
Felix Brandt  
Institut für Informatik  
TU München  
85748 Garching, Germany

Markus Brill  
Department of Computer Science  
Duke University  
Durham, NC 27708, USA

Edith Hemaspaandra  
Department of Computer Science  
Rochester Institute of Technology  
Rochester, NY 14623, USA

Lane A. Hemaspaandra  
Department of Computer Science  
University of Rochester  
Rochester, NY 14627, USA

April 23, 2010; revised September 10, 2013

Abstract

For many election systems, bribery (and related) attacks have been shown NP-hard using constructions on combinatorially rich structures such as partitions and covers. It is important to learn how robust these hardness protection results are, in order to find whether they can be relied on in practice. This paper shows that for voters who follow the most central political-science model of electorates—single-peaked preferences—those protections vanish. By using single-peaked preferences to simplify combinatorial covering challenges, we for the first time show that NP-hard bribery problems—including those for Kemeny and Llull elections—fall to polynomial time for single-peaked electorates. By using single-peaked preferences to simplify combinatorial partition challenges, we for the first time show that NP-hard partition-of-voters problems fall to polynomial time for single-peaked electorates. We show that for single-peaked electorates, the winner problems for Dodgson and Kemeny elections, though $\Theta^p_2$-complete in the general case, fall to polynomial time. And we completely classify the complexity of weighted coalition manipulation for scoring protocols in single-peaked electorates.

*This work was supported in part by ARC grant DP1101011792, DFG grants BR-2312/3-2, DFG grant BR-2312/6-1, NSF grants CCF-0426761, CCF-0915792, CCF-1101452, CCF-1101479, and IIS-0713061, the European Science Foundation’s EUROCORES program LogICCC, and Friedrich Wilhelm Bessel Research Awards to Edith Hemaspaandra and Lane A. Hemaspaandra. This work was done in part while Felix Brandt was at Ludwig-Maximilians-Universität München and Markus Brill was at Ludwig-Maximilians-Universität München and TU München, and was done in part during visits of Edith Hemaspaandra and Lane A. Hemaspaandra to Heinrich-Heine-Universität Düsseldorf and Ludwig-Maximilians-Universität München. A preliminary version of this paper appeared in the proceedings of the 24th AAAI Conference on Artificial Intelligence, July 2010 [BBHH10].
1 Introduction

Elections are perhaps the most important framework for preference aggregation. An election (rule) is a mapping that takes as input the preferences of the voters with respect to the set of candidates (alternatives) and returns a set of “winners,” which is some subset of the candidate set. Elections are central in preference aggregation among humans—in everything from political elections to selecting good singers on popular television shows. Elections are rapidly increasing in importance in electronic settings such as multiagent systems, and have been used or proposed for such varied tasks as recommender systems and collaborative filtering [GMHS99, PHG00], web spam reduction and improved web-search engines [DKNS01], and planning [ER97]. In electronic settings, elections may have huge numbers of voters and alternatives.

One natural worry with elections is that agents may try to slant the outcome, for example, by bribing voters. Motivated by work from economics and political science showing that reasonable election systems always allow manipulation in some cases [Gib73, Sat75, DS00], starting in 1989, Bartholdi, Orlin, Tovey, and Trick [BTT89, BO91, BTT92] made the thrilling suggestion that elections be protected via complexity theory—namely, by making the attacker’s task NP-hard. This line has been active ever since, and has resulted in NP-hardness protections being proven for many election systems, against such attacks as bribery (the attacker has a budget with which to buy and alter voters’ votes [FHH09]), manipulation (a coalition of voters wishes to set its votes to make a given candidate win [BTT89, BO91]), and control (an agent seeks to make a given candidate win by adding/deleting/partitioning voters or candidates [BTT92]). The book chapter [FHHR09b] surveys such NP-hardness results, which apply to many important election systems such as plurality, single transferable voting, and approval voting.

In the past few years, a flurry of papers have come out asking whether the NP-hardness protections are satisfying. In particular, the papers explore the possibility that heuristic algorithms may do well frequently or that approximation algorithms may exist. These papers can themselves be questioned. For example, the most influential “frequency” paper [FKN08] (see also its journal version [FKKN11]) assumes each voter has a random and independent candidate preference ordering, and that model does not seem to reflect typical voter behavior. And as to approximations, many people would not be happy with using an algorithm that may well not output the optimal solution, although perhaps in electronic settings such approximations might sometimes be acceptable.

The present paper questions the NP-hardness results from a completely different direction. In political science, perhaps the most “canonical” model of electorates is the unidimensional single-peaked model, in which the electorate has preferences over some one-dimensional spectrum (e.g.,
“very liberal through very conservative”) along which the candidates are also located, and in which each voter’s preferences (loosely put) have a peak, with affinity declining as one moves away from the peak. A brilliant paper by Walsh [Wal07] recently asked whether NP-hardness protections against manipulation fall apart if electorates are single-peaked. For the case Walsh looked at, the answer he proved is “no”; he looked at a particular NP-hardness manipulation protection and proved it holds even for single-peaked societies. Faliszewski et al. [FHR11], inspired by Walsh’s work, looked at a range of election systems and came to the sharply differing conclusion that for many crucial cases, NP-hardness protections against manipulation and control vanish for single-peaked electorates.

The present paper is in this young line of research on complexity of manipulative actions in the context of single-peaked electorates. Our work seeks to take this line of research in new directions, and to improve one existing direction, though the following contributions:

1. We (Section 3) show that checking who the winner is in Dodgson, Young, and Kemeny elections, which is known to be \( \Theta^p_2 \)-complete in the general case (by [HHR97], by this paper’s Theorem B.2 based on adapting a proof of [RSV03], and by [HSV05], respectively), is in polynomial time for single-peaked electorates.

   Our algorithm that shows this for Dodgson elections is a good example of the general technical theme of this paper: That single-peakedness often precludes combinatorial explosion. In this particular case, single-peakedness will simplify the seemingly exponential-sized search space over “series of exchanges to provide upper bounds on Dodgson scores,” and will allow us to instead search over a polynomial-sized possibility space related to a particular, simple set of exchanges happening and limited to at most two voters.

2. We (Section 4) for the first time study the effect of single-peaked electorates on the complexity of bribery. We show that many NP-hardness protections against bribery in the general case vanish for single-peaked electorates. To show this, we give polynomial-time bribery algorithms for single-peaked electorates in many settings. Our polynomial-time algorithms apply to approval voting and to the rich range of “weak-Condorcet consistent” election systems and even to systems that are merely known to be weak-Condorcet consistent when the electorate is single-peaked, including weakBlack, weakDodgson, Fishburn, Kemeny, Llull, Maximin, Schwartz, Young, and the two variants of Nanson due to Fishburn and Schwartz.

   The right general interpretation of what underlies this is that the NP-hardness results use (in the outputs of the reductions establishing NP-hardness) sets of voter preferences that are so intricate that they simply cannot be realized by single-peaked societies. The practical lesson
is that we should be very skeptical about NP-completeness results if our electorate may have limitations (such as single-peakedness) on the ensembles of votes it produces. And the specific technical reason we can obtain polynomial-time bribery algorithms is that the NP-hardness proofs were based on the combinatorially rich structure of covering problems (whose core challenge is the “incomparability” of voters), but we will (see the proof of Theorem 4.2) use single-peakedness to create a “directional” attack on covering problems that has the effect of locally removing incomparability.

3. We (Section 5) for the first time study the effect of single-peaked electorates on the complexity of control by partition of voters, in which the voters are partitioned into two groups that vote on the candidates in “primary” elections, and only the winners of the primaries compete in the final election. This is one of the seven types of control introduced in the seminal control paper of Bartholdi, Tovey, and Trick [BTT92], but control by partition of voters has not been previously addressed for the single-peaked case. We show that some known NP-hardness protections against control-by-partition vanish for single-peaked electorates, and we do so by giving polynomial-time algorithms for single-peaked control by partition.

The “general interpretation” and “practical lesson” from this are the same as was just mentioned for the bribery case. However, the technical way we obtain this control-by-partition result differs here. The technical challenge here is the exponential number of partitions, and our algorithms circumvent this by using single-peakedness to allow us to in effect structure that huge number of partitions into a polynomial number of classes of partitions such that for each class we can look just at the class rather than having to explore each of its member partitions.

The shared technical theme here and in the bribery case is that single-peakedness can be used to tame the combinatorial explosion (of partitions and covers) that in the general case protected elections from attack, and in particular single-peakedness yields polynomial-time attack algorithms.

4. Our final contribution is a strong extension of an important result from Faliszewski et al. [FHHR11]. For the broad class of election systems known as scoring protocols, Faliszewski et al. gave a complete characterization of the computational complexity of the (weighted, coalition) manipulation problem in the case of single-peaked elections with three candidates. Such characterizations are important as they tell both which systems are manipulable and what it is about the systems that makes them manipulable. We (Section 6) extend this by providing, for single-peaked electorates, a complete characterization of easy manipu-
lability for scoring protocols. That is, we extend the three-candidate theorem of [FHHR11] to a result that holds for any number of candidates, and that allows one to immediately “read off” the complexity of manipulation of any scoring protocol, for single-peaked electorates.

Our proof organization is as follows. Appendix A contains one “spotlight” proof for each of the paper’s four results sections. These seek to give the key flavor of our techniques, and in the main body’s text, we will typically try to informally describe these proofs’ ideas or approaches. The first three of the four spotlight proofs directly support (as does the fourth spotlight proof, in part) the paper’s technical theme that single-peakedness tames combinatorial explosion. Appendix B contains, for completeness, proofs of all our other results (and some definitions omitted from the main text).

2 Preliminaries

Election Systems, Preferences, and weakCondorcet-Consistency

An election system is a mapping from a finite set of candidates $C$ and a finite collection $V$ of voter preferences over those candidates to a collection $W \subseteq C$ called the winner set.\footnote{In social choice theory, this is called a social choice correspondence. Social choice theorists often exclude the case allowing the function to have an empty set of winners, but following Bartholdi, Tovey, and Trick [BTT92] and many computationally oriented papers, we do not artificially exclude that case in our definitions. However, except for elections with zero candidates, the only systems we discuss that might ever output an empty set of winners are Condorcet and weakCondorcet.} For all but one of the election systems we cover, each voter’s preference is a linear order (by which we always mean a strict linear order: an irreflexive, antisymmetric, complete, transitive relation) over the candidates. For the election system called approval voting, each voter votes by a bit-vector, approving or disapproving of each candidate separately. Voters’ preferences are input as a list of ballots (i.e., votes), so if multiple voters have the same preference, the ballot of each will appear separately in $V$.

We now very briefly describe the election systems most central to this paper. In approval voting, preferences are approval vectors, and each candidate who gets the highest number of approvals among the candidates belongs to the winner set. In all the other systems we use, voters will vote by linear orders. A candidate is said to be a Condorcet winner (respectively, weak Condorcet winner), if that candidate is preferred to each other candidate by a strict majority (respectively, by at least half) of the voters. In Condorcet voting the winners are precisely the set of Condorcet winners. In the election system weakCondorcet, the winners are precisely the set of weak Condorcet winners. It has been known for two hundred years that some election instances have neither Condorcet winners
nor weak Condorcet winners [Con85]. And of course, no election instance can have more than one Condorcet winner, whereas there might be several weak Condorcet winners.

Let $\alpha$, $0 \leq \alpha \leq 1$, be a rational number. Copeland$^\alpha$ is the election system where for each pair of distinct candidates we see who is preferred between the two by a strict majority of the voters ([Cop51] for $\alpha = \frac{1}{2}$; [FHHR09a] for general rational $\alpha$). That one gets one “Copeland point” from the pairwise contest and the other gets zero “Copeland points.” If they tie in their pairwise contest (which can happen only when the number of voters is even), each gets $\alpha$ points. Copeland$^1$ is known as Llull elections, a system defined by the mystic Ramon Llull in the thirteenth century, [HP01], and is known to be remarkably resistant, computationally, to bribery and control attacks ([FHHR09a], although see also [ENR09, ER10, EPR11, Men13] for different highly resistant systems and see [HHR09] regarding how extremely resistant artificial systems can be constructed).

An important class of elections, which we will focus on in Section 6, is the class of scoring protocols. Each scoring protocol has a fixed number $m$ of candidates and is defined by a scoring vector $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m) \in \mathbb{N}^m$, $\alpha_1 \geq \alpha_2 \geq \ldots \geq \alpha_m$. Voters’ votes are linear orders, and each voter contributes $\alpha_1$ points to his or her most preferred candidate, $\alpha_2$ points to his or her next most preferred candidate, and so on. Each candidate whose total number of points is at least as great as the totals of each other candidate is a winner. For example, $m$-candidate plurality voting is the scoring protocol defined by the scoring vector $\alpha = (1, 0, \ldots, 0)$. And $m$-candidate Borda voting is the scoring protocol defined by the scoring vector $\alpha = (m - 1, m - 2, \ldots, 0)$.

