by the numbers in $[\ell]$. For each $i \in [\ell]$ the edges colored with color i are those in edge set S_i.

Note how for any $u, v \in V$ with $u \neq v$ at most one of the edges $u \to v$ and $v \to u$ can appear in $S_1, \ldots, S_\ell$. Moreover, an edge cannot occur in more than one of $S_1, \ldots, S_\ell$. As a result, the colorful graph has neither parallel nor anti-parallel edges, and no self-loops. When either edge $a \to b$ or $b \to a$ exists in the colorful graph, we say that $a \text{---} b$ exists in the colorful graph. In this case, we write $c(a, b) = c(b, a)$ to denote the color of the respective oriented edge. When the "undirected" edge $a \text{---} b$ exists in the graph, and the corresponding directed edge is $a \to b$, we say that $a \text{---} b$ is *oriented* from a to b. Finally, when edge $a \to b$ $(a \text{---} b)$ bears color c, we write $a \xrightarrow{c} b$ $(a \overset{c}{\text{---}} b)$. Note that the colorful graph is only defined for NB constraint sets $\mathcal{C}$ for which the complementary pairs partition exists.

We can now reformulate our task in terms of the colorful graph. In particular, we want to check whether the edges of the colorful graph can be reoriented such that the resulting graph is acyclic, under the constraint that for each color we choose between flipping (reversing) the direction of all edges of that color or doing nothing. When flipping the direction of edges of a certain color, we say that we "flipped" that color.

6   The Case of Approval Ballots

We now switch our attention to colorful graphs induced by approval preference profiles $\mathcal{P}$; i.e., corresponding to the set of NB constraints $\mathcal{C}_{\mathcal{P}}$. The structure of these graphs is still combinatorially rich, making reasoning about acyclic orientations challenging, but at the same time significantly less general than for arbitrary sets of NB constraints. Examples of such induced colorful graphs can be found in Fig. 2. For brevity, we henceforth assume that we have already checked in polynomial time whether there are any monochromatic cycles and rejected the input if so. In this section we will prove that if the colorful graph has passed this test, then an acyclic orientation is guaranteed to exist and can be determined in polynomial time. To show this, we prove a number of structural results about inducible colorful graphs, centering around the concept of *biclique colors*, which are colors whose corresponding edges can be described by a cartesian product $A \times B$ for two disjoint sets $A, B \subseteq V$. A color is *star biclique* if it is a biclique color with either $|A| = 1$ or $|B| = 1$. The two concepts are

1 2 3 4 5
0 1 0 1 0
0 0 0 0 1
0 0 1 1 0
1 1 0 1 0
1 0 0 0 0

1 2 3 4 5 6 7
0 0 0 0 0 1 0
1 0 0 1 0 0 0
0 1 0 0 0 0 0
0 0 1 0 1 0 0
1 0 0 0 0 0 0
0 0 1 0 0 1 0
1 0 0 0 0 0 1

(a) Profile $\mathcal{P}_1$.   (b) Col. graph of $\mathcal{P}_1$.   (c) Profile $\mathcal{P}_2$.   (d) Colorful graph of $\mathcal{P}_2$.
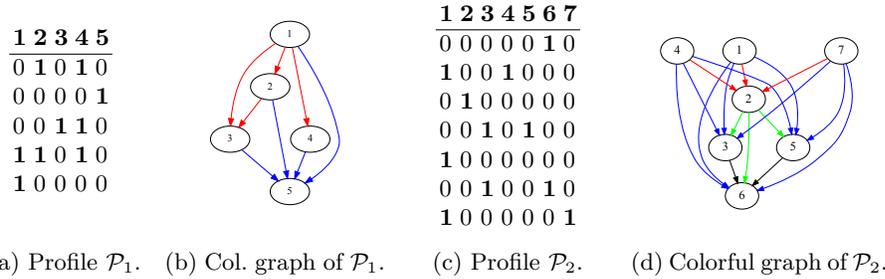
Fig. 2: Two example preference profiles $\mathcal{P}_1, \mathcal{P}_2$, together with the colorful graphs they induce. In Fig. 2d all colors are biclique colors. Namely, the red edges are given by $\{1, 4, 7\} \times \{2\}$, the green edges by $\{2\} \times \{3, 5, 6\}$, the black edges by $\{3, 5\} \times \{6\}$ and the blue edges by $\{1, 4, 7\} \times \{3, 5, 6\}$. Out of these, red, green and black are star biclique. In general, not all colors will be biclique, e.g., in Fig. 2b, only the blue color is biclique.

illustrated in Fig. 2. Not all colors in the inducible colorful graphs are biclique, as shown in Fig. 2b. However, our result concerning them will be that whenever edges $a — b — c — a$ exist in the colorful graph and bear different colors, these three colors are biclique, and, moreover, the three cartesian products describing them are of the form $A \times B$, $B \times C$ and $C \times A$. In Fig. 2d this can be observed for the triangle $1 — 2 — 3 — 1$. Using this result, together with an analogue for star biclique colors, we then prove that if some orientation of the colorful graph has a cycle, then the smallest such cycle is of length three, showing that, intuitively, triangles are all one needs to worry about. To complete the proof of the main result, we then show that non-monochromatic triangles neccesarily consist of three different colors, implying that we can ignore all colors taking part in no three-colored triangles and focus on the rest. Afterward, a direct argument reusing our theorem about three-colored triangles can be used to construct an acyclic orientation of the colorful graph in polynomial time, proving the main claim of the paper.

6.1   Biclique Colors

From now on, all colorful graphs we consider are induced by approval preferences, unless stated otherwise. In this section, the goal is the prove that whenever $a — b — c — a$ is a three-colored triangle in the colorful graph, then these three colors are biclique and their edges are given by $A \times B$, $B \times C$ and $C \times A$ for some disjoint sets $A, B, C \subseteq V$ where $a \in A, b \in B, c \in C$. Following up, we then also give an instrumental analog for star biclique colors, namely that if $a — b — c$ are different-colored edges in the colorful graph, and edge $a — c$ does not exist in the colorful graph, then there is a set $B \subseteq V \setminus \{a, c\}$ with $b \in B$ such that the two colors consist of edges $\{a\} \times B$ and $B \times \{c\}$ respectively. This can be intuitively thought of as the case $|A| = |C| = 1$ of the result for three-colored triangles.
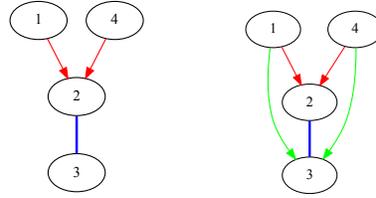
(a) Assumed colors.   (b) Implied colors.

Fig. 3: Illustration of Lemma 8. Fig. 3a depicts the assumptions: edge $(1,2) — (4,2)$ exists in the formula graph (red) and edge $(2,3)$ exists in the colorful graph and bears a different color (blue). Additionally, edge $(1,3)$ is neither red nor blue. Fig. 3b shows the edge colors implied by the setup, where green is a color different from red and blue.

Henceforth, unless stated otherwise, the preference profiles $\mathcal{P}$ that we consider consist of approval ballots, and are represented through approval matrices.

Our high-level argument will hinge on a number of lower-level results concerning the effect that small local structures in the approval matrix have on the formula and colorful graphs. Below is the first such result that we will need. Its proof is a combination of direct reasoning with case analysis over 4×6 partially filled-in approval matrices. While we have considerably simplified the proof over naive case exhaustion, we found the details inessential, so left them for the appendix.

Lemma 8. *Consider four voters, say* $1, 2, 3, 4$*, such that the formula graph contains the edge* $(1,2) — (4,2)$ *and that edge* $(2,3)$ *exists in the colorful graph and has a different color than* $(1,2)$*. If edge* $(1,3)$ *is either not present in the colorful graph, or it has a different color than* $(1,2)$ *and* $(2,3)$*, then edge* $(1,3) — (4,3)$ *is guaranteed to be in the formula graph. The scenario is illustrated in Fig. 3.*

It is crucial to notice at this point that Lemma 8, as well as all other results of a similar flavor that we will prove, continue to hold if some of the involved voters coincide. This can be seen by imagining cloning the recurring voters and applying the result with distinct voters. Armed with the previous lower-level result, we are now ready to prove the main result of the section, concerning the biclique structure of colors involved in three-colored triangles.

Theorem 9. *Consider three voters, say* $1, 2$ *and* 3*, such that edges* $(1,2)$*,* $(2,3)$ *and* $(3,1)$ *exist in the colorful graph and have different colors, then there exist disjoint sets of voters* S_1*,* S_2 *and* S_3 *such that* $1 \in S_1$*,* $2 \in S_2$ *and* $3 \in S_3$ *and the connected components of* $(1,2)$*,* $(2,3)$ *and* $(3,1)$ *in the formula graph are* $S_1 \times S_2$*,* $S_2 \times S_3$ *and* $S_3 \times S_1$ *respectively.*
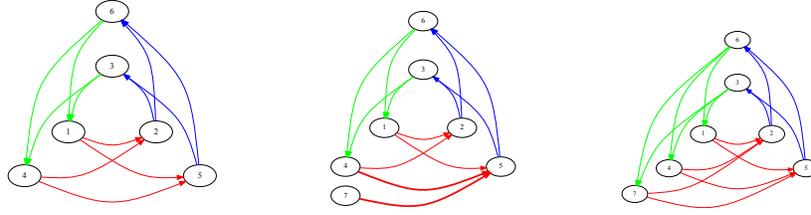
(a) Col. graph for (S_1, S_2, S_3). (b) Assume $(4, 5, 7) \in \mathcal{C}_\mathcal{P}$. (c) Contradicts maximality.

Fig. 4: Illustration of the proof of Theorem 9. We consider an inclusion-maximal triple (S_1, S_2, S_3) such that $1 \in S_1, 2 \in S_2, 3 \in S_3$ and $S_1 \times S_2, S_2 \times S_3$ and $S_3 \times S_1$ are connected in the formula graph, in this case $S_1 = \{1, 4\}, S_2 = \{2, 5\}$ and $S_3 = \{3, 6\}$ (Fig. 4a). Then, we assume for a contradiction that these connected sets are not connected components, in this case because of NB constraint $(4, 5, 7)$ (Fig. 4b). Finally, this implies that $(S_1 \cup \{7\}, S_2, S_3)$ also has the property, contradicting maximality (Fig. 4c).

Proof. Our proof strategy is summarized in Fig. 4. Consider an inclusion-maximal triple of disjoint sets (S_1, S_2, S_3) such that $1 \in S_1$, $2 \in S_2$, $3 \in S_3$ and the sub-graphs of the formula graph induced by $S_1 \times S_2$, $S_2 \times S_3$ and $S_3 \times S_1$ are connected. By inclusion-maximal we mean that for every $v \notin (S_1 \cup S_2 \cup S_3)$ the triples $(S_1 \cup \{v\}, S_2, S_3)$, $(S_1, S_2 \cup \{v\}, S_3)$ and $(S_1, S_2, S_3 \cup \{v\})$ do not satisfy the property. Note that the connectivity assumption implies that in the colorful graph edges in $S_1 \times S_2$ have the color of $(1, 2)$, edges in $S_2 \times S_3$ have the color of $(2, 3)$ and edges in $S_3 \times S_1$ have the color of $(3, 1)$. Denote by C_{12}, C_{23} and C_{31} the connected components of $(1, 2)$, $(2, 3)$ and $(3, 1)$ in the formula graph. We will now show that $C_{12} = S_1 \times S_2$, $C_{23} = S_2 \times S_3$ and $C_{31} = S_3 \times S_1$. Assume for a contradiction that this was not the case, then, without loss of generality, $C_{12} \neq S_1 \times S_2$. Together with the above, this means that $S_1 \times S_2 \subsetneq C_{12}$. Because C_{12} induces a connected subgraph in the formula graph, and $\varnothing \neq S_1 \times S_2 \subsetneq C_{12}$, it follows that there is an edge in the formula graph crossing the cut $(S_1 \times S_2, C_{12} \setminus (S_1 \times S_2))$. Without loss of generality, this edge is of the form $(v_1, v_2) — (x, v_2)$, where $v_1 \in S_1, v_2 \in S_2$ and $x \notin S_1$.

Now, we show that for every $v_3 \in S_3$ there is an edge $(v_1, v_3) — (x, v_3)$ in the formula graph. Note that this means that all edges of the form (x, v_3) for $v_3 \in S_3$ have the color of $(1, 3)$. Consider an arbitrary $v_3 \in S_3$, and note that we can instantiate Lemma 8 with $1 \mapsto v_1, 2 \mapsto v_2, 3 \mapsto v_3$ and $4 \mapsto x$ because $(v_1, v_2) — (x, v_2)$ is in the formula graph and $(v_1, v_2), (v_2, v_3)$ and (v_3, v_1) bear the colors of $(1, 2), (2, 3)$ and $(3, 1)$, which are different. The instantiation gives us that there is an edge $(v_1, v_3) — (x, v_3)$ in the formula graph, as required.

From the above, we know that the edge $(v_1, 3) — (x, 3)$ exists in the formula graph. Similarly, we now show that for every $v_2' \in S_2$ there is an edge $(v_1, v_2') — (x, v_2')$ in the formula graph. This is done analogously, by invoking

Lemma 8 with $1 \mapsto v_1, 2 \mapsto 3, 3 \mapsto v_2'$ and $4 \mapsto x$, which is sound because $(v_1, 3) - (x, 3)$ is in the formula graph and $(v_1, 3)$, $(3, v_2')$ and (v_2', v_1) bear the colors of $(1, 3)$, $(3, 2)$ and $(2, 1)$, which are different.

At this point, we have shown that $(S_1 \cup \{x\}) \times S_2$ and $S_3 \times (S_1 \cup \{x\})$ both induce connected subgraphs in the formula graph; in particular, all edges from x to S_2 have the color of $(1, 2)$ and all edges from x to S_3 have the color of $(1, 3)$. We will now additionally show that $x \notin (S_2 \cup S_3)$, from which the triple $(S_1 \cup \{x\}, S_2, S_3)$ also satisfies the property we started with, contradicting the hypothesis that (S_1, S_2, S_3) was inclusion-maximal. First, if $x \in S_3$, then $(1, 2)$ and $(2, 3)$ immediately have the same color, which cannot be the case. Otherwise, if $x \in S_2$, then from our argument $(x, 3)$ has the same color as $(1, 3)$, while the assumptions tell us that it has the same color as $(2, 3)$, meaning that $(1, 3)$ and $(2, 3)$ share the same color, which is again not possible.

The following result can be seen as a star biclique analogue of Theorem 9. Intuitively, one can think of this as the orthogonal case where the edge $1 - 3$ does not exist in the colorful graph and $S_1 = \{1\}, S_3 = \{3\}$. While the two proofs are very similar in spirit, we could not find a transparent way of unifying the two arguments, partly because of less symmetry in the scenario. In particular, this lack of symmetry will require a second low-level result in the spirit of Lemma 8, stated and proven in the appendix. See the appendix for the relevant proofs.

Theorem 10. *Consider three voters, say $1, 2$ and 3, such that edges $(1, 2)$ and $(2, 3)$ exist in the colorful graph and have different colors, while edge $(1, 3)$ does not exist in the colorful graph. Then, there exists a set of voters S such that $2 \in S$ and $1, 3 \notin S$ with the property that the connected components of $(1, 2)$ and $(2, 3)$ in the formula graph are $\{1\} \times S$ and $S \times \{3\}$ respectively.*

6.2   Only Three-Colored Triangles Matter, the Main Result

In this section we prove that if some orientation of the colorful graph has a cycle, then the smallest such cycle is a triangle. Then, we show that non-monochromatic triangles neccesarily consist of three different colors. Finally, we prove that any colorful graph with no monochromatic cycles has an acyclic orientation, hinging on Theorem 9.

Lemma 11. *An orientation of the colors in the colorful graph is acyclic if and only if it induces no directed cycles of length three.*

Proof. The "only if" direction is straightforward. In what follows we prove the "if" direction. Consider an arbitrary orientation of the colors in the colorful graph. Cycles of length one and two are not possible because the colorful graph lacks self-loops and parallel edges, so assume for a contradiction that the length of the shortest cycle induced by our orientation is at least four. Let such a shortest cycle be $C = v_1 \to v_2 \to \ldots \to v_k \to v_1$, where $k \geq 4$. Observe that the minimum-length assumption implies that C is an induced cycle, because any extra edge in the subgraph of the colorful graph induced by $\{v_1, \ldots, v_k\}$ would

imply a shorter cycle. If all edges in C are of the same color, this would contradict the fact that all colors are acyclic when seen individually, so C consists of at least two distinct colors, so at least two distinct colors occur along C.

We now prove that all colors c occurring along C are star biclique colors. To do so, for any such c note that there exists an index $1 \leq i \leq k$ such that $v_i \xrightarrow{c} v_{i+1}$ and $v_{i+1} \xrightarrow{c'} v_{i+2}$, where $c' \neq c$ is some color other than c and index arithmetic is performed modulo k such that $v_{k+1} = v_1$, etc. Because C is an induced cycle, it follows that edge $v_i - v_{i+2}$ does not appear in the colorful graph. Since additionally $c \neq c'$, instantiating Theorem 10 with $1 \mapsto v_k, 2 \mapsto v_{k+1}, 3 \mapsto v_{k+2}$ gives that c and c' are star biclique colors.

Since all colors occurring along C are star biclique, every color can only occur in C at most once, as any second occurrence would imply that the common node of the edges in a star is reused, meaning C is not a simple cycle, and hence not minimal. Therefore, let $c_1, c_2, \ldots, c_k$ be the pairwise distinct colors of the edges in C, in order.

Now, applying Theorem 10 with $1 \mapsto v_1, 2 \mapsto v_2, 3 \mapsto v_3$ gives that the connected component of (v_1, v_2) in the formula graph is $\{v_1\} \times S$ and that of (v_2, v_3) is $S \times \{v_3\}$, for some S such that $v_1, v_3 \notin S$. Similarly, applying Theorem 10 with $1 \mapsto v_2, 2 \mapsto v_3, 3 \mapsto v_4$ gives that the connected component of (v_2, v_3) in the formula graph is $\{v_2\} \times S'$ and that of (v_3, v_4) is $S' \times \{v_4\}$, for some S' such that $v_2, v_4 \notin S'$. As a result, the connected component of (v_2, v_3) is at the same time $S \times \{v_3\}$ and $\{v_2\} \times S'$. Setting the two equal yields $S \times \{v_3\} = \{v_2\} \times S'$, from which the connected component is actually exactly $\{(v_2, v_3)\}$, contradicting the fact that (v_2, v_3) occurs in the colorful graph.