In Black elections (respectively, weakBlack elections), if there is a Condorcet winner (respectively, if there are weak Condorcet winners), then that defines the winners, and otherwise Borda’s method is used to select the winners. Black elections were introduced in [Bla58] and weakBlack elections (somewhat confusingly called Black elections there) were introduced by Fishburn [Fis77]. In Dodgson elections (respectively, weakDodgson elections), whichever candidates can by the fewest repeated transpositions of adjacent candidates in voters’ orders become Condorcet winners (respectively, weak Condorcet winners) are the winners. Dodgson elections were introduced in the 1800s by Dodgson [Dod76] and weakDodgson elections (somewhat confusingly called Dodgson elections there) were introduced by Fishburn [Fis77] and further studied by McCabe-Dansted, Pritchard and Slinko [MPS08]. In Young elections (respectively, strongYoung elections), whichever candidates can by the deletion of the fewest voters become weak Condorcet (respectively, Condorcet) winners are the winners. Young elections were introduced by Young [You77] and strong-Young elections (somewhat confusingly called Young elections there) were introduced by Rothe, Spakowski, and Vogel [RSV03].
An important notion in this paper is that of being weakCondorcet-consistent. An election system is said to be weakCondorcet-consistent (which we earlier wrote, equivalently, as weak-Condorcet consistent), if on every input that has at least one weak Condorcet winner, the winners of the election system are exactly the set of weak Condorcet winners. Some of our bribery results will hold for all election systems that are weakCondorcet-consistent, and even for all election systems that when restricted to single-peaked electorates are weakCondorcet-consistent on those.

Fishburn [Fis77] has noted that the election systems weakBlack, weakDodgson, Fishburn, Maximin, and Young are weakCondorcet-consistent. We add to that the observation that Llull elections are easily seen from their definition to be weakCondorcet-consistent. We also make the (new) observation that the election systems Kemeny, Schwartz, and the two variants of Nanson due to Fishburn and Schwartz are weakCondorcet-consistent when restricted to single-peaked electorates. (By [Fis77] and [Nio87], those systems are known not to be weakCondorcet-consistent in the general case.) We also observe that Black, Dodgson, strongYoung, the original version of Nanson, and for each $\alpha$, $0 \leq \alpha < 1$, Copeland$^\alpha$ elections are not weakCondorcet-consistent even when restricted to single-peaked electorates. Appendix B includes definitions of the election systems Fishburn, Kemeny, Maximin (a.k.a. Simpson), and Nanson, and proves all of the new observations made in this paragraph.

**Single-Peaked Preferences** This paper’s theme is that combinatorial protections crumble for the case of single-peaked electorates. We now briefly define what single-peaked preferences are and what their motivation is.

The single-peaked preference model was introduced over half a century ago by Black [Bla48, Bla58] and has been influential ever since. The model captures the case where the electorate is polarized by a single issue or dimension, and each voter’s utility along that dimension has either one peak or just rises or just falls. Candidates have positions (locations) along that dimension. And a voter’s preferences (in the linear order model) simply order the candidates by utility (except with no ties allowed). Since the utility curves are very flexible, what this amounts to is that there is an overall societal ordering $L$ of the candidates, and each voter can be placed in some location such that for all the candidates to his or her right the preferences drop off and the same to the left, although within that framework, the right and the left candidates can be interspersed with each other. A picture will make this clearer. Figure 1 shows an electorate with four voters and five candidates, in which

---

2 The nomenclature in the literature is varied here. Some authors use the term “weak Condorcet-consistent” to mean systems that always select all weak Condorcet winners but perhaps have additional winners. And what we denote “weakCondorcet-consistent” is precisely what Fishburn [Fis77] calls “[obeying the] strict Condorcet principle.”
Figure 1: Example of a single-peaked electorate of four voters, with their utility functions shown.

society’s polarization is on a (liberal-to-conservative) axis. From the picture, we can see that $v_1$ has preferences $c_5 > c_4 > c_3 > c_2 > c_1$, $v_2$ has preferences $c_1 > c_2 > c_3 > c_4 > c_5$, $v_3$ has preferences (note the interleaving) $c_2 > c_3 > c_1 > c_4 > c_5$, and $v_4$ has preferences $c_4 > c_5 > c_3 > c_2 > c_1$.

Formally, there are many equivalent ways to capture this behavior, and we use the following as our definition. A collection $V$ of votes (each a linear ordering $>_i$ of the candidates) over candidate set $C$ is said to be single-peaked exactly if there exists a linear ordering $L$ over $C$ such that for each triple of candidates $a$, $b$, and $c$, it holds that $(a L b L c \lor c L b L a) \Rightarrow (\forall i)[a >_i b \Rightarrow b >_i c]$.

The single-peaked model has been intensely studied, and has both strengths and limitations. On the positive side, it is an excellent rough model for a wide range of elections. Votes on everything from American presidential elections to US Supreme Court votes to hiring votes within a CS department are often shockingly close to reflecting single-peaked preferences. It certainly is a vastly more reasonable model in most settings than is assuming that all voters are random and independent, although the latter model has been receiving a huge amount of study recently. In fact, a wide range of scholarly studies have argued for the value of the single-peaked model [Bla48, Bla58, DHO70, NW87, PR07, Kre98], and the model is one of the first taught to students in positive (i.e., theoretical) political science courses. On the other hand, some electorates certainly are driven by multidimensional concerns, and even a heavily unidimensional electorate may have a few out-of-the-box voters (e.g., voters who in an election polarized on a liberal-conservative axis decide their votes instead based on, for example, religion or race). Simply put, it is a model, and so it speaks of a simplified version of the world.

The single-peaked model also makes sense for approval voting [FHHR11]: There, a voter in-
tuitively may be thought to have some utility threshold starting at which he or she approves of candidates. What this means is that each voter’s “approved” candidates must be contiguous within society’s linear order $L$. Formally, we define this by saying that an election instance (of approval voters) is single-peaked exactly if there exists a linear order $L$ such that for each triple of candidates $a$, $b$, and $c$, we have $aLbLc \Rightarrow (\forall i) \left[\{a, c\} \subseteq \text{Approves}_i \Rightarrow b \in \text{Approves}_i\right]$, where $\text{Approves}_i$ is the set of candidates voter $i$ approves.

Although following the suggestion in Walsh’s [Wal07] seminal work we will assume (except when we make something else clear) that society’s linear order is part of the input in our single-peaked winner, bribery, manipulation, and control problems, we mention in passing that given an election instance, one can in polynomial time tell whether the voters are single-peaked and when so can also in polynomial time compute a societal linear order instantiating the single-peakedness ([BT86, DF94] for linear-order preferences and, as pointed out in [FHHR11, Theorem 2.1], [FG65, BL76] for approval preferences); one of course also can easily, in polynomial time, check whether a given linear order is one with respect to which a given set of votes is single-peaked.

Because we want to get to the results as quickly as possible, we will define the needed notions of winner, bribery, control, and manipulation each at the start of the section on the particular topic.

3 weakCondorcet Elections, Single-Peaked Electorates, and Bypassing Winner-Problem Complexity

The main results sections of this paper study whether single-peakedness bypasses complexity-theoretic protections against attacks on elections. Before moving to those sections, we quickly present some results showing that single-peakedness also bypasses the complexity results some systems have for even telling who won. Unlike the “protection from attack” complexity-shield bypassings, which are in some sense bad news (for the security of the election systems), these “winner-hardness” complexity-shield bypassings are good news—taming the complexity of election systems such as Dodgson and Kemeny for the single-peaked case, despite the fact that they are known to have NP-hard winner problems in the general case.

For a given election system $\mathcal{E}$, the winner problem takes as input an election, $\langle C, V \rangle$, and a candidate $p \in C$, and asks if $p$ is a winner in the election whose candidates are $C$ and whose votes are $V$ (where $V$ is a collection of votes over the candidate set $C$). When we speak of the single-peaked case of the winner problem, the input will also contain a linear order $L$ relative to which the election is single-peaked. (Formally, part of the winner-problem task is to check that that holds for $L$. But since that is a polynomial-time check for all cases—linear orders and approval vectors—
that we deal with, we will tacitly view the appropriateness of \( L \) as a “syntactic condition” on the input, although it is not really syntactic.) Note that the weakCondorcet winner problem is in \( P \) in the general case and thus certainly in the single-peaked case. Furthermore, something used often in our paper’s proofs is the following standard fact about Condorcet voting and medians.

**Fact 3.1.** Let us be given \((C, V)\) and \( L \) with respect to which \((C, V)\) is single-peaked, with votes being linear orders over \( C \). Associate each voter with the candidate at the top of that voter’s preference ordering. If we order the voters with respect to \( L \) in terms of that association, then if \( |V| \) is odd, the weakCondorcet and Condorcet winner set is the top preference of the “median” voter, and if \( |V| \) is even, the weakCondorcet winner set is the set of all candidates who in \( L \) fall in the range, inclusively, from the top preference of the leftmost of the two median voters through the top preference of the rightmost of the two median voters (and if those two coincide, then that candidate is a Condorcet winner and otherwise there is no Condorcet winner).

That of course is obvious. For example, if this is our ordered-by-\( L \) picture of the candidates and what the voters’ top choices are (we have renamed candidates and voters in \( L \)-location top-choice order for simplicity):

```
c_1 c_2 c_3 c_4 c_5 c_6
v_1 v_2 v_3 v_4 v_5 v_6
```

then there is no Condorcet winner and the weak Condorcet winners are \( c_2, c_3, \) and \( c_4 \).

An immediate consequence is the well-known fact that for single-peaked elections, there is always at least one weak Condorcet winner (we are tacitly here assuming \( C \neq \emptyset \)). Since we earlier noted that the winner problem is in \( P \) for weakCondorcet elections, the following holds.

**Theorem 3.2.** For each election system \( \& \) that is weakCondorcet-consistent when restricted to single-peaked electorates, the winner problem is in \( P \) when restricted to single-peaked elections.

Of course, for many such systems the winner problem is obviously in \( P \) even in general. Yet we do get some interesting consequences from Theorem 3.2 such as the following (keeping in mind our earlier claims about the behaviors of various systems).

**Corollary 3.3.** When restricted to single-peaked electorates, the winner problems for Kemeny, Young, and weakDodgson elections are in \( P \).
In contrast, the general-case Kemeny winner problem problem was proven by Hemaspaandra, Spakowski, and Vogel [HSV05] to be $\Theta_2^p$-complete.\textsuperscript{3} And we prove in this paper that the general-case winner problems for Young and weakDodgson elections are $\Theta_2^p$-complete as well (see Theorems B.2 and B.4 in the appendix). So, Theorem 3.2 implies sharp complexity simplifications for these three election systems.

The “identify with weakCondorcet” approach that just worked on Young and weakDodgson elections does not apply to Dodgson and strongYoung elections. However, we have constructed direct algorithms that solve their winner problems in polynomial time in the single-peaked case.

**Theorem 3.4.** When restricted to single-peaked electorates, the winner problems for Dodgson and strongYoung elections are in $P$.

Both claims in Theorem 3.4 contrast directly with the known $\Theta_2^p$-completeness of the general case Dodgson [HHR97] and strongYoung [RSV03] winner problems, and thus reflect a substantial complexity simplification that holds when electorates are single-peaked.

Although Theorem 3.4 is about winners rather than about bribery/manipulation/control protections, its proof is a good, simple example of this paper’s theme that single-peakedness tames combinatorial explosions. Taking Dodgson as an example: In the general case (not necessarily single-peaked votes), the set of “paths” to potentially implement best Dodgson scores is combinatorially explosive (to the best of current knowledge). In contrast, in the single-peaked case in searching for “paths” to implement best Dodgson scores it turns out we can restrict ourselves to changing just two voters in a particularly simple way that yields a polynomial-sized set of options in our search space.

Our appendix of spotlight proofs contains one particularly representative, interesting, or important proof from each section of the paper, and for Section 3, that proof is the proof of Theorem 3.4. We will typically in our main text describe briefly why each spotlight proof reflects a taming of combinatorial explosion (when it does), as we for example did in the above paragraph.

### 4 Bribery of Single-Peaked Elections

This section shows that single-peakedness undercut many, although not all, NP-hardness protections for bribery problems.

\textsuperscript{3}$\Theta_2^p$ is the class of sets that can be solved through polynomial-time parallel access to NP [PZ83, Hem89]. Throughout this paper, completeness always refers to completeness with respect to polynomial-time many-one reductions.
4.1 Notions

All bribery notions presented here, except negative approval bribery, are from the paper that started the complexity-theoretic study of bribery [FHH09]. Given an election system $\mathcal{E}$, the $\mathcal{E}$-bribery problem takes as input $C$, $V$, $p \in C$, and $k \in \{0, 1, 2, \ldots\}$, and asks if, by changing the votes of at most $k$ members of $V$, $p$ can be made a winner of this election with respect to $\mathcal{E}$. That is the basic bribery problem. And it can be modified by any combination of the following items. “$\$$” means each voter has a price (belonging to $\{1, 2, 3, \ldots\}$) and the question is whether there is some set of voters whose total price is at most $k$ such that by changing their votes we can make $p$ be a winner. The intuition for prices is that some voters can be swayed more easily than others. “Weighted” means each vote has a weight (belonging to $\{1, 2, 3, \ldots\}$), and each weight $w$ vote is bribed as an indivisible object, but when applying $\mathcal{E}$, is viewed as $w$ identical “regular” votes. The intuition for weights is that in some elections—e.g., by stockholders—voters have differing weights.

For the case where $V$ consists of linear orders, by “negative” we mean that if we bribe a voter then after we bribe the voter the voter must not have $p$ as his or her top choice unless $p$ already was the top choice before the bribe.\footnote{The [FHH09] definition of negative bribery more naturally can be read to have the quite different semantics that “each bribed voter must, after the bribe, not have $p$ as his or her top choice.” Since [FHH09] used negative bribery only for plurality, this issue made no difference in [FHH09], and indeed since we look at negative bribery for linear orders here mostly with respect to weakCondorcet in single-peaked contexts, it is not a key issue here either. But we have switched to a definition that captures a more attractive notion: You cannot directly boost your preferred candidate $p$ to the top, but for votes where $p$ is already at the top you can shift the remaining preferences. The distinction between these two approaches to “negative” does change proofs for the case of approval voting, and so for that we will give separate definitions that capture each notion.} The intuition is that in negative bribery one is trying to stay under the radar by not directly helping one’s candidate. For voting by approval vectors, we give definitions to capture both the analog of the linear-preference negative notion we just defined (“negative”) and of the one one would get by taking [FHH09] utterly literally (“strong negative”)—see Footnote 4 for more background. For approval-vector votes, by “negative” we mean that when you bribe a voter, his or her after-bribe vector can approve $p$ only if his or her before-bribe vector approved $p$. By “strongnegative” we mean that when you bribe a voter the voter after being bribed cannot approve $p$.