Next, we prove that only three-colored triangles need to be considered; e.g., the case in Fig. 1c cannot occur. For this we will need the following stronger form of Lemma 8:

Lemma 12 (Strengthened Lemma 8). *Consider four voters, say $1, 2, 3, 4$, such that the formula graph contains the edge $(1, 2) - (4, 2)$ and that edge $(2, 3)$ exists in the colorful graph. If the formula graph does not contain edges $(3, 2) - (1, 2)$ and $(3, 2) - (4, 2)$, then the formula graph is guaranteed to contain the edge $(1, 3) - (4, 3)$.*

The proof closely follows that of Lemma 8, but the weaker assumptions make it significantly more difficult to proceed as before in a principled manner. Instead, we resort to a computer program to perform the case analysis (see appendix for details).

Lemma 13. *Consider three voters, say $1, 2, 3$ such that edges $1 - 2$ and $2 - 3$ are the same color and oriented $1 \rightarrow 2$ and $2 \rightarrow 3$ in the colorful graph. If edge $1 - 3$ appears in the colorful graph, then it is of the same color as $1 - 2 - 3$.*

Proof. Without loss of generality, assume that $1 - 3$ is oriented $1 \rightarrow 3$ in the colorful graph. Assume for a contradiction that the color of $1 - 2 - 3$ is c and the color of $1 - 3$ is $c' \neq c$. Because $1 \xrightarrow{c} 2$ and $2 \xrightarrow{c} 3$ are the same

color, it follows that there is a simple path in the formula graph $(1,2) = (a_1, b_1) - (a_2, b_2) - \ldots - (a_k, b_k) = (2,3)$. We will prove by induction on $1 \leq \ell \leq k$ that $a_\ell \xrightarrow{c'} 3$ and $b_\ell \xrightarrow{c} 3$, which is a contradiction since $b_k = 3$ and there are no self-loops in the colorful graph. The base case $\ell = 1$ is true by assumption. Assume that $a_\ell \xrightarrow{c'} 3$ and $b_\ell \xrightarrow{c} 3$ for some $\ell < k$, we then want to prove that $a_{\ell+1} \xrightarrow{c'} 3$ and $b_{\ell+1} \xrightarrow{c} 3$. By construction of the formula graph, distinguish two cases:

If $a_{\ell+1} = a_\ell$, then we instantiate Lemma 12 with $1 \mapsto b_\ell, 2 \mapsto a_\ell, 3 \mapsto 3, 4 \mapsto b_{\ell+1}$. This is sound because edge $(b_\ell, a_\ell) - (b_{\ell+1}, a_\ell)$ is in the formula graph and edges $(3, a_\ell) - (b_\ell, a_\ell)$ and $(3, a_\ell) - (b_{\ell+1}, a_\ell)$ are not, since $3 - a_\ell$ is color c', while the other two are color c by assumption. The instantiation gives that edge $(b_\ell, 3) - (b_{\ell+1}, 3)$ is in the formula graph, from which $b_{\ell+1} \xrightarrow{c} 3$.

If $b_{\ell+1} = b_\ell$, then we instantiate Lemma 12 with $1 \mapsto a_\ell, 2 \mapsto b_\ell, 3 \mapsto 3, 4 \mapsto a_{\ell+1}$. This is sound because edge $(a_\ell, b_\ell) - (a_{\ell+1}, b_\ell)$ is in the formula graph and edges $(3, b_\ell) - (a_\ell, b_\ell)$ and $(3, b_\ell) - (a_{\ell+1}, b_\ell)$ are not, since the colorful graph has edges $a_\ell \xrightarrow{c} b_\ell$, $a_{\ell+1} \xrightarrow{c} b_\ell$ and $b_\ell \xrightarrow{c} 3$, which together with any of $(3, b_\ell) - (a_\ell, b_\ell)$ or $(3, b_\ell) - (a_{\ell+1}, b_\ell)$ in the formula graph would lead to a path between $(b_\ell, 3)$ and $(3, b_\ell)$ in the formula graph, which cannot be the case. The instantiation gives us that the edge $(a_\ell, 3) - (a_{\ell+1}, 3)$ exists in the formula graph, from which $a_{\ell+1} \xrightarrow{c} 3$.

Corollary 14. *An orientation of the colors in the colorful graph is acyclic if and only if it induces no directed cycles of length three consisting of three different colors.*

Theorem 15. *An acyclic orientation of the colorful graph exists and can be computed in poly-time.*

Proof. By Corollary 14, only colors participating in three-colored triangles have to be considered (all other colors we can orient arbitrarily). Let $C = \{c_1, \ldots, c_k\}$ be the set of such colors. For each color $c_i \in C$, applying Theorem 9 to an arbitrary three-colored triangle involving c_i gives that c_i is a biclique color, so the set of edges colored with color c_i in the colored graph is $A_i \times B_i$ for some disjoint sets of vertices A_i, B_i. For a finite set of numbers S, we write $\min S$ for the minimum number in S. We now construct an orientation of the colors in C as follows: for color $c_i \in C$ we orient the edges of color c_i to go $A_i \rightarrow B_i$ if $\min A_i < \min B_i$ and $B_i \rightarrow A_i$ otherwise. Note that $\min A_i \neq \min B_i$ as $A_i \cap B_i = \varnothing$. We now prove that this orientation is acyclic to complete the proof. Assume for a contradiction that the orientation induced a cycle $a \rightarrow b \rightarrow c \rightarrow a$. Then, by Corollary 14, the three edges involved in the cycle bear different colors in the colorful graph, so, by Theorem 9, there exist three disjoint vertex sets A, B, C such that $a \in A, b \in B, c \in C$ and the connected components of $a \rightarrow b$, $b \rightarrow c$ and $c \rightarrow a$ in the formula graph are given by $A \times B$, $B \times C$ and $C \times A$. However, by construction of the orientation, this would mean that $\min A < \min B < \min C < \min A$, a contradiction.

Armed as such, we can now prove the main result of our paper.

Theorem 16. *There is a polynomial time algorithm taking as input a collection of approval ballots that outputs a profile of single-crossing preferences that could have lead to the given approval votes. If this is not possible, the algorithm reports accordingly.*

Proof. Start with the approval ballots and build the formula graph. If there are pairs of complementary nodes in the same connected component, report impossibility. Otherwise, build the colorful graph. If a monochromatic cycle exists, report impossibility. Otherwise, use Theorem 15 to compute an acyclic orientation, and extend it to a linear order ◁. Finally, use the approach of Elkind et al. [15], explained in Appendix B, to compute a single-crossing profile with respect to the known axis ◁.

7   Conclusions and Future Work

We gave a polynomial time algorithm computing a collection of fully-ranked ballots that could have been induced by the given approval ballots. Furthermore, we showed that there is no finite substructures characterization of possibly single-crossing elections (deferred to Appendix A). To prove our main result, we developed a new algorithm for the non-betweenness problem, showing that it is FPT with respect to the number of colors in a certain colorful graph. We expect this FPT algorithm to lead to polynomial algorithms for other ordering problems reducing to the non-betweenness problem by combinatorial analysis of the induced colorful graphs. We note that betweenness reduces to non-betweenness since the constraint "b has to be between a and c" can be modeled as the conjunction of two non-betweenness constraints, namely: "a should not be between b and c" and "c should not be between a and b", so our framework can also be used for betweenness.

As steps for future research, the most natural next question to ask is the complexity of recognizing possibly single-crossing for multi-valued approval, the case of three indifference classes being a natural first candidate to consider. We have empirically found that our proof of correctness no longer applies in this case, since for instance Lemma 8 fails to hold for 3-valued approval. However, we could not find instances where the algorithm as a whole fails the recognition task. We leave it open to settle this question. Another promising avenue for further investigation stands in the case when our algorithm reports that voters' innate preferences are not single-crossing. In this case, one might hope that they are at least somewhat close to being single-crossing, so it would be interesting to investigate how to recover preferences that are nearly single-crossing from approval ballots (see [18, Section 4.8] for a discussion of related results).

checking whether our proposed approach works for 3-valued approval and making suggestions to improve the paper. We thank the anonymous reviewers for their constructive feedback and useful suggestions contributing to improving the paper.