The notions described above can occur in any combination, e.g., we can speak of Llull-negative-weighted-$\$$bribery.

When we speak of the single-peaked case of any of the above, we mean the electorate is single-peaked, and an $L$ relative to which the votes are single-peaked is itself part of the input. Further, all bribes must result only in votes that are consistent with the input societal order $L$. Taking $L$ as part of the input, and as binding the legal bribes, is the natural bribery analog of the manipulation model of
Walsh [Wal07] and Faliszewski et al. [FHHR11]. Binding the bribes to respect $L$ is natural, e.g., if $L$ is widely known, the central authority may simply reject (as obviously manipulative votes) votes that violate $L$. But although this is our core model, many of our results carry over to models more flexible on these points, and we will at times point that out.

We will often speak about bribery (or, later, manipulation and control) problems being “in P” (or “in polynomial time”). Although formally that just asserts that a P algorithm exists to say whether a successful bribery (or manipulation or control) action exists, in fact in every such instance we in our proof show how to in addition obtain an actual, successful bribery (or manipulation or control) action when one exists.

### 4.2 Approval-Bribery Results

As our spotlight result for approval-bribery, we will prove that the bribery protection that complexity gives there fails on single-peaked electorates.

**Theorem 4.1** ([FHH09]). Approval-bribery is NP-complete.

**Theorem 4.2.** Approval-bribery is in P for single-peaked electorates.\(^5\)

Theorem 4.2 is proven in the “spotlight proof” appendix. But since it is one of our spotlight results, we now will informally explain what key challenge (namely, “incomparability”) exists for this proof and how our proof overcomes that challenge (namely, by using “directionality”).

So, recall that in approval bribery in the single-peaked setting, the societal order, $L$, is part of the input and each voter approves of some (possibly empty) set of candidates that are contiguous with respect to $L$. Suppose the input linear order $L$ is $c_1Lc_2Lc_3L\ldots Lc_{100}$ with respect to which the society is single-peaked. Suppose the candidate the briber is trying to make win is $c_{25}$. Suppose that the input limit on the number of people the briber can bribe is 2009 and suppose our input election has 5000 voters of which 3000 (call them $V_-$) initially do not approve of $c_{25}$ and 2000 (call them $V_+$) initially do approve of $c_{25}$. Now, clearly, we will not spend any of our 2009 bribes on voters from $V_+$, as those voters already approve of $c_{25}$, and so bribing a voter from $V_+$ is never better than bribing a voter from $V_-$. So, our goal is to seek a good set of 2009 voters from $V_-$, if such exists.

The key challenge, even given single-peakedness, can be stated in a word: incomparability. That is, given that we know that the number of approvals for $c_{25}$ will go up by exactly 2009 after the

---

\(^5\)This result holds both in the model where $L$ is part of the input and the model in which we must find an $L$ consistent with the input and relative to which bribery is possible. The result also holds even in the model—not our core model for which we will prove it—in which the bribed voters need not respect the societal order after being bribed.
bribe, and given we know the total number of approvals each other candidate gets before the bribe, we for each candidate $c_i$ other than $c_{25}$ have a target number $n_i$ such that among the 2009 votes we choose to bribe from $V_-$, at least $n_i$ must initially have approved of $c_i$ (in order for $c_{25}$ to beat $c_i$ after the bribe).

Now, here is the rub. Consider two voters from $V_-$, one of whom approves of $c_{30}$ through $c_{55}$ and the other of whom approves of $c_{40}$ through $c_{80}$. Among these two votes, only the former helps us address positive $n_i$ values for $30 \leq i \leq 39$, and only the latter helps us address positive $n_i$ values for $56 \leq i \leq 80$. Since neither voter’s approval set contains the other’s, they offer differing advantages, and neither is, at first glance, obviously one we “should” include in our 2009. And in fact the 3000 voters of $V_-$ are a thicket of such incomparabilities. Indeed, trying to find a subset of size 2009 (in this particular example, the number will vary with the input) feels very much like a covering problem, which in fact is exactly the path by which the general case was proven NP-hard by [FHH09].

However, we will use single-peakedness to tame the combinatorial challenge of choosing a good subset. We in particular will use single-peakedness to induce a directional attack that will locally make incomparability disappear at each moment at which we need to make a decision about choosing some of the 2009. (Although the focus here is bribery, and our construction and arguments are tailored to that, we mention that our line of attack is modeled on what [FHHR11] calls “smart greedy” algorithms and used to study control attacks.) Our full proof is in the appendix, but the key idea can be easily conveyed if the reader will be so kind as to visualize along with us for a moment. Consider the largest $i$ such that $n_i > 0$, and (for the purpose of this example) suppose that $i$ is strictly greater than 25—let us say that it is 93, and suppose $n_{93} = 3$. Then we must include in the 2009 at least three members of $V_-$ who approve of $c_{93}$. But which three? Isn’t incomparability still a problem? No! Since we chose the largest $i$ for which $n_i > 0$, and (for the purpose of this example) suppose that $i$ is strictly greater than 25—let us say that it is 93, and suppose $n_{93} = 3$. Then we must include in the 2009 at least three members of $V_-$ who approve of $c_{93}$. But which three? Isn’t incomparability still a problem? No! Since we chose the largest $i$ for which $n_i > 0$, clearly $n_{94} = \ldots = n_{100} = 0$. So although among the voters in $V_-$ who approve of $c_{93}$ there may be incomparability of approval ranges, range differences to the right of $c_{93}$ are utterly irrelevant, as $c_{25}$ is already beating all those candidates anyway. The only interesting issue is, among the voters of $V_-$ who approve of $c_{93}$, what their leftmost approved candidate is—the more leftward the better as that will help on the most possible $n_i$ deficits. But now we can do direct comparisons and take action: We will put into our 2009 the three voters (among the $V_-$ voters who initially approve of $c_{93}$) whose approval range is the leftmost. (If $V_-$ lacks three voters who approve $c_{93}$, then since $n_{93} = 3$, successful bribery is impossible.) And the process now continues as one would expect: Based on those three votes, all $n_i$ values are updated and the next leftmost $n_i > 0$ satisfying $i \geq 26$ is similarly handled, and so until all $n_i > 0$ with $i \geq 26$ are handled, and then starting from the other end of $L$ we will analogously handle
If we neutralize all $n_i > 0$ within 2009 bribes among $V_-$, we have a successful bribery, and otherwise none is possible. This concludes our example of how single-peakedness creates a directionality that tames the rich covering problem caused by incomparability. (In fact, this “example” is essentially (if one removes the particular integer values we used) a complete proof for the case that $k \leq ||V_-||$. And note that if $k \geq ||V_-||$, we can always make $p$ a winner by bribing all voters from $V_-$, since after the bribe all voters approve of $p$. Appendix A does contain a more careful exposition of the process, but a reader comfortable with the somewhat informal presentation just given may wish to at least initially simply move on within the main (nonappendix) text rather than reading now that particular spotlight proof.)

By the same general approach—using a “directional” attack to in the single-peaked setting tame the incomparability challenges of covering problems—we can establish the following two additional cases in which NP-hard bribery problems fall to P for the single-peaked case.

**Theorem 4.3.** The following hold:\(^6\)

1. Approval-negative-bribery and approval-strongnegative-bribery are NP-complete.

2. For single-peaked electorates, approval-negative-bribery and approval-strongnegative-bribery are in P.

Faliszewski et al. [FHH11a] (see also [FHH11b]) have observed that the constructions in our proofs of Theorem 4.2 and of Theorem 4.3 (part 2) can be used to show that those results hold even in the more flexible model in which up to a logarithmic number of voters can violate the societal linear order.

### 4.3 Llull-Bribery and Kemeny-Bribery Results: weakCondorcet consistency

We now state the following eight-case result. The P cases below are proved by direct algorithmic attacks using the connection between weakCondorcet and “median” voters, and the NP-complete cases are shown by using the problem to capture a partition instance.

---

\(^6\)The claim is in our standard model: nonunique-winner model (i.e., we ask if the preferred candidate $p$ can be made to be a winner); the societal order $L$ is given as part of the input; and bribed voters must still respect $L$. However, we note in passing that the claim still holds with any of the choices one can make regarding: nonunique-winner model vs. unique-winner model; $L$ is part of the input vs. we are asking whether there exists a valid $L$ with respect to which a successful bribery can be accomplished; and the bribed voters respect $L$ model vs. the model in which the bribed voters may violate $L$. Seeing that the result holds in these various alternate models requires natural modifications of the proof in some cases (e.g., nonunique vs. unique), and requires taking advantage of specific properties of the construction in other cases (e.g., regarding allowing bribed voters to violate $L$, the constructions actually only bribe voters to end up approving zero or one candidate, and such votes are consistent with every ordering).
Theorem 4.4. For single-peaked electorates, weakCondorcet-weighted-bribery, weakCondorcet-negative-weighted-bribery, and weakCondorcet-negative-weighted-bribery are NP-complete, and the remaining five cases (weakCondorcet-bribery, weakCondorcet-$\$bribery, weakCondorcet-weighted-bribery, weakCondorcet-negative-bribery, weakCondorcet-negative-$\$bribery) are in P.

Theorem 4.4 is most interesting not for what it says about weakCondorcet elections, but for its immediate consequences on other election systems.

Corollary 4.5. Let $\mathcal{E}$ be any election system that is weakCondorcet-consistent or that is merely weakCondorcet-consistent on single-peaked inputs. Then the three NP-completeness and five P results of Theorem 4.4 hold (for single-peaked electorates) for $\mathcal{E}$.

From our discussions earlier in the paper, Corollary 4.5 applies to the Llull, Kemeny, Young, weakDodgson, Maximin, Schwartz, weakBlack, Fishburn, and the two variants of Nanson due to Fishburn and Schwartz. In light of this, Corollary 4.5 is quietly establishing a large number of claims about NP-hardness shields failing for single-peaked electorates. For example, we have the following claims.

Theorem 4.6 ([FHH09]). Llull-bribery, Llull-$\$bribery, Llull-weighted-bribery, and Llull-weighted-$\$bribery are each NP-complete.

Theorem 4.7 (follows from Corollary 4.5). For single-peaked electorates, Llull-bribery, Llull-$\$bribery, Llull-weighted-bribery, and Llull-weighted-$\$bribery are each in P.

To the best of our knowledge, bribery of Kemeny elections has never been studied. Note, however, that the winner problem for any election system $\mathcal{E}$ many-one reduces to each of the eight types of bribery problems mentioned in Theorem 4.4, except with “weakCondorcet” replaced by “$\mathcal{E}$.” This holds because we can ask whether the preferred candidate wins given the bribe limit of 0, and this captures the winner problem. So, from the known $\Theta^p_2$-completeness of the winner problem for Kemeny elections [HSV05], we have the following result, which gives us eight contrasts of hardness (three between $\Theta^p_2$-hardness and NP membership and five between $\Theta^p_2$-hardness and P membership).

Theorem 4.8 (corollary, in light of the comments just made, to [HSV05]). For Kemeny elections, all eight types of bribery mentioned in Theorem 4.4 are $\Theta^p_2$-hard.

Theorem 4.9 (follows from Corollary 4.5). For single-peaked electorates, Kemeny-weighted-$\$bribery, Kemeny-negative-weighted-bribery, Kemeny-negative-weighted-$\$bribery are NP-complete (and so in particular each belongs to NP). For single-peaked electorates, the remaining five types of bribery of Kemeny elections are in P.
As a final remark regarding Theorem 4.4, we note that even within the single-peaked cases that it studies, there is one twist, in which changing from bribery to negative bribery changes the complexity, namely, for single-peaked electorates, weakCondorcet-weighted-bribery is in P but weakCondorcet-negative-weighted-bribery is NP-complete. Here, decreasing the set of bribes available to the briber actually boosts the complexity of the briber’s task. (The explanation for this is, very loosely and intuitively speaking, that among the set of bribes that negativity removes from the search space are the set of bribes used in the P-time nonnegative case bribery attack.)

5 Control by Partition of Single-Peaked Electorates

The control problems for elections ask whether by various types of changes in an election’s structure a given candidate can be made a winner. The types of control that were introduced in 1992 by Bartholdi, Tovey, and Trick [BTT92], and that (give or take some slight refinements) have been studied in subsequent papers, are addition/deletion/partition of voters/candidates. However, to the best of our knowledge previous study of the complexity of control for single-peaked electorates (such as that in Faliszewski et al. [FHHR11]) focused exclusively on additions and deletions of candidates and voters.

We for the first time study the complexity of partition problems for the case of single-peaked electorates. And we show that for a broad range of election systems the control by partition of voters problem is in P for single-peaked electorates. Among the systems we do this for are Llull and Condorcet elections, whose control by partition of voters problem is known to be NP-complete for general electorates. Our proofs again work by using single-peakedness to tame combinatorial explosion—in this case, the number of partitions that must be examined is reduced from an exponential number of partitions to a polynomial number of classes of partitions each of which can be checked as a block.

5.1 Notions

The control by partition of voters problem for an election system $\mathcal{E}$ takes as input an election instance $(C, V)$ and a candidate $c \in C$ and asks whether there is a partition of the votes $(V_1, V_2)$ (i.e., $V_1 \cup V_2 = V$ and $V_1 \cap V_2 = \emptyset$) such that if the “appropriate candidates” move forward from the preliminary elections $(C, V_1)$ and $(C, V_2)$ to a final election in which those candidates are voted on by $V$, then $c$ “wins.” How one clarifies the quoted strings determines the precise type of voter control one studies. In particular, one can study the nonunique-winner model (the question is: can $c$ be made a winner of the final election?) or the unique-winner model (can $c$ be made to uniquely
win the final election?). And as to what the “appropriate candidates” move forward means, one can study the Ties Promote (TP) model (all winners of the preliminary election move forward) or the Ties Eliminate (TE) model (each subelection moves forward its unique winner if it has a unique winner, and otherwise moves no one forward). We find most natural for study the TP, nonunique-winner model. (TP and nonunique-winner pair naturally, as do TE and unique-winner.) However, since the seminal control paper of Bartholdi, Tovey, and Trick [BTT92] used the unique-winner model, we have checked that our results hold for all four model combinations. We will henceforth refer to these models (TP and unique-winner, TP and nonunique winner, TE and unique-winner, TE and nonunique-winner) as the four standard models.

Partition of voters models the case where the partitioned electorate has primaries. An example from [FHR09a] is a business group where a powerful manager studies an issue by splitting his or her group into two task forces each to (by voting) recommend alternatives to be part of a final vote to be conducted by the entire group. This (loosely) corresponds to control by partition.

We will briefly mention control results about adding and deleting voters and candidates. The definitions of those are just what one would expect, and we refer the reader to [FHR09a] for those definitions.

When we speak about a control problem for single-peaked electorates, we will mean that the societal order \( L \) is part of the input. However, for the case of control, it turns out that “P” holds for that model if and only if “P” holds for the model in which one is not given an \( L \) as part of the input but rather one is asked whether there exists some linear order \( L \) relative to which the entire input (even including all potentially added candidates and voters) is single-peaked and is such that the goal can be achieved by the given control action. This claim is formalized as Theorem B.5 and proven in Appendix B. In light of Theorem B.5, we will simply assume all our control results are in the default model (societal order is part of the input).

### 5.2 Control Results Related to weakCondorcet Elections

In this section we present our control results, with a focus on control by partition of voters. We will see that although Llull and Condorcet elections have NP-hard voter-partition control problems, those problems fall to polynomial time for single-peaked electorates. Our spotlight result for this section states that partition-by-voters control for weakCondorcet elections is in P.

**Theorem 5.1.** For weakCondorcet elections, (constructive) control by partition of voters in all four standard models is in P for single-peaked electorates.
Before describing the flavor of Theorem 5.1’s proof, let us note some consequences and contrasts.

**Corollary 5.2.** Let $\mathcal{E}$ be any election system that is weakCondorcet-consistent on single-peaked inputs. Then for election system $\mathcal{E}$, (constructive) control by partition of voters in all four standard models is in $P$ for single-peaked electorates. In particular, this holds for the election systems Llull, Kemeny, Young, weakDodgson, Maximin, Schwartz, weakBlack, Fishburn, and the two variants of Nanson due to Fishburn and Schwartz.

For Llull elections, this provides a clear contrast with the known NP-completeness for that same control type in the general case.

**Theorem 5.3 ([FHHR09a]).** For Llull elections, (constructive) control by partition of voters in all four standard models is NP-complete.

The “spotlight proofs” appendix contains a full proof of Theorem 5.1. But let us right now try to convey the basic idea behind the proof for the TP, nonunique-winner model. This idea differs completely from the approach used in the P-time control proofs in [FHHR11], and is, we think, novel.

Suppose the candidates in our input election have societal single-peaked order $L$. And suppose our goal is to find a partition making $p$ win overall. Let us here completely ignore the extreme partitions $(\emptyset, V)$ and $(V, \emptyset)$, although the appendix’s proof will handle those. Now, each part of the partition we make will yield some winners, namely, all candidates who in $L$ are in the range of candidates who are or are between (in $L$) the first-choice candidates of the voter (or two voters when $|V|$ is even) that are the “median” voters (when one sorts the voters with respect to where their first choices fall in $L$, breaking ties arbitrarily).

Now, our approach will be to look over all intervals of candidates that might be the winners from $V_1$ (of a hypothetical partition $(V_1, V_2)$) and over all intervals of candidates that might be the winners from $V_2$. That is, we will loop over all $a, b, c, d \in C$ such that $a \leq_L b$ and $c \leq_L d$ (writing $x \leq_L y$ for $(x = y$ or $xLy)$). This loop will have at most $|C|^4$ values to go through. For each $a, b, c, d$ of this loop, we will check whether, if the $(C, V_1)$ winners are $\{e \in C \mid a \leq_L e \leq_L b\}$ and the $(C, V_2)$ winners are $\{e \in C \mid c \leq_L e \leq_L d\}$, then $p$ is a winner of the final election $\{e \in C \mid a \leq_L e \leq_L b \lor c \leq_L e \leq_L d\}, V)$. We can do this check without knowing $(V_1, V_2)$. If $p$ is not a winner, we go to the next iteration of the $a, b, c, d$ loop. However, if $p$ is a winner of the final election that $a, b, c, d$ would induce, then we must see whether there exists some partition $(V_1, V_2)$ that has precisely the two first-round victory sets we just described, as that will give us a successful control attack. We
could do that by going through all partitions. But there are an exponential number of them. So instead, we will do something to exploit the limitations of single-peaked electorates. Namely, we will loop through all sizes of $V_1$, i.e., $\|V_1\| = 1, \|V_1\| = 2, \ldots, \|V_1\| = \|V\| - 1$. ($\|V_1\| \in \{0, \|V\|\}$ is in concept legal, and our actual appendix proof will handle that case, but let us ignore it here.) So, for a given iteration through the “size of $V_1$” loop, there still might be an exponential number of partitions (e.g., when $\|V_1\| \approx \|V\|/2$ the number of partitions is huge).

But our core trick is that we do not have to look individually at all of them. Rather, we only have to divide the $\|V_1\|$ we are looping over into a small (and globally upper-bounded by a constant) number of parts. The actual number of parts will depend on the relation of $a, b, c,$ and $d$ to each other. But as an example, we will here talk our way through a case that is the “worst” in terms of number of interesting parts whose size we need to determine. In particular, let us consider the case where $a <_L b <_L c <_L d$. (This is just one among many cases; see the full proof in the appendix.) So, the picture we have is:

In this picture, we have partitioned the candidates into nine regions, $R_1$ through $R_9$. Now, in our particular loop iteration, we already know the $\|V_1\|$ we are handling, call it $\hat{n}$. So, for each voter in $V$, we must decide whether to put him or her in $V_1$ or $V_2$. But since going voter by voter would be too expensive, we go region by region, because within each region, the only thing that affects the winner set is the number of voters we put into $V_1$, not which voters we use to achieve that number.

Now at first glance, we might think that we have to look at all ways to divide $\hat{n}$ into nine ordered numbers that add up to $\hat{n}$ (e.g., $490 = 1 + 100 + 50 + 251 + 80 + 0 + 4 + 4 + 0$). And that would be fine, since there are $O(\|V\|^8)$ such ways. But looking at the picture, we can see things that help us further. Clearly, since the winners in $V_2$ are the interval of candidates from $c$ through $d$, $V_2$ must have its median voters on top of $c$ and $d$ and must have no voters whose top choice is in region $R_7$. So every voter whose top choice is in $R_7$ must go into $V_1$ and, similarly, no voters whose top choice is in region $R_3$ go into $V_1$. So, there are only seven regions whose size is not totally “forced,” and so there are at most $O(\|V\|^5)$ divisions we must try. (In the full proof, we of course handle all cases, e.g., not just the one pictured case, but it turns out that each case has at very most $O(\|V\|^5)$ divisions.)

So, we have just sketched a polynomial-time algorithm that finds a successful control partition when one exists. And the algorithm’s theme perfectly supports the theme of this paper: The
algorithm used single-peakedness to bypass the combinatorial richness of partitions.

Before we turn from weakCondorcet to Condorcet elections, we state a result that will quickly give us a number of additional contrasts between general-case control complexity and single-peaked control complexity.

**Theorem 5.4.** For weakCondorcet elections, (constructive) control by adding voters and (constructive) control by deleting voters are each in \( P \) for single-peaked electorates, both in the nonunique-winner model and in the unique-winner model.

As usual, it immediately follows that the above result applies to our standard long list of systems (e.g., Kemeny, Young, and weakDodgson elections) that are weakCondorcet-consistent on single-peaked electorates. Appendix B contains similar results for Condorcet elections. However, the winner problem in the general case trivially many-one polynomial-time reduces to control by adding voters (via asking if \( p \) can be made to win by adding 0 voters; see [HHR09, Section 2.4]). Thus, the existing \( \Theta_2^p \)-hardness results for the Kemeny winner problem (both in the nonunique-winner model [HSV05] and in the unique-winner model [HHR09]), the Young winner problem (both in the nonunique-winner model and in the unique-winner model, see this paper’s Theorem B.2), and the weakDodgson winner problem (both in the nonunique-winner model and in the unique-winner model, see this paper’s Theorem B.4), themselves imply that control by adding voters is \( \Theta_2^p \)-hard for Kemeny, Young, and weakDodgson elections (in both the nonunique-winner model and the unique-winner model). The same comments hold for control by deleting voters. Thus the single-peaked and general cases of control by adding and deleting voters differ for Kemeny, Young, and weakDodgson elections.

### 5.3 Control Results Related to Condorcet Elections

Control of Condorcet elections has been studied in much detail [BTT92, HHR07], and (see Table 1 of [HHR07]) each standard control type is known to either never change the outcome at all or to have a polynomial-time algorithm, with three exceptions. Those three exceptions are that Bartholdi, Tovey, and Trick in their seminal paper on control [BTT92] proved that control by addition of voters, control by deletion of voters, and control by partition of voters are all \( NP \)-complete for Condorcet elections. (Their entire paper is in the unique-winner model, and so all the above discussion is in the unique-winner model. And we will thus need to establish our “contrasting” P-time results in the unique-winner model if we want a meaningful contrast. To address this, we will ensure that our contrasting results hold in both models. But that holds trivially if we prove it in either model, as for Condorcet elections, nonunique-winner and unique-winner coincide and Ties Promote and
Ties Eliminate coincide—both because one can never have two or more winners.) However, the following results show that these resistance results vanish for single-peaked electorates.

**Theorem 5.5.** (Constructive) control of Condorcet elections by partition of voters is in polynomial time for single-peaked electorates (in both the nonunique-winner model and the unique-winner model, and in both the Ties Promote model and the Ties Eliminate model—although all four cases coincide here).

**Theorem 5.6.** (Constructive) control by adding voters and (constructive) control by deleting voters is in polynomial time for Condorcet elections over single-peaked electorates (in both the nonunique-winner model and the unique-winner model).

So, for all the standard control cases that Condorcet voting is known to be NP-hard for in the general case, we have shown that Condorcet-voting control falls to polynomial time for single-peaked electorates.\(^7\)

### 6 Manipulation of Single-Peaked Electorates: Dichotomy for Constructive Coalition Weighted Manipulation

Faliszewski et al. [FHHR11] completely characterized, for three-candidate elections, which scoring protocols have polynomial-time constructive coalition weighted manipulation problems and which have NP-complete constructive coalition weighted manipulation problems. We achieve a far more sweeping dichotomy theorem—our result applies to all scoring protocols, regardless of the number of candidates. In the constructive coalition weighted manipulation problem, the input is the candidate set \(C\), the nonmanipulative voters (each a preference order over \(C\) and a weight), the manipulative voters (each just a weight), and a candidate \(p \in C\), and the question is whether there is a way of setting the preferences of the manipulative voters such that \(p\) is a winner under the given election rule when all the manipulative and nonmanipulative voters vote in a weighted election.

\(^7\)The sharp-eyed reader may wonder whether it is in concept possible that some of the general-case polynomial-time results for control (e.g., Condorcet control by deleting candidates) might suddenly, freakishly get harder in the single-peaked case. After all, [FHHR11] shows a freakish case where limiting to the single-peaked case increases manipulation complexity. However, it is not hard to see—by reasoning related to that used to prove Theorem B.5—that if a control type is in polynomial time in the general case (i.e., in technical terminology (see [HHR07]) “immune” or “vulnerable”) then in the single-peaked case it remains in polynomial time (in technical terminology, if it was immune it remains immune, and if it was vulnerable, it either becomes immune or remains vulnerable).
Theorem 6.1 ([FHHR11]). Consider a three-candidate scoring protocol, namely, \( \alpha = (\alpha_1, \alpha_2, \alpha_3), \alpha_1 \geq \alpha_2 \geq \alpha_3, \alpha_1 \in \mathbb{N}, \alpha_2 \in \mathbb{N}, \alpha_3 \in \mathbb{N}. \) For the single-peaked case, the constructive coalition weighted manipulation problem (in both the nonunique-winner model and the unique-winner model) is NP-complete when \( \alpha_1 - \alpha_3 > 2(\alpha_2 - \alpha_3) > 0 \) and is in polynomial time otherwise.