References

1. Arrow, K.: Social Choice and Individual Values. John Wiley and Sons (1951)
2. Barberà, S., Moreno, B.: Top monotonicity: A common root for single peakedness, single crossing and the median voter result. Games Econ. Behav. **73**, 345–359 (2011)
3. Bartholdi, J., Trick, M.A.: Stable matching with preferences derived from a psychological model. Oper. Res. Lett. **5**(4), 165–169 (Oct 1986)
4. Beresnev, V.L., Davydov, A.I.: On matrices with connectedness property. Upravlyaemye Sistemy **19** (1979), in Russian
5. Betzler, N., Slinko, A., Uhlmann, J.: On the computation of fully proportional representation. J. Artif. Int. Res. **47**(1), 475–519 (May 2013)
6. Black, D.: On the rationale of group decision-making. Journal of Political Economy **56**(1), 23–34 (1948)
7. Booth, K.S., Lueker, G.S.: Testing for the consecutive ones property, interval graphs, and graph planarity using pq-tree algorithms. JCSS **13**(3), 335–379 (1976)
8. Brandt, F., Conitzer, V., Endriss, U., Lang, J., Procaccia, A.D. (eds.): Handbook of Computational Social Choice. Cambridge University Press, USA, 1st edn. (2016)
9. Bredereck, R., Chen, J., Woeginger, G.J.: A characterization of the single-crossing domain. Social Choice and Welfare **41**(4), 989–998 (2013)
10. Busing, F.M.T.A.: Advances in multidimensional unfolding. Ph.D. thesis (2010)
11. Chen, J., Hatschka, C., Simola, S.: Efficient algorithms for monroe and cc rules in multi-winner elections with (nearly) structured preferences. In: ECAI'23. pp. 397–404 (2023)
12. Constantinescu, A.C., Elkind, E.: Proportional representation under single-crossing preferences revisited. AAAI'21 **35**(6), 5286–5293 (May 2021)
13. Doignon, J.P., Falmagne, J.C.: A polynomial time algorithm for unidimensional unfolding representations. J. Algorithms **16**(2), 218–233 (Mar 1994)
14. Elkind, E., Faliszewski, P.: Recognizing 1-euclidean preferences: An alternative approach. In: Lavi, R. (ed.) Algorithmic Game Theory. pp. 146–157. Springer Berlin Heidelberg (2014)
15. Elkind, E., Faliszewski, P., Lackner, M., Obraztsova, S.: The complexity of recognizing incomplete single-crossing preferences. AAAI'15 **29**(1) (Feb 2015)
16. Elkind, E., Faliszewski, P., Slinko, A.: Clone structures in voters' preferences. Proceedings of the ACM Conference on Electronic Commerce pp. 496–513 (2012)
17. Elkind, E., Lackner, M.: Structure in dichotomous preferences. In: IJCAI'15. pp. 2019–2025 (2015)
18. Elkind, E., Lackner, M., Peters, D.: Preference restrictions in computational social choice: A survey (2022)
19. Escoffier, B., Lang, J., Öztürk, M.: Single-peaked consistency and its complexity. pp. 366–370. IOS Press, NLD (2008)
20. Faliszewski, P., Hemaspaandra, E., Hemaspaandra, L.A., Rothe, J.: The shield that never was: Societies with single-peaked preferences are more open to manipulation and control. In: TARK'09. pp. 118–127. ACM, New York, NY, USA (2009)

21. Fitzsimmons, Z.: Single-peaked consistency for weak orders is easy. In: Ramanujam, R. (ed.) TARK'15. vol. 215, pp. 127–140 (2016)
22. Guttmann, W., Maucher, M.: Variations on an ordering theme with constraints. In: IFIP TCS (2006)
23. Klinz, B., Rudolf, R., Woeginger, G.J.: Permuting matrices to avoid forbidden submatrices. Discrete Applied Mathematics **60**(1), 223–248 (1995)
24. Knoblauch, V.: Recognizing one-dimensional euclidean preference profiles. Journal of Mathematical Economics **46**, 1–5 (01 2010)
25. Lackner, M., Skowron, P.: Multi-Winner Voting with Approval Preferences. Springer International Publishing, Cham (2023)
26. Laslier, J.F., Sanver, M.R. (eds.): Handbook on Approval Voting. Springer Berlin, Heidelberg (2010)
27. Lu, T., Boutilier, C.: Budgeted social choice: From consensus to personalized decision making. In: IJCAI'11. pp. 280–286 (01 2011)
28. Magiera, K., Faliszewski, P.: Recognizing top-monotonic preference profiles in polynomial time. In: IJCAI'17. pp. 324–330 (2017)
29. Mandal, D., Procaccia, A.D., Shah, N., Woodruff, D.P.: Efficient and thrifty voting by any means necessary. In: NeurIPS'19. Curran Associates Inc., Red Hook, NY, USA (2019)
30. Mattei, N., Walsh, T.: Preflib: a library for preferences http://www.preflib.org. In: Algorithmic Decision Theory. pp. 259–270. Springer Berlin Heidelberg (2013)
31. Mirrlees, J.: An exploration in the theory of optimum income taxation. The Review of Economic Studies **38**(2), 175–208 (1971)
32. Peters, D.: Recognising multidimensional euclidean preferences. In: AAAI'17. pp. 642–648 (2017)
33. Pierczyński, G., Skowron, P.: Core-stable committees under restricted domains. In: WINE'22. pp. 311–329. Springer International Publishing, Cham (2022)
34. Procaccia, A., Rosenschein, J., Zohar, A.: On the complexity of achieving proportional representation. Social Choice and Welfare **30**, 353–362 (02 2008)
35. Roberts, K.: Voting over income tax schedules. Journal of Public Economics **8**(3), 329–340 (1977)
36. Skowron, P., Yu, L., Faliszewski, P., Elkind, E.: The complexity of fully proportional representation for single-crossing electorates. Theoretical Computer Science **569**, 43–57 (2015)
37. Sornat, K., Williams, V.V., Xu, Y.: Near-tight algorithms for the chamberlin-courant and thiele voting rules. In: De Raedt, L. (ed.) IJCAI'22. pp. 482–488 (2022)
38. Terzopoulou, Z., Karpov, A., Obraztsova, S.: Restricted domains of dichotomous preferences with possibly incomplete information. pp. 2023–2025. AAMAS'20 (2020)
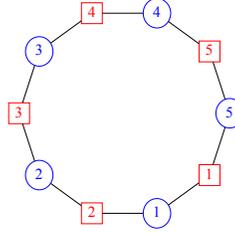
(a) Approval matrix of $\mathcal{P}_5$.   (b) Bipartite graph representation of $\mathcal{P}_5$.

Fig. 5: Construction used in the impossibility proof for $n = 5$ voters/alternatives. Fig. 5a shows the approval matrix. Fig. 5b shows the corresponding bipartite preference graph, with voters represented by red squares and candidates by blue circles.

A   Impossibility of Finite Forbidden Substructures Characterization

Single-crossing elections can be characterized as those elections not containing as subprofiles two specific elections consisting of three voters and six alternatives, and four voters and four alternatives [9]. In this appendix, we show that, in contrast, possibly single-crossing elections do not admit such a characterization in terms of finitely many forbidden subprofiles. To do so, for $n \geq 2$ define an approval profile $\mathcal{P}_n$ where n voters cast approval ballots over $m = n$ alternatives, voter $i \in [n]$ approving of candidates i and $i - 1$ and disapproving of the rest (Fig. 5a).[6] A useful notion for understanding these profiles is that of the *preference graph* of an approval profile, obtained by introducing a vertex for each voter and a vertex for each candidate, together with undirected edges drawn between voters and the candidates that they approve of.[7] Subprofiles of a profile then correspond to induced subgraphs of the preference graph respecting node types. In the case of $\mathcal{P}_n$, the preference graph is a cycle (Fig. 5b). We call a profile that is not seemingly single-crossing *minimal* if it has no non-seemingly-single-crossing proper subprofiles. The following shows that $(\mathcal{P}_n)_{n \geq 4}$ is an infinite family of minimal not seemingly single-crossing profiles, implying that this family has to be part of any forbidden substructures characterization, and hence the conclusion.

Lemma 17. *For $n \geq 4$ the profile $\mathcal{P}_n$ is not seemingly single-crossing. Moreover, any profile formed from $\mathcal{P}_n$ by removing voters and/or candidates is seemingly single-crossing.*

---

[6] Here we assume that addition and subtraction are performed modulo n, using the fact that $0 \equiv n \pmod{n}$

[7] This is the same as interpreting the approval matrix as a bipartite adjacency matrix.

Proof. To show the first part, fix some $n \geq 4$ and for brevity write $\mathcal{P} = \mathcal{P}_n$. We proceed by considering the set of NB constraints $\mathcal{C}_{\mathcal{P}}$ induced by $\mathcal{P}$. We first look at voter 1. Consider another voter $i \notin \{1, 2, n\}$. Note that voter 1 approves of candidate 1, but disapproves of candidate i, while voters i and $i+1$ disapprove of candidate 1, but approve of candidate i. Consequently, we get the NB constraints $(i, 1, i+1)$ for $i \notin \{1, 2, n\}$. Additionally, voter 1 approves of candidate n, but disapproves of candidate 2, while voters 2 and 3 do the opposite,[8] so we also get the NB constraint $(2, 1, 3)$. Together, so far we have the NB constraints $(i, 1, i+1)$ for $i \notin \{1, n\}$. In other words, voter 1 cannot be between any other two voters, so has to go either first or last in order for $\mathcal{C}_{\mathcal{P}}$ to be satisfied. By symmetry of the construction; i.e., see Fig. 5b; the same argument also applies to the other voters, but this cannot be the case, as at most two voters can go first/last in the order. To show the second part, by symmetry (see Fig. 5b) it is enough to consider the cases of removing voter 1 or candidate n. In either case, one can check that the linear order $1 \lhd 2 \lhd \ldots \lhd n$ witnesses the seemingly single-crossing property.