Our extension of this three-candidate, single-peaked electorate result to the case of any scoring protocol over single-peaked electorates is somewhat complicated. Yet, since it is a complete characterization—a dichotomization of the complexities, in fact—it is in some sense simply reflecting the subtlety and complexity of scoring systems. (For the general—i.e., not necessarily single-peaked—case, the known characterization is simple regardless of the number of candidates: NP-completeness holds when \( ||C|| \geq 2 \) and \( \alpha_2 \neq \alpha_{||C||} \) and otherwise the problem is in P ([HH07], see also [CSL07, PR07]).)

Theorem 6.2. Consider an \( m \)-candidate scoring protocol, namely, \( \alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m) \in \mathbb{N}^m, \alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_m. \)

- If \( m \geq 2 \) and \( \alpha_2 > \alpha_{\lceil \frac{m-1}{2} \rceil + 2} \) and there exist integers \( i, j > 1 \) such that \( i + j \leq m + 1 \) and \( (\alpha_1 - \alpha_i)(\alpha_1 - \alpha_j) > (\alpha_i - \alpha_{i+1})(\alpha_j - \alpha_{j+1}) \), then the constructive coalition weighted manipulation problem for the single-peaked case is NP-complete.

- If \( m \geq 2 \) and \( \alpha_2 = \alpha_{\lceil \frac{m-1}{2} \rceil + 2} \) and \( \alpha_1 > \alpha_2 > \alpha_m \) and \( (\alpha_2 > \alpha_{m-1} \text{ or } \alpha_1 - \alpha_m > 2(\alpha_2 - \alpha_m)) \), then the constructive coalition weighted manipulation problem for the single-peaked case is NP-complete.

- In all other cases, the constructive coalition weighted manipulation problem for the single-peaked case is in P.

For example, the constructive coalition weighted manipulation for the single-peaked case for \( m \)-candidate plurality and \( m \)-candidate veto is in P, and for \( m \)-candidate Borda it is in P if \( m \leq 3 \) and NP-complete otherwise.

The proof of Theorem 6.2 appears in the “spotlight” appendix. Note that Theorem 6.1 follows from Theorem 6.2, since \( \alpha_1 - \alpha_3 > 2(\alpha_2 - \alpha_3) \) is equivalent to \( \alpha_1 - \alpha_2 > \alpha_2 - \alpha_3 \). We also note that the specific cases of more-than-three-candidate scoring protocols—such as four-candidate Borda and \( m \)-candidate veto—that were analyzed for the single-peaked case in [FHHR11] yielded results completely consistent with Theorem 6.2’s characterization. The “P” cases of Theorem 6.2’s dichotomy align with our theme of single-peakedness often foiling combinatorial protections.
7 Related Work, Additional Discussion, and Open Problems

The two papers most related to our work are Walsh [Wal07] and Faliszewski et al. [FHHR11]. Walsh’s [Wal07] paper first raised (among many other interesting issues, such as possible and necessary winners [KL05] in single-peaked settings) the issue of the effect of single-peaked electorates on manipulation, and for the particular case he looked at—weighted coalition manipulation under single transferable vote elections—he showed that manipulation remains hard even for single-peaked electorates. Faliszewski et al. [FHHR11] showed cases where single-peakedness removes complexity shields against manipulation, and also opened the study of (nonpartition) control. Our paper in contrast with Walsh’s stresses cases where single-peakedness removes combinatorial protections. And we go beyond Faliszewski et al. [FHHR11] by for the first time studying bribery of single-peaked electorates and partition-control of single-peaked electorates. For both these new cases, we show that many election systems (for example Llull elections) have polynomial-time algorithms for single-peaked electorates, even if the system is known to be NP-hard in the analogous general case. We also generalize the [FHHR11] dichotomy theorem for manipulation of three-candidate scoring protocols to the case of arbitrary scoring protocols. Although it is more closely related to Faliszewski et al. [FHHR11] than to the present paper, and came after the present paper, it is important to mention that Faliszewski et al. [FHH11b] explores the interesting issue of seeing whether results such as those of Faliszewski et al. [FHHR11] still hold even when the electorate is merely near to being single-peaked (see also [BCW13] and [ELP13] regarding nearness to single-peakedness).

Although [Wal07, FHHR11] are by far the most related work, other work is much worth mentioning. Bartholdi and Trick [BT86], Doignon and Falmagne [DF94], and Escoffier, Lang, and Öztürk [ELÖ08] provided efficient algorithms for finding single-peaked orderings. And Conitzer [Con09] studied the effect of single-peaked electorates on preference elicitation. Indeed, single-peakedness is of much current interest in computational settings. For example, at least four papers in the IJCAI-2013 conference, including the Bredereck et al. [BCW13] paper cited above, are related to single-peakedness.

The \( \Theta_2^p \)-completeness of the winner problems of Dodgson, Kemeny, and strong Young elections was established, respectively, by Hemaspaandra, Hemaspaandra, and Rothe [HHR97], Hemaspaandra, Spakowski, and Vogel [HSV05], and Rothe, Spakowski, and Vogel [RSV03]. The literature now contains many papers on the complexity (when single-peaked preferences are not assumed) of manipulation and control (as a pointer to some of those, see [FHHR09b, FHH10] and the citations therein), and even contains a number of papers on the younger topic of the complexity of bribery (e.g., [FHH09, Fal08, FHHR09a]). Although the nonunique-winner model and the unique-
winner model very typically have the same complexity results, Faliszewski, Hemaspaandra, and Schnoor [FHS08] (drawing also on Conitzer, Sandholm, Lang [CSL07]) show that this is not always the case.

NP-completeness and $\Theta^p_2$-completeness are worst-case notions. So it is natural to wonder whether problems in those classes can be solved frequently by heuristic algorithms. There has been much experimental study on that theme (see, e.g., [Wal09]). On the other hand, it is known (see [HW12]) that if any polynomial-time deterministic heuristic algorithm for any NP-complete (or $\Theta^p_2$-complete) problem asymptotically makes subexponentially many errors, then the polynomial hierarchy collapses.

A worry that comes immediate to the minds of social choice theorists can be expressed as follows: Since it is known that, for single-peaked electorates, “median voting” leaves voters with voting sincerely being an optimal strategy, single-peaked elections are not interesting in terms of other election systems, since median voting should be used. A detailed discussion of this worry would itself fill a paper. But we briefly mention why the above objection is not as serious as it might at first seem. First, the nonmanipulability claims regarding single-peaked elections and median voting are about manipulability, and so say nothing at all about, for example, control. Indeed, weakCondorcet in effect is a type of median voting on single-peaked electorates, and so for example our partition of voters algorithm makes it clear that control can be exercised in interesting ways. Second, even if median voting does have nice properties, the simple truth is that in the real world, society—for virtually all elections and electorates—has chosen (perhaps due to transparency, comfort, or tradition) to use voting systems that clash sharply with median voting. The prominence of plurality voting is the most dramatic such case. So since in the real world we do use a rich range of election systems, it does make sense to understand their behavior. Third, one must be very careful with terms such as “strategy-proof.” The paper people most commonly mention as showing that median voting is strategy-proof is Barberà [Bar01]. But that paper’s results are about “social choice functions” (election rules that when $|C| \geq 1$ always have exactly one winner), not—as this paper is—about election rules that select a set of winners. So one cannot simply assume that for our case median voting (say, weakCondorcet elections) never gives an incentive to misrepresent preferences.

Actually, it is true that for many nice notions one can show that one indeed never has an incentive to misrepresent one’s top choice (in single-peaked weakCondorcet elections one’s top choice is all that affects the outcome) in weakCondorcet elections (which are a social choice correspondence).

---

8To be more explicit, under the most natural way of framing median voting, median voting and weakCondorcet (and so all weakCondorcet-consistent rules) are exactly the same on single-peaked electorates. And this paper establishes many results regarding bribery and control of weakCondorcet in the context of single-peaked electorates.
For example, if one’s goal is “Seek to make your top choice be a weakCondorcet winner,” then one never has an incentive to misstate one’s top choice. As another example, if one’s goal is (for any fixed $k$) “Seek to make at least one of your first $k$ choices be among the weakCondorcet winners,” then again one never has an incentive to misstate one’s top choice (which is the only thing that matters about one’s vote).⁹ On the other hand, in some reasonable cases, misstating may make sense. If one’s goal is “Make your top choice the unique winner or failing that make your second choice the unique winner or ... or failing that make your last choice the unique winner or failing that make there be multiple winners,” then the two-candidate single-peaked election where voter 1 votes $a > b$ but voter 2 prefers $b$ to $a$ already gives voter 2 an incentive to vote, insincerely, $a > b$. (Before leaving this topic, we should stress that discussions of strategy-proofness typically assume that manipulators come in with complete preference orders, but in the Bartholdi, Tovey, and Trick [BTT89] model (which this paper and most complexity papers use when studying manipulation), the manipulative coalition is a blank slate with its only goal being to make a certain candidate $p$ be a winner.)

Another open direction is to find cases where partition-of-candidates control shifts from NP-hard to polynomial time when restricted to single-peaked electorates.

An open issue not already mentioned in this paper is the following. Section 6 provided for single-peaked electorates a manipulation-complexity dichotomy that applies to all scoring rules (and see [HH07] for what holds without the single-peaked restriction). Although that is a broad set of rules, our theorem is connecting the specification of the system to the system’s complexity—a natural connection. However, it is also natural to wonder whether one can tightly link the social-choice-favored properties of a rule to its manipulation (or bribery or control) complexity. To give an idea of the kind of theorem we are thinking of, we mention the following known theorem linking social-choice-properties to winner-problem complexity (the statement involves some notions we will not define here): Every election system that is neutral, Condorcet-consistent, and consistent has a $\Theta_2^p$-complete winner problem ([HSV05], see also the discussion in [HH00]). However, the

---

⁹We mention in passing that the two “no incentive to manipulate” claims we just made for weakCondorcet elections for single-peaked electorates also hold for a family of related election systems for single-peaked electorates. In weak-Condorcet, after sorting the voters by first choice (under the single-peaked ordering of candidates), all candidates who are or fall between the median voter(s) are winners. If we think about that as being “MedianVoting₁,” then for each rational $\alpha, 0 \leq \alpha \leq \frac{1}{2}$, for single-peaked voting we can consider the rule, MedianVoting$_\alpha$, that after ordering the voters by first-choice makes the winners be all candidates that in the societal order $L$ fall in the inclusive interval between (a) the leftmost voter $v_{left}$ such that at least $\lceil \alpha \cdot \|V\| \rceil$ voters have first preference the same as $v_{left}$ or to the left under $L$ as that of $v_{left}$, and (b) the rightmost voter $v_{right}$ such that at least $\lceil \alpha \cdot \|V\| \rceil$ voters have first preference the same as $v_{right}$ or to the right under $L$ as that of $v_{right}$. Each rule MedianVoting$_\alpha$ will share the “no incentive to manipulate” properties mentioned in the text.
just-quoted winner result is something of a cheat as there is just one system that satisfies those properties—Kemeny elections. The dream case for manipulation—and bribery and control—would be to find a broad link between social-choice properties and complexity in the single-peaked case or in the general case. In the true dream case, we might completely characterize in terms of some statement about social-choice properties the election systems with easy manipulation (or bribery or control) problems, in the single-peaked case or in the general case.

8 Conclusions

The theme of this paper is that single-peaked electorates often tame combinatorial explosion. We saw this first for the case of the winner problem. In that case, this taming is good. It shows that for single-peaked electorates, election systems such as Kemeny have efficient winner algorithms, despite their $\Theta_2^p$-hardness in the general case. But then for bribery and control (and in part, manipulation), we saw many cases where NP-hard problems fell to polynomial time for single-peaked electorates, via algorithms that bypassed the general-case combinatorial explosions of covers and partitions. Since those NP-hardness results were protections against attacks on elections, our results should serve as a warning that those protections are at their very core dependent on the extreme flexibility of voter preference collections the general case allows. For single-peaked electorates, those protections vanish.

Acknowledgments We thank Steven Brams, Piotr Faliszewski, Felix Fischer, Paul Harrenstein, Jérôme Lang, Ariel Procaccia, Jörg Rothe, Hans Georg Seedig, Troels Sørensen, and the anonymous conference referees for helpful comments and suggestions. We are grateful to Piotr Faliszewski for the email exchange that led to Theorem B.5 and to Paul Harrenstein for preparing the figures.

References


P. Faliszewski, E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. The shield that never was: Societies with single-peaked preferences are more open to manipulation and control. *Information and Computation*, 209:89–107, 2011.


A Spotlight Proofs

We here include a “spotlight” proof for each of the paper’s four results sections. These proofs are chosen as they either are particularly representative of their sections’ techniques or they simply are of a particularly important result.

A.1 Proof of Theorem 3.4

Recall the following easy characterization of Condorcet winners in the single-peaked setting. If $\|V\|$ is odd, the top choice of the median voter is a Condorcet winner. If $\|V\|$ is even, there are two cases: either both median voters have the same top choice or not. In the former case, the median voters’ preferred candidate is a Condorcet winner, and in the latter case there is no Condorcet winner (since the top choices of two medians, if different, will tie each other).

Given an election instance $(C, V)$ and a valid single-peaked order $L$, we now show how to compute all strongYoung winners in P-time. Recall that strongYoung winners are all candidates that can be made Condorcet winners by the fewest voter deletions. Although we typically assume $C \neq \emptyset$, we mention that if $C = \emptyset$ there can never be winners. If there are zero voters, all candidates are strongYoung winners, as they all tie at distance $\infty$, by convention. If $(C, V)$ has a Condorcet winner, then that is the unique strongYoung winner. Otherwise $\|V\| \geq 2$ is even and the two median voters have different top choices, say $m_\ell$ and $m_r$. Then, the strongYoung winner set is $\{m_\ell, m_r\}$, as those two candidates have a strongYoung score of 1, no one has score 0, and everyone else has score at least 2.