B   Equivalence of SSC and PSC for Weak Orders

In this appendix, we show that a profile of weak orders is seemingly single-crossing (SSC) with respect to an axis $\lhd$ if and only if it is possibly single-crossing (PSC) with respect to $\lhd$. The latter implies the former straightforwardly, so we focus on the non-trivial direction, namely that SSC with respect to $\lhd$ implies PSC with respect to $\lhd$. To this end, we show that under the assumption that the profile is SSC with respect to $\lhd$, we can extend each weak order into a linear order such that the resulting profile is single-crossing with respect to $\lhd$. As is required by our main algorithm (Theorem 16), this can be done in polynomial time. The proof that we give is a streamlined version of the argument of Elkind et al. [15].

Theorem 18. *Given a preference profile of weak orders $\mathcal{P} = (\succ_i)_{i \in [n]}$ and a linear order $\lhd$ over $[n]$ such that $\mathcal{P}$ is seemingly single-crossing with respect to $\lhd$, one can compute in polynomial time for each voter $i \in [n]$ a linear extension $\succ'_i$ of $\succ_i$ such that $(\succ'_i)_{i \in [n]}$ is single-crossing with respect to $\lhd$.*

Proof. Recall that $C = [m]$ is the set of candidates/alternatives. First, if there exists a pair of distinct candidates $(a, b) \in C^2$ such that $a \approx_i b$ for all voters $i \in [n]$, then remove b and apply the theorem recursively to the reduced instance. After getting an answer for the reduced instance, add back candidate b into all voters' rankings right next to a such that $a \succ'_i b$ holds for all voters $i \in [n]$. Hence, we may now assume that no such pair (a, b) exists.[9] Without loss of generality, further assume that $1 \lhd \ldots \lhd n$.

---

[8] No longer true for $n \leq 3$, from which the need for $n \geq 4$ becomes apparent. In fact, $\mathcal{P}_2$ and $\mathcal{P}_3$ are actually SSC.

[9] Note that if the input is presented in matrix form, the recursive steps that are performed for a given instance correspond to identifying and removing equal rows in the preference matrix, and then adding back the removed candidates at the end.

We construct the linear extensions $(\succ_i')_{i \in [n]}$ of $(\succ_i)_{i \in [n]}$ by specifying for each pair of alternatives $(a, b) \in C^2$ with $1 \leq a < b \leq m$ and each voter $i \in [n]$ whether $a \succ_i' b$ or $b \succ_i' a$. Consider such a pair (a, b). For each voter $i \in [n]$ proceed as follows: compute the highest j such that $1 \leq j \leq i$ and $a \napprox_j b$. If no such j exists, then take j to be the smallest j such that $i < j \leq n$ and $a \napprox_j b$. Note that such a j has to exist since we assumed that for any pair of distinct alternatives, at least one voter is not indifferent between them. If $a \succ_j b$, then we make $\succ_i'$ compare a and b as $a \succ_i' b$. Conversely, if $b \succ_j a$, then we make it compare a and b as $b \succ_i' a$. In Iverson bracket notation, the two cases can be more succinctly expressed as setting $[a \succ_i' b] = [a \succ_j b]$. Implicitly, we also set $[b \succ_i' a] = [b \succ_j a]$. This completes the construction, which can be seen to only require polynomial time to compute.

Next, we show the correctness of the construction. First, observe that orders $(\succ_i')_{i \in [n]}$ are extensions of $(\succ_i)_{i \in [n]}$. This is because if $a \napprox_i b$ then $j = i$ will be chosen in the construction, implying the required. Next, let us show that $(\succ_i')_{i \in [n]}$ is single-crossing with respect to $\lhd$. This amounts to showing that for any two distinct candidates $(a, b) \in C^2$ there are no three voters $i \lhd j \lhd k$ such that $a \succ_i' b$, $b \succ_j' a$ and $a \succ_k' b$. Equivalently, for any two candidates $(a, b) \in C^2$ with $a < b$ there are no three voters $i \lhd j \lhd k$ such that $a \succ_i' b, b \succ_j' a, a \succ_k' b$, or $b \succ_i' a, a \succ_j' b$, $b \succ_k' a$. To show this, consider the sequence $(s_i')_{i \in [n]}$ where $s_i' = [a \succ_i' b]$; i.e., $s_i' = 1$ if $a \succ_i' b$ and 0 if $b \succ_i' a$. The statement then amounts to there not existing indices $1 \leq i < j < k \leq n$ such that $(s_i', s_j', s_k') \in \{(0, 1, 0), (1, 0, 1)\}$, which is in turn equivalent to s' being either monotonically increasing or decreasing. We now show that this monotonicity property is indeed satisfied: consider additionally the sequence $(s_i)_{i \in [n]}$ where $s_i \in \{\bot, 0, 1\}$ satisfies $s_i = \bot$ if $a \approx_i b$ and otherwise $s_i = [a \succ_i b]$ holds. Our construction of $(\succ_i')_{i \in [n]}$ from $(\succ_i)_{i \in [n]}$ can then be reformulated as follows: for every pair of distinct candidates $(a, b) \in C^2$ with $a < b$ we construct s' from s by replacing $s_i = \bot$ entries by either 0 or 1. In particular, if $s_i = \bot$, then we compute the last non-$\bot$ element to the left of i inclusively, or the first non-$\bot$ element to the right of i if none of the previous kind were found, and then setting s_i' to that element. Because $\mathcal{P}$ is SSC with respect to $\lhd$, it follows that s is either monotonically increasing or decreasing if we ignore $\bot$ elements, so this procedure completes s into a sequence s' without $\bot$ elements that is also monotonically increasing/decreasing, as required.

It remains to show that the relations $(\succ_i')_{i \in [n]}$ produced from $(\succ_i)_{i \in [n]}$ are indeed linear orders. By construction, they are already total, irreflexive, and antisymmetric, so it remains to show that they are also transitive. Assume for a contradiction that for some voter $i \in [n]$ order $\succ_i'$ is not transitive. Hence, there exist three candidates $a, b, c \in C$ such that $a \succ_i' b \succ_i' c \succ_i' a$. We will show that this cannot be the case. To do so, consider how the values $[a \succ_i' b], [b \succ_i' c]$ and $[c \succ_i' a]$ were constructed. For ease of writing, define a linear $\sqsubset$ order on $[n]$ such that $i \sqsubset \ldots \sqsubset 1 \sqsubset i + 1 \sqsubset \ldots \sqsubset n$. Then:

- $[a \succ_i' b] = [a \succ_{j_{ab}} b]$ where $j_{ab} = \min_{\sqsubset}\{j \mid a \napprox_j b\}$;
- $[b \succ_i' c] = [b \succ_{j_{bc}} c]$ where $j_{bc} = \min_{\sqsubset}\{j \mid b \napprox_j c\}$;
- $[c \succ_i' a] = [c \succ_{j_{ca}} a]$ where $j_{ca} = \min_{\sqsubset}\{j \mid c \napprox_j a\}$.

Without loss of generality, assume $\min_{\sqsubseteq}(j_{ab}, j_{bc}, j_{ca}) = j_{ab}$. Recall that $\succ_{j_{ab}}$ is a weak order and note that $a \succ_{j_{ab}} b$.

If $a \approx_{j_{ab}} c$, then $a \succ_{j_{ab}} b$ implies that $c \succ_{j_{ab}} b$. Consequently, $j_{bc} = j_{ab}$, from which $c \succ'_i b$, contradicting the assumption that $b \succ'_i c$. As a result, $a \not\approx_{j_{ab}} c$ must hold.

If $b \approx_{j_{ab}} c$, then $a \succ_{j_{ab}} b$ implies that $a \succ_{j_{ab}} c$. Consequently, $j_{ca} = j_{ab}$, from which $a \succ'_i c$, contradicting the assumption that $c \succ'_i a$. As a result, $b \not\approx_{j_{ab}} c$, must hold.

Since $a \not\approx_{j_{ab}} c$ and $b \not\approx_{j_{ab}} c$ both hold, it follows that $j_{ab} = j_{bc} = j_{ca}$, and so $\succ_{j_{ab}}$ restricted to a, b, c is a total order, from which $\succ'_i$ also is, contradicting the assumed violation of transitivity $a \succ'_i b \succ'_i c \succ'_i a$.

C   Proofs Omitted From the Main Body

In this appendix, we provide the proofs missing from the main text. In some cases, the proofs will require stating and proving additional intermediate results.

C.1   Proof of Lemma 1

Lemma 1. *$\mathcal{P}$ is SSC with respect to a linear order $\lhd$ iff $\mathcal{C}_\mathcal{P}$ is satisfied by $\lhd$.*

Proof. Assume $\mathcal{P}$ is not SSC with respect to $\lhd$. Then, there are voters $i \lhd j \lhd k$ and candidates $a, b \in C$ such that $a \succ_i b, b \succ_j a$ and $a \succ_k b$. By construction of $\mathcal{C}_\mathcal{P}$, this means that $(i, j, k) \in \mathcal{C}_\mathcal{P}$, so $\mathcal{C}_\mathcal{P}$ is not satisfied by $\lhd$. Conversely, assume $\mathcal{C}_\mathcal{P}$ is not satisfied by $\lhd$. Then, there is $(i, j, k) \in \mathcal{C}_\mathcal{P}$ such that either $i \lhd j \lhd k$ or $k \lhd j \lhd i$. By construction of $\mathcal{C}_\mathcal{P}$, this means that $a \succ_i b, b \succ_j a, a \succ_k b$ and either $i \lhd j \lhd k$ or $k \lhd j \lhd i$. Either way, this means that $\mathcal{P}$ is not SSC with respect to $\lhd$.

C.2   Proof of Lemma 2

Recall the following basic fact:

Proposition 20. *A NB constraint (i, j, k) is satisfied by a linear order $\lhd$ iff $[i \lhd j] = [k \lhd j]$.*

Lemma 2. *Linear orders $\lhd$ satisfying the NB constraints in $\mathcal{C}$ correspond bijectively to satisfying assignments of $\Phi_\mathcal{C}$ by $x_{i,j} = [i \lhd j]$. Hence, $\mathcal{C}$ is satisfiable iff $\Phi_\mathcal{C}$ is satisfiable.*

Proof. Assume linear order $\lhd$ satisfies the constraints in $\mathcal{C}$, then let us set $x_{i,j} = [i \lhd j]$. As a result, constraints in $\langle 2 \rangle$ are satisfied due to Proposition 20, while constraints in $\langle 1 \rangle$ and $\langle 3 \rangle$ are satisfied because $\lhd$ is antisymmetric and transitive. Conversely, assume values $x_{i,j}$ are a satisfying assignment of $\Phi_\mathcal{C}$. Then, one can define $\lhd$ by setting $[i \lhd j] = x_{i,j}$, which is a linear order because $\langle 1 \rangle$ and $\langle 3 \rangle$ enforce antisymmetry and transitivity. Moreover, $\lhd$ satisfies the constraints in $\mathcal{C}$ because $\Phi_\mathcal{C}$ satisfies constraint set $\langle 2 \rangle$, by using Proposition 20.

C.3   Proof of Lemmas 3 and 4

Lemma 3. *An assignment $(x_u)_{u \in V(G_C)}$ satisfies constraint set $\langle 2' \rangle$ if and only if for each connected component S of G_C it holds that $x_u = x_v$ for all $u, v \in S$.*

Proof. Assume x is an assignment satisfying set $\langle 2' \rangle$. Let S be a connected component of G_C and $u, v \in S$ be arbitrary. Since S is a connected component, consider a path $u = t_0 \,\text{---}\, t_1 \,\text{---}\, \ldots \,\text{---}\, t_k = v$ in G_C. By construction of G_C, this path corresponds to constraints in set $\langle 2' \rangle$ stipulating that $x_{t_0} = \ldots = x_{t_k}$, from which we get $x_u = x_v$ as x satisfies constraint set $\langle 2' \rangle$. Conversely, assume x is an assignment and that for each connected component S of G_C it holds that $x_u = x_v$ for all $u, v \in S$. Consider a constraint $x_u = x_v$ in set $\langle 2' \rangle$. By construction of G_C, the edge $u \,\text{---}\, v$ exists in G_C, so u and v are in the same connected component of G_C, from which our assumption gives that $x_u = x_v$, so the constraint in $\langle 2' \rangle$ is satisfied.

Lemma 4. *Let S be a connected component in G_C, then $\overline{S} := \{\overline{s} : s \in S\}$ is also a connected component in G_C. Note moreover that $S = \overline{S}$ or $S \cap \overline{S} = \varnothing$.*

Proof. By construction of G_C, an edge $a \,\text{---}\, b$ exists in G_C if and only if the edge $\overline{a} \,\text{---}\, \overline{b}$ also exists in G_C. By induction, a and b are connected by a path in G_C if and only if $\overline{a}$ and $\overline{b}$ are connected by a path in G_C. This means that vertex set $\overline{S}$ is connected in G_C. To show that $\overline{S}$ is a connected component, assume for a contradiction that $\overline{S}$ was contained in a larger connected component $\overline{S} \subsetneq S' \subseteq V(G_C)$. Note that this also means that $\overline{\overline{S}} = S \subsetneq \overline{S'}$. By the same argument as before applied to S' instead of S, we get that $\overline{S'}$ is connected in G_C. However, $S \subsetneq \overline{S'}$, contradicting the fact that S is a connected component.

C.4   Proof of Lemma 8

In this section, we prove Lemma 8. Before commencing, we make an elementary observation:

Proposition 24. *Appending to an approval matrix $\mathcal{P}$ a copy of one of the rows of $\mathcal{P}$ leaves the set of NB constraints $\mathcal{C}_{\mathcal{P}}$ unchanged. Naturally, the formula and colorful graphs also remain unchanged.*

This fact will be useful in our proofs because it allows for assuming without loss of generality that any collection of NB constraints in $\mathcal{C}_{\mathcal{P}}$ is generated by pairwise disjoint unordered pairs of rows/candidates; e.g., if $(1, 2, 3)$ and $(4, 5, 6)$ are NB constraints witnessed by candidate pairs $\{a, b\}$ and $\{c, d\}$, then without loss of generality $\{a, b\} \cap \{c, d\} = \varnothing$. This is because had this not been the case, one could have added copies of the rows which are mentioned multiple times.

To prove Lemma 8, we will need two intermediate results that together will give the conclusion.
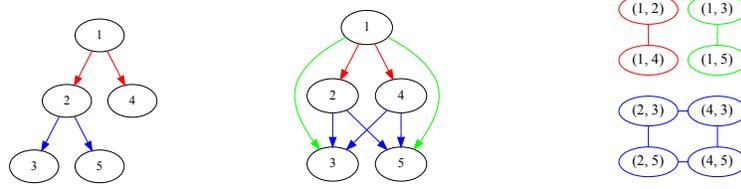
(a) Assumed edge colors.   (b) Implied edge colors.   (c) Impl. edges in formula graph.

Fig. 6: Illustration of Lemma 25. Fig. 6a depicts the assumptions: edges $(1, 2) — (1, 4)$ and $(2, 3) — (2, 5)$ exist in the formula graph, with the first two colored red and the last two blue. Additionally, edge $(1, 3)$ is neither red nor blue. Fig. 6b shows the edge colors implied by the setup, where green is some arbitrary color other than red and blue. Fig. 6c shows the implied edges in the formula graph, whose existence lead to the color configuration in the second figure.

```
0 1 x 1 z        0 1 0 1 z        0 1 0 1 z        0 1 0 1 0        0 1 0 1 0
1 0 y 0 t        1 0 0 0 t        1 0 0 0 0        1 0 0 0 0        1 0 0 0 0
u 0 1 p 1        0 0 1 p 1        0 0 1 p 1        0 0 1 p 1        0 0 1 0 1
v 1 0 q 0        0 1 0 q 0        0 1 0 q 0        0 1 0 q 0        0 1 0 q 0
```

(a) Notation.   (b) x, y, u, v are 0.   (c) t is 0.   (d) z is 0.   (e) p is 0.

Fig. 7: Steps in the proof of Lemma 25. The final matrix reached implies the conclusions regardless of the value of $q \in \{0, 1\}$.

Lemma 25. *Consider five voters, say $1, 2, 3, 4$ and 5, such that the formula graph has edges $(1, 2) — (1, 4)$ and $(2, 3) — (2, 5)$, and that, moreover, $(1, 2)$ and $(2, 3)$ have different colors in the colorful graph. If edge $(1, 3)$ is either not present in the colorful graph, or it has a different color than $(1, 2)$ and $(2, 3)$, then the formula graphs is guaranteed to also contain edges $(1, 3) — (1, 5)$, $(4, 3) — (4, 5)$, $(2, 3) — (4, 3)$ and $(2, 5) — (4, 5)$. The scenario is illustrated in Fig. 6.*

Proof. Matrix $\mathcal{P}$ induces NB constraints $(2, 1, 4)$ and $(3, 2, 5)$. By Proposition 24 and the discussion afterward, we can assume that matrix $\mathcal{P}$ has the form in Fig. 7a. First, we prove that $x = y$. Assume the contrary, and make a case distinction:

If $xy = 01$, then $\mathcal{P}$ induces the NB constraint $(1, 2, 3)$, so edge $(1, 2) — (3, 2)$ is in the formula graph, so $(1, 2)$ and $(3, 2)$ are in the same component, meaning that they bear the same color, contradicting the hypothesis.

If $xy = 10$, then $\mathcal{P}$ induces the NB constraint $(2, 1, 3)$, so edge $(2, 1) — (3, 1)$ is in the formula graph, so the component of edge $(3, 1)$ has size at least two, meaning that it occurs in the colored graph, contradicting the hypothesis.

Subsequently, we prove that $u = v$. Assume the contrary. If $uv = 10$, then $\mathcal{P}$ induces the NB constraint $(1, 2, 3)$, and a contradiction follows similarly to the first case above. If $uv = 01$, then $\mathcal{P}$ induces the NB constraint $(2, 3, 1)$, from which the edge $(2, 3)$ — $(1, 3)$ is in the formula graph, so the component of edge $(3, 1)$ has size at least two, meaning that it occurs in the colored graph, contradicting the hypothesis.

Now, let us prove that $x = u$. Assume the contrary. If $xu = 01$, then $\mathcal{P}$ induces the NB constraint $(1, 2, 3)$, and a contradiction follows as before. If $xu = 10$, then $\mathcal{P}$ again induces the NB constraint $(1, 2, 3)$, again a contradiction.