We now show that Algorithm 1, which clearly runs in polynomial time, computes all Dodgson winners. Recall that Dodgson winners are the candidates who can by the fewest repeated transpositions of adjacent candidates in voters’ orders (so-called switches) become Condorcet winners. If $\|C\| = 0$, then there are no winners, if $\|V\| = 0$, all of $C$ ties as winners, and if $(C, V)$ has a Condorcet winner, this candidate is the unique Dodgson winner. Otherwise $\|V\| \geq 2$ is even and the two median voters have different top choices, say $m_\ell$ and $m_r$. Then, the strongYoung winner set is $\{m_\ell, m_r\}$, as those two candidates have a strongYoung score of 1, no one has score 0, and everyone else has score at least 2.

Claim A.1. Algorithm 1 will find the correct Dodgson score of each candidate $p \in [m_\ell, m_r]_L$.

Claim A.2. Every Dodgson winner is in $[m_\ell, m_r]_L$.

In our proofs of these two claims, we use the following simple claim.
Claim A.3.  
1. Let \( p \in [m_\ell, m_r]_L \). If \((aLbLp \text{ or } pLbLa)\) and \( a \text{ ties } p \), then \( b \text{ ties } p \).

2. If \( m_rLd \), then \( m_r \text{ beats } d \). If \( dLm_\ell \), then \( m_\ell \text{ beats } d \).\(^{10}\)

Proof.

1. We know that half of the voters prefer \( a \) to \( p \) and half of the voters prefer \( p \) to \( a \). All voters that prefer \( a \) to \( p \) also prefer \( b \) to \( p \). This implies that \( p \) at best ties \( b \). By Fact 3.1, \( p \) is a weak Condorcet winner. It follows that \( p \) ties \( b \).

2. We prove the first statement. The proof of the second statement is analogous. Suppose for a contradiction that \( m_r \) does not beat \( d \). Since \( m_r \) is a weak Condorcet winner, \( m_r \) ties \( d \). Let \( \hat{d} \) be the candidate immediately to the right of \( m_r \) (with respect to \( L \)). Using part 1 of this claim, it follows that \( \hat{d} \) ties \( m_r \).

Since \( \hat{d} \) is not a weak Condorcet winner, there exists a candidate \( c \) such that \( c \) beats \( \hat{d} \). Note that \( cLm_rLd \) or \( m_rL\hat{d}Lc \). If \( cLm_rLd \), then every voter who prefers \( c \) to \( \hat{d} \) also prefers \( m_r \) to \( \hat{d} \). And so \( m_r \) beats \( \hat{d} \), which contradicts the fact that \( m_r \) ties \( \hat{d} \). If \( m_rL\hat{d}Lc \), then every voter who prefers \( c \) to \( \hat{d} \) also prefers \( \hat{d} \) to \( m_r \). And so \( c \) and \( \hat{d} \) beat \( m_r \), which contradicts the fact that \( m_r \) is a weak Condorcet winner.

\( \square \)

Proof of Claim A.1. Consider an optimal (with respect to the number of switches) way to turn \( p \) into a Condorcet winner. We first assume that \( T_\ell \neq \emptyset \) and that \( T_r \neq \emptyset \). Let \( \ell \) be the leftmost candidate in \( T_\ell \) and let \( r \) be the rightmost candidate in \( T_r \). Since \( p \) needs to gain a vote over \( \ell \), there exists a voter \( v_\ell \) such that \( \ell > v_\ell \ p \) and \( p \) gets switched beyond \( \ell \) in \( v_\ell \). Since \( p \) needs to gain a vote over \( r \), there exists a voter \( v_r \) such that \( r > v_r \ p \) and \( p \) gets switched beyond \( r \) in \( v_r \). Let \( A_\ell, B_\ell, \) and \( C_\ell \) be sets of candidates such that \( v_\ell \)'s order is of the form

\[ A_\ell > \ell > B_\ell > p > C_\ell. \]

Let \( A_r, B_r, \) and \( C_r \) be sets of candidates such that \( v_r \)'s order is of the form

\[ A_r > r > B_r > p > C_r. \]

\(^{10}\)Note that it is possible for a candidate that is not a weak Condorcet winner to tie a weak Condorcet winner. For example, in the “universal counterexample” at the end of Section B.2, candidate \( a \), which is not a weak Condorcet winner, ties weak Condorcet winner \( c \).
Note that $T_t >_{v_t} p >_{v_t} T_r$ and $T_r >_{v_r} p >_{v_r} T_t$. Clearly, $v_t \neq v_r$ and it takes $|B_t| + 1$ switches to switch $p$ beyond $\ell$ in $v_t$ and it takes $|B_r| + 1$ switches to switch $p$ beyond $r$ in $v_r$. After these $|B_t| + |B_r| + 2$ switches, $v_t$’s order has turned into $A_t > p > \ell > B_t > C_t$ and $v_r$’s order has turned into $A_r > p > r > B_r > C_r$. Since $T_t \subseteq C_r$, we still have to ensure that $p$ gains a vote over every candidate in $A_t \cap T_t$ and since $T_r \subseteq C_t$, we still have to ensure that $p$ gains a vote over every candidate in $A_r \cap T_r$. So,

$$\text{DodgsonScore}(p) \geq |B_t| + |B_r| + 2 + |A_t \cap T_t| + |A_r \cap T_r|.$$ 

We now show that Algorithm 1 correctly computes the Dodgson score of $p$. First note that the algorithm computes an upper bound on the Dodgson score, since $p$ is made a Condorcet winner in each iteration of the for loop (recall from Fact 3.1 that $p$ is already a weak Condorcet winner). Now consider the score computed by the algorithm for voters $v_t$ and $v_r$ above (since $T_t >_{v_t} p$ and $T_r >_{v_r} p$, the algorithm will consider these two voters). In the analysis, it may help to keep in mind that $\ell LT_t - \ell \ell p LT_t - \ell \ell r LT_r$.

If the top choice of $v_t$ is $\ell$ itself or to the left of $\ell$, then $A_t \cap T_t = \emptyset$. In this case, moving $p$ up in the order of $v_t$ beyond every candidate in $T_t$ gives

$$A_t > p > \ell > B_t > C_t.$$ 

If the top choice of $v_t$ is to the right of $\ell$, then for every candidate $c \in A_t$, $\ell \ell c \ell p$. It follows by Claim A.3.1 that $A_t = A_t \cap T_t$. In this case, the algorithm changes $v_t$ to

$$p > A_t > \ell > B_t > C_t.$$ 

In both cases, the algorithm uses $|B_t| + 1 + |A_t \cap T_t|$ switches on $v_t$. The same argument shows that the algorithm uses $|B_r| + 1 + |A_r \cap T_r|$ switches on $v_r$. This clearly makes $p$ a Condorcet winner using $|B_t| + |B_r| + 2 + |A_t \cap T_t| + |A_r \cap T_r|$ switches. Since this is also an upper bound (see above), it follows that

$$\text{DodgsonScore}(p) = |B_t| + |B_r| + 2 + |A_t \cap T_t| + |A_r \cap T_r|.$$ 

We still have to handle the case that $T_t = \emptyset$ or $T_r = \emptyset$. Without loss of generality, assume that $T_t = \emptyset$. Since $m_t$ and $m_r$ tie $p$ and $m_t \neq m_r$, it follows that $T_t \neq \emptyset$. Let $\ell$ be the leftmost candidate in $T_t$. As in the previous case, there exist a voter $v_t$ and sets of candidates $A_t$, $B_t$, and $C_t$ such that $v_t$’s order is of the form

$$A_t > \ell > B_t > p > C_t$$
and
\[ \text{DodgsonScore}(p) \geq \|B\| + 1 + \|A \cap T\|. \]

We now show that Algorithm 1 correctly computes the Dodgson score of \( p \). Consider the score computed by the algorithm for voter \( v_\ell \) above and letting \( v_r \) be an arbitrary voter. Since \( T_\ell > v_\ell \) \( p \) and \( T_r > v_r \) \( p \) (since \( T_r = \emptyset \)), the algorithm will consider these two voters.

As in the previous case, the algorithm uses \( \|B\| + 1 + \|A \cap T\| \) switches on \( v_\ell \). And since \( V_r = \emptyset \), the algorithm uses zero switches on \( v_r \). This clearly makes \( p \) a Condorcet winner using \( \|B\| + 1 + \|A \cap T\| \) switches. Since this is also an upper bound, it follows that \( \text{DodgsonScore}(p) = \|B\| + 1 + \|A \cap T\| \).

\[ \square \]

Claim A.1

Proof of Claim A.2. Let \( d \not\in [m_\ell, m_r]_L \). Without loss of generality, assume that \( m_r LD \). We will show that \( \text{DodgsonScore}(d) \) \( \geq \text{DodgsonScore}(m_r) \), which implies that \( d \) is not a Dodgson winner. Let \( T \) be the set of candidates in \( C - \{m_r\} \) that \( m_r \) ties with. Note that \( T \neq \emptyset \), since \( m_\ell \) ties with \( m_r \).

For every \( c \in T \), \( cLM_r \) (by Claim A.3.2) and \( d \) does not beat \( c \) (half of the voters prefer \( c \) to \( m_r \), and since \( cLM_r LD \), these voters prefer \( c \) to \( m_r \) to \( d \)).

Consider an optimal (with respect to the number of switches) way to turn \( d \) into a Condorcet winner. Let \( \ell \) be the leftmost candidate in \( T \). Since half of the voters prefer \( \ell \) to \( m_r \) to \( d \), there exists a voter \( v \) such that \( v > \ell > m_r > d \) and \( d \) gets switched beyond \( \ell \) in \( v \). Let \( A, B, C_1, \) and \( C_2 \) be sets of candidates such that \( v \)'s order is of the form
\[ A > \ell > B > m_r > C_1 > d > C_2. \]

It takes \( \|B\| + \|C_1\| + 2 \) switches to switch \( d \) beyond \( \ell \) in \( v \), and after these switches, \( v \)'s order has turned into
\[ A > d > \ell > B > m_r > C_1 > C_2. \]

We still have to ensure that \( d \) gains a vote over every candidate in \( A \cap T \). So,
\[ \text{DodgsonScore}(d) \geq \|B\| + \|C_1\| + 2 + \|A \cap T\|. \]

Now consider \( m_r \). Since \( \ell \) is the leftmost candidate in \( T \) and for every \( c \in T \), \( cLM_r \), it holds that \( T > v m_r \). Since \( m_r \) is a weak Condorcet winner, moving \( m_r \) up in the order of \( v \) beyond every candidate in \( T \) makes \( m_r \) a Condorcet winner and gives an upper bound on the Dodgson score of \( m_r \).
Algorithm 1 Dodgson winners

1: if \( \|V\| = 0 \) or \( \|C\| = 0 \) then
2: \quad return \( C \)
3: else if \( \|V\| \) is odd or (\( \|V\| \) is even and the two median voters have the same top choice) then
4: \quad return the candidate chosen by the median voter(s)
5: else
6: \quad Let \( m_\ell \) and \( m_r, m_\ell L m_r \), be the top choices of the median voters
7: \quad for all \( p \in [m_\ell, m_r]_L \) do
8: \quad \quad DodgsonScore\( (p) \) \( \leftarrow \) \( \infty \)
9: \quad \quad Let \( T_\ell \) be the set of candidates \( c \) such that \( cLp \) and \( c \) ties \( p \)
10: \quad \quad Let \( T_\ell \) be the set of candidates \( c \) such that \( pLc \) and \( c \) ties \( p \)
11: \quad \quad for all \( v_\ell, v_r \in V \) such that \( T_\ell > v_\ell \) \( \) and \( T_r > v_r \) \( p \) do
12: \quad \quad \quad Move \( p \) up in the order of \( v_\ell \) beyond every candidate in \( T_\ell \)
13: \quad \quad \quad Move \( p \) up in the order of \( v_r \) beyond every candidate in \( T_r \)
14: \quad \quad \quad Let \( n \) be the number of switches used to do this
15: \quad \quad \quad if \( n < \) DodgsonScore\( (p) \) then
16: \quad \quad \quad \quad DodgsonScore\( (p) \) \( \leftarrow \) \( n \)
17: \quad \quad return \{ \( p \in C \mid \) DodgsonScore\( (p) \) \( \leq \) DodgsonScore\( (c) \) for all \( c \in [m_\ell, m_r]_L \) \}

If the top choice of \( v \) is \( \ell \) itself or to the left of \( \ell \), then \( A \cap T = \emptyset \). In this case, moving \( m_r \) up in the order of \( v \) beyond every candidate in \( T \) gives

\[ A > m_r > \ell > B > C_1 > d > C_2. \]

If the top choice of \( v \) is to the right of \( \ell \), then for every candidate \( c \in A_\ell LcLm_r \). It follows by Claim A.3.1 that \( A = A \cap T \). In this case, moving \( m_r \) up in the order of \( v \) beyond every candidate in \( T \) gives

\[ m_r > A > \ell > B > C_1 > d > C_2. \]

In both cases, we use \( \|B\| + 1 + \|A \cap T\| \) switches on \( v \). This clearly makes \( m_r \) a Condorcet winner, and

\[ \text{DodgsonScore}(m_r) \leq \|B\| + 1 + \|A \cap T\| < \text{DodgsonScore}(d). \]

It follows that \( d \) is not a Dodgson winner. \( \square \) Claim A.2
A.2 Proof of Theorem 4.2

Let \((C, V)\) be an instance of a single-peaked election where the societal order \(L\) is given by \(c_1 L c_2 L \ldots L C\|C\|\) and let \(k\) be our bribe limit. We have to decide whether a designated candidate \(p \in C\) can be made an approval winner by bribing \(\leq k\) voters. Without loss of generality, we can assume that after the bribe, all bribed voters approve of \(p\) and disapprove of all other candidates.