Since $\mathcal{P}$ and $\overline{\mathcal{P}}$, which is the profile obtained from $\mathcal{P}$ by exchanging zeros and ones, induce the same graphs, note that the cases with $x = 1$ and $x = 0$ are isomorphic with respect to complementing the matrix and permuting the row pairs $(1, 2)$ and $(3, 4)$. Therefore, it is enough to consider the case $x = 0$, so our matrix has the form in Fig. 7b.

Now, note that if $t = 1$, then $\mathcal{P}$ induces the NB constraint $(1, 2, 5)$, from which $(1, 2)$ and $(5, 2)$ have the same color in the colored graph, contradicting our hypothesis, so $t = 0$. Therefore, matrix $\mathcal{P}$ has the form in Fig. 7c. Notice that this induces the NB constraint $(3, 1, 5)$.

Moreover, observe that if $z = 1$, then $\mathcal{P}$ induces the NB constraint $(2, 1, 5)$, which, together with the presence of constraint $(3, 1, 5)$ means that $(1, 2)$ and $(1, 3)$ are colored with the same color, contradicting the hypothesis, so $z = 0$ holds. The matrix right now looks like in Fig. 7d.

Finally, observe that if $p = 1$, then $\mathcal{P}$ induces the NB constraint $(4, 1, 5)$, which, together with the presence of constraints $(3, 1, 5)$ and $(2, 1, 4)$, means that $(1, 2)$ and $(1, 3)$ are colored with the same color, contradicting the hypothesis, so $p = 0$ holds. The matrix now looks like in Fig. 7e. Looking only at the first three rows, we already get the conclusion. It can also be checked that both options for q lead to no contradictions, but this is not necessary.

Recall that a critical point to note is that Lemma 25 continues to hold when some of the five voters happen to be the same, as can be seen by creating distinct copies of such voters and then invoking the lemma. Similar considerations hold for all our results of this sort.

Lemma 26. *Consider five voters, say* $1, 2, 3, 4$ *and* 5, *such that the formula graph has edges* $(1, 2)$ — $(4, 2)$ *and* $(2, 3)$ — $(2, 5)$, *and that, moreover,* $(1, 2)$ *and* $(2, 3)$ *have different colors in the colorful graph. If edge* $(1, 3)$ *is either not present in the colorful graph, or it has a different color than* $(1, 2)$ *and* $(2, 3)$, *then the formula graphs is guaranteed to also contain edges* $(4, 3)$ — $(1, 3)$, $(4, 3)$ — $(4, 5)$, $(1, 3)$ — $(1, 5)$ *and* $(4, 5)$ — $(1, 5)$. *The scenario is illustrated in Fig. 8.*

Proof. Matrix $\mathcal{P}$ induces NB constraints $(1, 2, 4)$ and $(3, 2, 5)$. By Proposition 24 and the discussion afterwards, we can assume that matrix $\mathcal{P}$ has the form in Fig. 9a. Notice that matrices in Fig. 9a and Fig. 7a differ only in the two red entries in column 4 being swapped, so the proof of Lemma 25 works unmodified until the first time column 4 is mentioned, so we get that it is enough to consider the case $x = y = u = v = z = t = 0$, depicted in Fig. 9b. Moreover, notice that, in
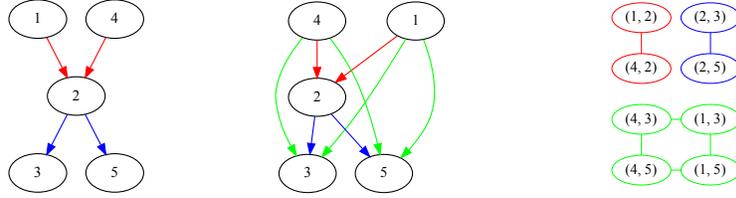
(a) Assumed edge colors.   (b) Implied edge colors.   (c) Impl. edges in formula graph.

Fig. 8: Illustration of Lemma 26. Fig. 8a depicts the assumptions: edges $(1, 2) — (4, 2)$ and $(2, 3) — (2, 5)$ exist in the formula graph, with the first two colored red and the last two blue. Additionally, edge $(1, 3)$ is neither red nor blue. Fig. 8b shows the edge colors implied by the setup, where green is some arbitrary color other than red and blue. Fig. 8c shows the implied edges in the formula graph, whose existence lead to the color configuration in the second figure.

(a) Notation.   (b) x, y, u, v, z, t are 0.   (c) Self-symmetry.   (d) p, q are 0.

Fig. 9: Steps in the proof of Lemma 26.

contrast to the previous lemma, the scenario in Fig. 8a is symmetric with respect to interchanging the red/blue components. In particular, if we interchange the roles of candidate pairs $(1, 3)$ and $(4, 5)$ and additionally permute row pairs $(1, 4)$ and $(2, 3)$ in Fig. 9a we get the matrix in Fig. 9c, which is identical up to relabelling of the variables x, y, etc. Consequently, reapplying our argument so far to the matrix in Fig. 9c gives us that $v = u = y = x = q = p = 0$. Altogether, we now know that $x = y = u = v = z = t = p = q = 0$, fact depicted in Fig. 9d. Once again, looking only at the first three rows, we get our conclusion.

We can now prove Lemma 8 by combining Lemmas 25 and 26, as follows:

Lemma 8. *Consider four voters, say* $1, 2, 3, 4$, *such that the formula graph contains the edge* $(1, 2) — (4, 2)$ *and that edge* $(2, 3)$ *exists in the colorful graph and has a different color than* $(1, 2)$. *If edge* $(1, 3)$ *is either not present in the colorful graph, or it has a different color than* $(1, 2)$ *and* $(2, 3)$, *then edge* $(1, 3) — (4, 3)$ *is guaranteed to be in the formula graph. The scenario is illustrated in Fig. 3.*

Proof. Since edge $(2, 3)$ is in the colorful graph, an edge $(2, 3) — (x, y)$ has to exist in the formula graph. By construction of the formula graph, either $x = 2$
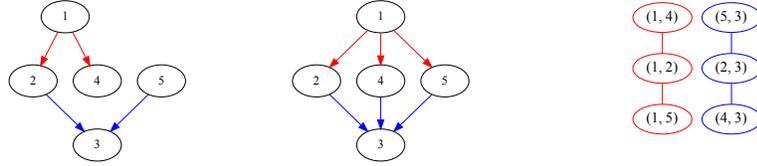
(a) Assumed edge colors.   (b) Implied edge colors.   (c) Impl. edges in formula graph.

Fig. 10: Illustration of Lemma 28. Fig. 10a depicts the assumptions: edges $(1,2) — (1,4)$ and $(2,3) — (5,3)$ exist in the formula graph, with the first two colored red and the last two blue. Additionally, edge $(1,3)$ is neither red nor blue. Fig. 10b shows the edge colors implied by the setup. Fig. 10c shows the implied edges in the formula graph, whose existence lead to the color configuration in the second figure.

(a) Notation.   (b) x, y, u, v are 0.   (c) t is 0.   (d) p is 0.

Fig. 11: Steps in the proof of Lemma 28. The final matrix reached implies the conclusions regardless of the values of $(q, z) \in \{0, 1\}^2$.

or $y = 3$. If $x = 2$, then Lemma 26 gives the conclusion. If $y = 3$, then Lemma 25 gives the conclusion.

C.5   Proof of Theorem 10

In this section, we prove Theorem 10. In order to do so, we will need yet another intermediate result similar in spirit to Lemmas 25 and 26, as follows:

Lemma 28. *Consider five voters, say $1, 2, 3, 4$ and 5, such that the formula graph has edges $(1,2) — (1,4)$ and $(2,3) — (5,3)$, and that, moreover, $(1,2)$ and $(2,3)$ have different colors in the colorful graph. If edge $(1,3)$ is either not present in the colorful graph, or it has a different color than $(1,2)$ and $(2,3)$, then the formula graphs is guaranteed to also contain edges $(1,2) — (1,5)$ and $(2,3) — (4,3)$. The scenario is illustrated in Fig. 10.*

Proof. Matrix $\mathcal{P}$ induces NB constraints $(2,1,4)$ and $(2,3,5)$. By Proposition 24 and the discussion afterwards, we can assume that matrix $\mathcal{P}$ has the form in Fig. 11a. Notice that matrices in Fig. 11a and Fig. 7a differ only in the two blue entries in column 5 being swapped, so the proof of Lemma 25 works unmodified
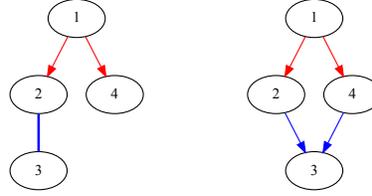
(a) Assumed colors.   (b) Implied colors.

Fig. 12: Illustration of Lemma 29. Fig. 12a depicts the assumptions: edge $(1,2) — (1,4)$ exists in the formula graph (colored red) and edge $(2,3)$ exists in the colorful graph and bears a different color (colored blue). Additionally, edge $(1,3)$ is neither red nor blue. Fig. 12b shows the colors implied by the setup.

until the first time column 5 is mentioned, so we get that it is enough to consider the case $x = y = u = v = 0$, depicted in Fig. 11b.

Now, notice that if $t = 1$, then $\mathcal{P}$ induces the NB constraint $(1,3,5)$, from which $(1,3)$ and $(5,3)$ have the same color in the colorful graph, which happens to also be the color of $(2,3)$, contradicting our hypothesis, so $t = 0$ holds, fact depicted in Fig. 11c.

Similarly, notice that if $p = 1$, then $\mathcal{P}$ induces the NB constraint $(3,1,4)$, from which $(1,3)$ and $(1,4)$ have the same color in the colorful graph, which happens to also be the color of $(1,2)$, contradicting our hypothesis, so $p = 0$ holds, fact depicted in Fig. 11d. As a remark, observe that we could have also argued that Fig. 8a is symmetric with respect to interchanging red/blue components, and then use a symmetry argument like in the proof of Lemma 26 to deduce that $p = 0$, but this time not much work is saved in contrast to a direct argument.

Finally, irrespective of the values of q and t, the current matrix $\mathcal{P}$ already induces NB constraints $(2,1,5)$ and $(2,3,4)$, implying our conclusion. It can also be checked that all four options for $(z,q) \in \{0,1\}^2$ lead to no contradictions, but this is not necessary.

In order to streamline the proof of Theorem 10, we summarize the essence of Lemmas 25 and 28 into the following lemma, similar in spirit to Lemma 8:

Lemma 29. *Consider four voters, say* $1,2,3,4$, *such that the formula graph contains the edge* $(1,2) — (1,4)$ *and that edge* $(2,3)$ *exists in the colorful graph and has a different color than* $(1,2)$. *If edge* $(1,3)$ *is either not present in the colorful graph, or it has a different color than* $(1,2)$ *and* $(2,3)$, *then edge* $(2,3) — (4,3)$ *is guaranteed to be in the formula graph. The scenario is illustrated in Fig. 12a and Fig. 12b.*

Proof. Since edge $(2,3)$ is in the colorful graph, an edge $(2,3) — (x,y)$ has to exist in the formula graph. By construction of the formula graph, either $x = 2$

or $y = 3$. If $x = 2$, then Lemma 25 gives the conclusion. If $y = 3$, then Lemma 28 gives the conclusion.

We are now ready to prove Theorem 10, as follows:

Theorem 10. *Consider three voters, say* $1, 2$ *and* 3*, such that edges* $(1, 2)$ *and* $(2, 3)$ *exist in the colorful graph and have different colors, while edge* $(1, 3)$ *does not exist in the colorful graph. Then, there exists a set of voters* S *such that* $2 \in S$ *and* $1, 3 \notin S$ *with the property that the connected components of* $(1, 2)$ *and* $(2, 3)$ *in the formula graph are* $\{1\} \times S$ *and* $S \times \{3\}$ *respectively.*

Proof. Consider an inclusion-maximal set S such that $2 \in S$, $1, 3 \notin S$ and the subgraphs of the formula graph induced by $\{1\} \times S$ and $S \times \{3\}$ are connected. Note that the connectivity assumption implies that in the colorful graph edges in $\{1\} \times S$ have the color of $(1, 2)$ and edges in $S \times \{3\}$ have the color of $(2, 3)$. Denote by C_{12} and C_{23} the connected components of $(1, 2)$ and $(2, 3)$ in the formula graph. We will now show that $C_{12} = \{1\} \times S$ and $C_{23} = S \times \{3\}$. Assume for a contradiction that this was not the case, then, by symmetry, without loss of generality, $C_{12} \neq \{1\} \times S$. Together with the above, this means that $\{1\} \times S \subsetneq C_{12}$. Because C_{12} induces a connected subgraph in the formula graph, and $\varnothing \neq \{1\} \times S \subsetneq C_{12}$, it follows that there is an edge in the formula graph crossing the cut $(\{1\} \times S, C_{12} \setminus (\{1\} \times S))$. This edge can be of one of two forms: either $(1, s) \text{---} (x, s)$ or $(1, s) \text{---} (1, x)$, where $s \in S$ and $x \notin \{1, s\}$. We tackle the two cases separately in the paragraphs below.

If the edge crossing the cut is of the form $(1, s) \text{---} (x, s)$, then instantiate Lemma 8 with $1 \mapsto 1, 2 \mapsto s, 3 \mapsto 3$ and $4 \mapsto x$. This is sound because the edge $(1, s) \text{---} (x, s)$ exists in the formula graph and $(1, s), (s, 3)$ have different colors in the colorful graph, together with the fact that $(1, 3)$ does not appear in the colorful graph. As a result, we get that edge $(1, 3) \text{---} (x, 3)$ is in the formula graph, contradicting the nonexistence of edge $(1, 3)$ in the colorful graph.

On the other hand, if the edge crossing the cut is of the form $(1, s) \text{---} (1, x)$, then this time we instantiate Lemma 29, with values $1 \mapsto 1, 2 \mapsto s, 3 \mapsto 3$ and $4 \mapsto x$. This is sound because edge $(1, s) \text{---} (1, x)$ is in the formula graph and $(1, s), (s, 3)$ have different colors in the colorful graph, together with the fact that $(1, 3)$ does not appear in the colorful graph. As a result, we get that edge $(s, 3) \text{---} (x, 3)$ appears in the formula graph. As a result, the subgraphs induced by $\{1\} \times (S \cup \{x\})$ and $(S \cup \{x\}) \times \{3\}$ in the formula graph are connected. If we could also show that $x \notin (S \cup \{1, 3\})$ then we would get that our S was not maximal with respect to inclusion, a contradiction. We now show this last fact to conclude the proof: $x = 1$ was forbidden above; $x = 3$ would make our cut edge be $(1, s) \text{---} (1, 3)$, implying that $(1, 3)$ appeared in the colorful graph, a contradiction; finally $x \in S$ would mean that $(1, s) \text{---} (1, x)$ is not a cut edge.

C.6   Proof of Lemma 12

In this section, we prove Lemma 12, which we repeat below for convenience:

Lemma 12 (Strengthened Lemma 8). *Consider four voters, say* $1, 2, 3, 4$*, such that the formula graph contains the edge* $(1, 2) - (4, 2)$ *and that edge* $(2, 3)$ *exists in the colorful graph. If the formula graph does not contain edges* $(3, 2) - (1, 2)$ *and* $(3, 2) - (4, 2)$*, then the formula graph is guaranteed to contain the edge* $(1, 3) - (4, 3)$*.*

The proof closely follows that of Lemma 8, except for requiring stronger versions of Lemmas 25 and 26, which we state and prove next. The proofs remain similar in spirit, by identifying small matrices with blanks to be filled in, like Fig. 7a. Unfortunately, the weaker assumptions make it rather challenging to proceed as before in a principled manner. Instead, we employ a computer program to try out all the possibilities.

Lemma 32 (Strengthened Lemma 25). *Consider five voters, say* $1, 2, 3, 4$ *and* 5*, such that the formula graph has edges* $(1, 2) - (1, 4)$ *and* $(2, 3) - (2, 5)$*, and that, moreover, the formula graph does not contain edges* $(1, 2) - (3, 2)$ *or* $(1, 2) - (5, 2)$*. Then, the formula graphs is guaranteed to contain the edge* $(1, 3) - (1, 5)$*.*

Proof. Similarly to the proof of Lemma 25, we can assume that matrix $\mathcal{P}$ has the form in Fig. 7a. However, this time our premises are weaker, so we did not manage to write a direct "linear" argument. However, we tried out all $2^8 = 256$ options for the tuple (x, y, z, t, u, v, p, q) with a computer and 68 of them did not contradict our hypothesis, and for all of them the conclusion held. The relevant code is included with the paper, and instructions for running it can be found in Appendix C.7.

Lemma 33 (Strengthened Lemma 26). *Consider five voters, say* $1, 2, 3, 4$ *and* 5*, such that the formula graph has edges* $(1, 2) - (4, 2)$ *and* $(2, 3) - (2, 5)$*, and that, moreover, the formula graph does not contain edges* $(1, 2) - (3, 2)$ *or* $(1, 2) - (5, 2)$*. Then, the formula graphs is guaranteed to contain the edge* $(1, 3) - (1, 5)$*.*

Proof. Similarly to the proof of Lemma 26, we can assume that matrix $\mathcal{P}$ has the form in Fig. 9a. However, this time our premises are weaker, so we did not manage to write a direct "linear" argument. However, we tried out all $2^8 = 256$ options for the tuple (x, y, z, t, u, v, p, q) with a computer and 59 of them did not contradict our hypothesis, and for all of them the conclusion held. The relevant code is included with the paper, and instructions for running it can be found in Appendix C.7.

C.7   Code for Proofs of Lemmas 32 and 33

Together with the paper, we include a C++ source file called `exhaust_proof.cpp`. This file can be found at https://github.com/Andrei1998/rec-sc-from-approval/.

To compile the code, use the following command:

```
g++ -std=c++11 -O2 exhaust_proof.cpp -o exhaust_proof
```

To run the check required in the proof of Lemma 32, invoke the compiled program as:

```
echo "1" | ./exhaust_proof
```

To run the check required in the proof of Lemma 33, invoke the compiled program as:

```
echo "2" | ./exhaust_proof
```