Partition the multiset \(V\) of all voters into the multiset \(V_+\) of voters that approve of \(p\) and the multiset \(V_-\) of voters that disapprove of \(p\). If \(k \geq \|V_-\|\), we can obviously make \(p\) a winner by bribing all voters from \(V_-\), since after the bribe all voters approve of \(p\).\footnote{The only case in which we need to bribe a voter from \(V_+\) is if everybody approves \(p\) (i.e., \(V_- = \emptyset\)), there exists a candidate other than \(p\) that is also approved by every voter, and we want to make \(p\) the unique approval winner. In this case, bribing one arbitrary voter obviously suffices.}

Now assume that \(k < \|V_-\|\). Without loss of generality, we can assume that we bribe exactly \(k\) voters, because there is a successful bribe that involves \(\leq k\) voters if and only if there is a successful bribe that involves exactly \(k\) voters. We can also argue that bribing a voter from \(V_+\) is never more profitable than bribing a voter from \(V_-\). The reason is that for each \(c_i \in C - \{p\}\), bribing a voter from \(V_+\) lowers the quantity “number of approvals for \(c_i\) − number of approvals for \(p\)” by at most 1, whereas bribing a voter from \(V_-\) lowers this quantity by at least 1. Thus we assume without loss of generality that we bribe only voters from \(V_-\) and we know that after the bribe, \(p\) has exactly \(\hat{k} = \|V_+\| + k\) approvals.

For each candidate \(c_i \in C - \{p\}\), let \(V_{c_i}\) be the multiset of voters that approve of \(c_i\) and define the surplus \(n_i\) of \(c_i\) as \(n_i = \|V_{c_i}\| - \hat{k}\). In order to make \(p\) a winner, we have to bribe at least \(n_i\) voters from \(V_{c_i} \cap V_-\) for all candidates \(c_i\) that have a positive surplus.

Let us first consider candidates to the right of \(p\), i.e., candidates \(c\) with \(p L c\). In order to avoid incomparability problems, we start at the right end of \(L\). Let \(c_i\) be the rightmost candidate that has a positive surplus \(n_i > 0\). We know that we have to bribe at least \(n_i\) voters from \(V_{c_i} \cap V_-\), but the question is which ones. As \(n_j \leq 0\) for all \(j > i\), we can solely focus on candidates to the left of \(c_i\) and bribe the \(n_i\) voters from \(V_{c_i} \cap V_-\) whose approval range extends the furthest to the left of \(c_i\). After the bribe, those voters approve of \(p\) only. We have thereby achieved that \(n_i = 0\) and it is clear that our choice was optimal in the sense that no other choice would have removed a greater number of approvals from candidates other than \(p\) (ignoring candidates to the right of \(c_i\)).

We now update the surplus for all candidates and move to the next rightmost candidate \(c_{i'}\) to the right of \(p\) with a positive surplus. (Observe that \(i' < i\) because for all \(j > i\), \(n_j\) was initially nonpositive and the surplus of a candidate never grows when bribing a voter to approve only of \(p\).)

We repeat this procedure until all candidates to the right of \(p\) have a nonpositive surplus, at which point...
point we mirror the societal order and repeat the whole process, thereby handling all candidates $c$ with $cLp$.

If we exceed our bribe limit $k$ during this process, there cannot be a successful bribery action: All the choices we have made during the algorithm are provably at least as good as any other choice would have been. If, on the other hand, bribing $k' \leq k$ voters suffices to make all surpluses nonpositive, we bribe $(k - k')$ additional voters chosen arbitrarily from $V_-$ (to ensure that $p$ has $\hat{k}$ approvals) and have thereby found a successful bribery action. 

A.3 Proof of Theorem 5.1

Let $(C, V)$ be an election and $L$ a linear order of $C$ with respect to which the electorate is single-peaked. We have to decide whether a designated candidate $p \in C$ can be made an overall winner by partitioning the set of voters in an appropriate way.

Our algorithm is tailored to the natural Ties Promote, nonunique-winner model, but we will at the end of the proof mention how it can be adapted to the other models. In our natural model, we want to find a partition $(V_1, V_2)$ such that $p$ is a weakCondorcet winner in the election $(C', V)$ where $C'$ is the union of the weakCondorcet winners in $(C, V_1)$ and $(C, V_2)$.

We show that Algorithm 2 returns a partition with this property whenever one exists. Algorithm 2 loops over all 4-tuples $a, b, c, d$ of candidates and tests whether the voters can be partitioned into $(V_1, V_2)$ in such a way that (a) the weakCondorcet winners of $(C, V_1)$ are $[a, b]$ (i.e., all candidates in $\{x \in C \mid a \leq x \leq b\}$), (b) the weakCondorcet winners of $(C, V_2)$ are $[c, d]$, and (c) $p$ is a weakCondorcet winner in $([a, b] \cup [c, d], V)$.

For each 4-tuple $a, b, c, d$, we divide the set $C$ of candidates into disjoint “interesting regions.” Regions are defined as follows. Each of the candidates $a, b, c, d$ constitutes a region in itself. Furthermore, each contiguous (with respect to $L$) interval between two of those four candidates is a region. Finally, there are two additional regions, namely one interval consisting of all candidates that are to the left of $a, b, c, d$ and one interval consisting of all candidates that are to the right of $a, b, c, d$.

Note that all of the intervals not containing $a, b, c, d$ may be empty, because the set $\{a, b, c, d\}$ may contain adjacent or even identical candidates. It is easy to see that the number of interesting (i.e., nonempty) regions is at most nine, and is equal to nine if and only if there are no adjacent or identical candidates among $a, b, c, d$. Assuming that this is the case, there are three possible situations, depending on the relation between the intervals $[a, b]$ and $[c, d]$.

1. The intervals are disjoint:
**Algorithm 2** weakCondorcet Control by Partition of Voters

1. **for all** \(a, b, c, d \in C\) such that \(a \leq_L b, c \leq_L d, a \leq_L c\), and \(p\) is a weakCondorcet winner in \((C', V)\) with \(C' = \{e \in C \mid a \leq_L e \leq_L b \lor c \leq_L e \leq_L d\}\) do
2. **for all** \(k \in \{0, 1, 2, \ldots, \|V\|\}\) do
3. let \(r\) be the number of interesting regions
4. **for all** \(\ell = (\ell_1, \ell_2, \ldots, \ell_r) \in \mathbb{N}^r\) with \(\sum_{i=1}^r \ell_i = k\) do
5. define \(\mathcal{P}_\ell\) as the set of partitions \((V_1, V_2)\) such that for each \(i, \ell_i\) is the number of voters in \(V_i\) whose top choice is in \(R_i\)
6. **if** \(\mathcal{P}_\ell \neq \emptyset\) **then**
7. let \((V_1, V_2)\) be an arbitrary partition from \(\mathcal{P}_\ell\)
8. **if** the set of weakCondorcet winners of \((C, V_1)\) equals \([a, b]\) and the set of weakCondorcet winners of \((C, V_2)\) equals \([c, d]\) **then**
9. **return** \((V_1, V_2)\)
10. **return** “there is no partition of voters that makes \(p\) a weakCondorcet winner”

---

2. The intervals have a nonempty intersection, but neither contains the other:

3. One interval contains the other:

If the set \(\{a, b, c, d\}\) contains adjacent or identical elements, some of the regions pictured above will be empty or identical, and so there will be less than nine interesting regions. If there are \(r\) interesting regions, we will denote them by \(R_1, \ldots, R_r\), from left to right with respect to \(L\).
Associate each voter with the candidate at the top of that voter’s preference order. The following observation turns out to be helpful. If \([a,b]\) is the set of weakCondorcet winners in \((C,V_1)\), then there can be no voters in \(V_1\) that have a top choice \(x\) with \(a <_L x <_L b\). Similarly, no voter in \(V_2\) can have a top choice strictly between \(c\) and \(d\). That is, a region that consists of voters whose top choice lies strictly between either \(a\) and \(b\) or \(c\) and \(d\) is fully determined with regard to the question of how many voters are in \(V_1\) and \(V_2\). For example, in the case where \([c,d]\) is contained in \([a,b]\), five of the nine regions are fully determined: All voters from \(R_3, R_4, R_6, \) and \(R_7\) have to be in \(V_2\) and there must not be any voters in region \(R_5\), as such voters would lie both between \(a\) and \(b\) and between \(c\) and \(d\). By this argument, one can see that the maximum number of regions that are not fully determined is 7 (in the case where the intervals \([a,b]\) and \([c,d]\) are disjoint). Clearly, the number of ways that \(\|V\|\) can be divided into 7 ordered numbers is bounded by \(\|V\|^6\).

The fact that the weakCondorcet winners in \((C,V_1)\) and \((C,V_2)\) can be efficiently checked is due to the following key observation. Within each region, the only thing that affects the winner set is the number of voters we put into \(V_1\), not which voters we use to achieve that number. That is, we do not need to check each partition individually (there are an exponential number of them), but rather deal with a large number of partitions simultaneously. More formally, suppose we have \(r\) interesting regions and let \(\ell = (\ell_1, \ell_2, \ldots, \ell_r)\) be an \(r\)-dimensional vector of natural numbers. We now define \(\mathcal{P}_\ell\) as the set of all partitions \((V_1,V_2)\) of \(V\) such that for each \(i\), \(\ell_i\) is the number of the voters from \(V_1\) whose top choice is in \(R_i\).

For \(\mathcal{P}_\ell \neq \emptyset\), our key observation can now be restated as follows. \([a,b]\) is the set of weakCondorcet winners in some election \((C,V_1)\) with \((V_1,V_2) \in \mathcal{P}_\ell\) if and only if \([a,b]\) is the set of weakCondorcet winners in every election \((C,V_1)\) with \((V_1,V_2) \in \mathcal{P}_\ell\). That is, if we want to check whether \([a,b]\) is the set of weakCondorcet winners in any of the primary elections \((C,V_1)\) induced by any of the partitions \((V_1,V_2) \in \mathcal{P}_\ell\), it suffices to check just one of them to obtain the answer. This check can easily be done just by counting. By symmetry, the same statement holds for \([c,d]\) and for weakCondorcet winners of the elections \((C,V_2)\) (all four combinations).

The reason why this is true is that, given the number of voters in each region, it is easy to recognize the region(s) of the median voter(s) (just by counting). Since \(a, b, c,\) and \(d\) all constitute a region on their own, it is equally easy to tell whether a median voter has any of these four candidates as his or her top choice.

Implementing this idea, the algorithm loops over all possible sizes \(k\) of \(V_1\) (line 2) and over all possibilities how \(k\) can be divided into \(r\) numbers \(\ell_1, \ell_2, \ldots, \ell_r\), and checks (line 8) if this gives a partition as required. As we have argued in the last paragraphs, this query can be efficiently answered.
The running time of Algorithm 2 can thus be bounded as follows. The number of iterations for the loops in lines 1, 2, and 4, are bounded by \( \|C\|^4, \|V\| + 1, \) and \( \|V\|^6, \) respectively. Moreover, we have just shown that the queries in line 8 can be answered in polynomial time. Altogether, this yields a running time that is obviously polynomial in the size of the input.

Correctness of Algorithm 2 should be clear from the explanations above: We find a partition that makes \( p \) an overall weak Condorcet winner if and only if such a partition exists. In particular, observe that setting \( k = 0 \) in line 2 handles the case where \( p \) is already a weak Condorcet winner in the original election.\(^{12}\)

This completes the proof for the TP, nonunique-winner model. For the TP, unique-winner model, line 1 of Algorithm 2 needs to be adapted to only loop over all choices of \( a, b, c, d \) that make \( p \) a unique weak Condorcet winner in \( (C', V) \). The TE, unique-winner model is dealt with in Theorem 5.5 on page 22. (By Fact 3.1 on page 10, being a unique weak Condorcet winner is tantamount to being a Condorcet winner for single-peaked electorates.)

The same is true for the TE, nonunique-winner model. Here, being a weak Condorcet winner in the final election suffices, but Algorithm 3 on page 64 can easily be adapted to take that into consideration. \( \square \)

### A.4 Proof of Theorem 6.2

For \( m \leq 1 \), the problem is trivial and thus in \( P \). So, assume that \( m \geq 2 \). We split the proof of Theorem 6.2 into three lemmas: The \( \alpha_2 > \alpha_{\left\lfloor \frac{m-1}{2} \right\rfloor + 2} \) NP-complete cases (Lemma A.5), the \( \alpha_2 > \alpha_{\left\lfloor \frac{m-1}{2} \right\rfloor + 2} \) \( P \) cases (follow from Lemma A.7), and the \( \alpha_2 = \alpha_{\left\lfloor \frac{m-1}{2} \right\rfloor + 2} \) cases (Lemma A.8).

Our NP-hardness proofs follow via reductions from the well-known NP-complete problem Partition (see, e.g., [GJ79]). In this problem, we are given a nonempty collection \( (k_1, \ldots, k_n) \) of positive integers that sum to \( 2K \), and we ask whether there exists a subcollection of \( k_1, \ldots, k_n \) that sums to \( K \).

In the proof, we use the following notation. For \( V \) a collection of voters and \( c \) a candidate, \( \text{score}_V(c) \) denotes the score of \( c \) in \( V \), i.e., the number of points that \( c \) receives from the voters in \( V \). If \( V \) is clear from context, we will simply write \( \text{score}(c) \). In this section, we usually denote the collection of nonmanipulators by \( S \) and the collection of manipulators by \( T \).

---

\(^{12}\)To see this, assume that \( p \in [m_l, m_r] \), where \( [m_l, m_r] \) is the interval of all weak Condorcet winners in \( (C, V) \). Observe that if we set \( a, b, c, d \) such that \( [a, b] = C \) and \( [c, d] = [m_l, m_r] \), then \( C' = \{ e \in C \mid a \leq L e \leq L b \land c \leq L e \leq L d \} = C \) and obviously \( p \) is a weak Condorcet winner of \( (C', V) \). Thus, this choice of \( a, b, c, d \) is considered in line 1. Now setting \( k \) to 0 uniquely defines \( \ell \) to be \((0, \ldots, 0)\) and \( \mathcal{P}_\ell \) consists of the partition \((\emptyset, V)\) only. Due to the choice of \( a, b, c, d \), the answer to the query in line 8 is “yes” and \((\emptyset, V)\) is correctly output as a partition that makes \( p \) an overall winner.
First we prove the following simple lemma.

**Lemma A.4.** In the constructive coalition weighted manipulation problem for the single-peaked case for scoring protocols, if \( p \) can be made a winner, then \( p \) can be made a winner by a manipulation in which all manipulators rank \( p \) first.

**Proof.** Let \( A > p > b_1 > \cdots > b_\ell \) be a single-peaked vote, where \( A \) is a set of candidates in some order. Then there exists an ordering of \( A \) such that \( p > A > b_1 > \cdots > b_\ell \) is also single-peaked. Note that in this new vote, for every candidate \( c \), score\( (p) - \text{score}(c) \) does not decrease. So, if \( p \) is a winner of an election, then \( p \) is a still a winner if we replace every single-peaked vote of the form \( A > p > b_1 > \cdots > b_\ell \) by a single-peaked vote of the form \( p > A > b_1 > \cdots > b_\ell \). \( \blacksquare \)

**Lemma A.5.** Let \( \alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m) \) be a scoring protocol such that \( m \geq 2 \) and \( \alpha_2 > \alpha_{\left\lfloor \frac{m-1}{2} \right\rfloor + 2} \). If there exist integers \( i_1, i_2 > 1 \) such that \( i_1 + i_2 \leq m + 1 \) and \( (\alpha_1 - \alpha_{i_1})(\alpha_1 - \alpha_{i_2}) > (\alpha_{i_1} - \alpha_{i_1+1})(\alpha_{i_2} - \alpha_{i_2+1}) \), then the constructive coalition weighted manipulation problem for the single-peaked case is NP-complete.

**Proof.** Let \( i_1, i_2 \) be integers that fulfill the conditions of the lemma such that \( i_1 \leq i_2 \) and \( i_2 \) is minimal, i.e., \( 1 < i_1 \leq i_2, i_1 + i_2 \leq m + 1, (\alpha_1 - \alpha_{i_1})(\alpha_1 - \alpha_{i_2}) > (\alpha_{i_1} - \alpha_{i_1+1})(\alpha_{i_2} - \alpha_{i_2+1}) \), and for all \( i'_1, i'_2 \) such that \( 1 < i'_1 \leq i'_2 < i_2 \) and \( i'_1 + i'_2 \leq m + 1 \), it holds that \( (\alpha_1 - \alpha_{i'_1})(\alpha_1 - \alpha_{i'_2}) \leq (\alpha_{i'_1} - \alpha_{i'_1+1})(\alpha_{i'_2} - \alpha_{i'_2+1}) \).

We will now reduce Partition to the constructive coalition weighted manipulation problem for single-peaked electorates. Let \( (k_1, \ldots, k_n) \) be an instance of Partition, i.e., \( (k_1, \ldots, k_n) \) is a nonempty collection of positive integers that sum to \( 2K \). Let society’s order be

\[
a_m L \cdots L a_1 L p L b_1 L \cdots L b_m,
\]

where \( m_2 = \max\left(\left\lfloor \frac{m-1}{2} \right\rfloor, i_2 - 1 \right) \) and \( m_1 = m - m_2 - 1 \). Note that \( i_2 \leq m_2 + 1 \). Since \( i_1 + i_2 \leq m + 1 \), it follows immediately that \( i_1 \leq m_1 + 1 \). Also note that \( 1 \leq m_1 \leq m_2 \leq m - 2 \). To make the reduction work, we also need the following claim.

**Claim A.6.** \( \alpha_{m_1+2} < \alpha_2 \).

**Proof.** If \( m_1 = \left\lfloor \frac{m-1}{2} \right\rfloor \), this is immediate, since \( \alpha_2 > \alpha_{\left\lfloor \frac{m-1}{2} \right\rfloor + 2} \). If not, then \( m_2 = i_2 - 1 \) and \( i_2 - 1 > \left\lfloor \frac{m-1}{2} \right\rfloor \geq 2 \). Since \( i_2 \) is minimal, the conditions of the lemma can not be fulfilled with \( i_1 = i_2 = 2 \), and so \( (\alpha_1 - \alpha_2) \leq (\alpha_2 - \alpha_3) \). If \( \alpha_2 = \alpha_{m_1+2} \), then \( \alpha_2 = \alpha_3 \), and it follows that \( \alpha_1 = \alpha_2 \) and thus \( \alpha_1 = \alpha_{m_1+2} \). But then this choice of \( i_1, i_2 \) does not fulfill the conditions of the lemma, since \( \alpha_1 - \alpha_{i_1} = 0 \). \( \blacksquare \)
Let $S$ consist of two voters, one voter of weight $\ell_1$ with preference order

$$a_1 > \cdots > a_{i-1} > p > b_1 > \cdots$$

(the $\cdots$ at the end of the vote denotes that the remaining candidates are listed in arbitrary, single-peaked order) and one voter with weight $\ell_2$ with preference order

$$b_1 > \cdots > b_{i'-1} > p > a_1 > \cdots,$$

and let the weights of $T$ be $\beta k_1, \beta k_2, \ldots, \beta k_n$, where

$$\beta = (\alpha_1 - \alpha_{i_1})(\alpha_1 - \alpha_{i_2}) - (\alpha_{i_1} - \alpha_{i_1+1})(\alpha_{i_2} - \alpha_{i_2+1}),$$

$$\ell_1 = ((2\alpha_1 - \alpha_2 - \alpha_{m_2+2})(\alpha_1 - \alpha_{i_2}) + (\alpha_{i_2} - \alpha_{i_2+1})(2\alpha_1 - \alpha_2 - \alpha_{m_1+2}))K,$$

$$\ell_2 = ((\alpha_1 - \alpha_{i_1})(2\alpha_1 - \alpha_2 - \alpha_{m_1+2}) + (\alpha_{i_1} - \alpha_{i_1+1})(2\alpha_1 - \alpha_2 - \alpha_{m_2+2}))K.$$

These values were chosen (using Cramer’s Rule) so that

$$(\alpha_1 - \alpha_{i_1})\ell_1 - (\alpha_{i_2} - \alpha_{i_2+1})\ell_2 = (2\alpha_1 - \alpha_2 - \alpha_{m_2+2})\beta K \text{ and}$$

$$-(\alpha_{i_1} - \alpha_{i_1+1})\ell_1 + (\alpha_1 - \alpha_{i_1})\ell_2 = (2\alpha_1 - \alpha_2 - \alpha_{m_1+2})\beta K.$$ Note that $\beta$ is positive, since $(\alpha_1 - \alpha_{i_1})(\alpha_1 - \alpha_{i_2}) > (\alpha_{i_1} - \alpha_{i_1+1})(\alpha_{i_2} - \alpha_{i_2+1})$. Note that $\ell_1$ and $\ell_2$ are also positive, since $(\alpha_1 - \alpha_{i_1})$ and $(\alpha_{i_1} - \alpha_{i_2})$ are positive, and so are $(2\alpha_1 - \alpha_2 - \alpha_{m_1+2})$ and $(2\alpha_1 - \alpha_2 - \alpha_{m_2+2})$, by Claim A.6 and the fact that $m_1 \leq m_2$.

To prove that the reduction works, first suppose that $k_1, \ldots, k_n$ has a partition, i.e., there is a subcollection of $k_1, \ldots, k_n$ that sums to $K$. We show that $p$ can be made a winner as follows. Let $\beta K$ weight of $T$ vote

$$p > a_1 > \cdots > a_{m_1} > b_1 > \cdots$$

and let $\beta K$ weight vote

$$p > b_1 > \cdots > b_{m_2} > a_1 > \cdots.$$ Note that for all $i, 1 \leq i \leq m_1$, score$(a_i) \leq \text{score}(a_1)$ and that for all $i, 1 \leq i \leq m_2$, score$(b_i) \leq \text{score}(b_1)$ and so it suffices to show that score$(a_1) \leq \text{score}(p)$ and that score$(b_1) \leq \text{score}(p)$. Note that

$$\text{score}(p) = \alpha_{i_1} \ell_1 + \alpha_{i_2} \ell_2 + 2\alpha_1 \beta K,$$

$$\text{score}(a_1) = \alpha_1 \ell_1 + \alpha_{i_1+1} \ell_2 + \alpha_2 \beta K + \alpha_{m_2+2} \beta K, \text{ and}$$

$$\text{score}(b_1) = \alpha_1 \ell_2 + \alpha_{i_1+1} \ell_1 + \alpha_2 \beta K + \alpha_{m_1+2} \beta K.$$
From the choice of $\beta$, $\ell_1$, and $\ell_2$, it follows that $\text{score}(a_1) = \text{score}(p)$ and $\text{score}(b_1) = \text{score}(p)$.

For the converse, suppose that the voters in $T$ vote such that $p$ becomes a winner in $S \cup T$. From the observations above, it follows that

$$\text{score}_T(a_1) \leq (\alpha_2 + \alpha_{m+2})\beta K$$

and that

$$\text{score}_T(b_1) \leq (\alpha_2 + \alpha_{m+2})\beta K.$$

By Lemma A.4 we can assume that $p$ is ranked first by every voter in $T$. This implies that every voter in $T$ ranks $a_1$ or $b_1$ second. Let $W_a$ be the total weight of all $T$ voters that rank $a_1$ second. It follows that

$$\text{score}_T(a_1) \geq \alpha_2 W_a + \alpha_{m+2}(2\beta K - W_a)$$

and

$$\text{score}_T(b_1) \geq \alpha_2(2\beta K - W_a) + \alpha_{m+2}W_a.$$

From these observations, and the fact that $\beta > 0$ and that $\alpha_2 > \alpha_{m+2}$ (by Claim A.6) (and thus also $\alpha_2 > \alpha_{m+2}$), it follows that $W_a \leq \beta K$ and that $(2\beta K - W_a) \leq \beta K$. So, exactly half of the vote weight of $T$ ranks $a_1$ second. Then the weights of the voters in $T$ that rank $a_1$ second correspond to a partition.

\[\square\]

\textbf{Lemma A.7.} Let $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m)$ be a scoring protocol. If for all $i, j > 1$ such that $i + j \leq m + 1$, it holds that

$$(\alpha_1 - \alpha_i)(\alpha_1 - \alpha_j) \leq (\alpha_i - \alpha_{i+1})(\alpha_j - \alpha_{j+1}),$$

then the constructive coalition weighted manipulation problem for the single-peaked case is in P.

\textbf{Proof.} Let $a_{m_1}L \cdots L a_1 L p L b_1 L \cdots L b_{m_2}$ be society’s order. It is immediate that if $\text{score}_S(a_i) \leq \text{score}_S(p)$ for all $i$, $1 \leq i \leq m_1$, then $p$ can be made a winner if and only if $p$ is a winner if every voter in $T$ votes

$$p > a_1 > \cdots > a_{m_1} > b_1 > \cdots > b_{m_2}.$$  

Similarly, if $\text{score}_S(b_i) \leq \text{score}_S(p)$ for all $i$, $1 \leq i \leq m_2$, then $p$ can be made a winner if and only if $p$ is a winner if every voter in $T$ votes

$$p > b_1 > \cdots > b_{m_2} > a_1 > \cdots > a_{m_1}.$$  

We will show that these are the only cases that can occur. This immediately implies Lemma A.7.

So, for the remainder of the proof, suppose for a contradiction that we have a collection of voters $S$ and integers $i_1$ and $i_2$ such that $1 \leq i_1 \leq m_1$, $1 \leq i_2 \leq m_2$, $\text{score}_S(a_{i_1}) > \text{score}_S(p)$, and $\text{score}_S(b_{i_2}) > \text{score}_S(p)$. 

46
For $1 < i \leq m_1 + 1$, let $\ell_i$ be the total weight of the voters in $S$ that rank some candidate in $\{a_1, \ldots, a_{m_1}\}$ first and that rank $p$ in $i$th place. Note that these voters rank all candidates in $\{b_1, \ldots, b_{m_2}\}$ after $p$. For $1 < i \leq m_2 + 1$, let $\ell'_i$ be the total weight of the voters in $S$ that rank some candidate in $\{b_1, \ldots, b_{m_2}\}$ first and that rank $p$ in $i$th place. Note that these voters rank all candidates in $\{a_1, \ldots, a_{m_1}\}$ after $p$. It follows immediately that

$$\text{scores}(a_i) - \text{scores}(p) \leq \sum_{1<i\leq m_1+1} (\alpha_1 - \alpha_i)\ell_i + \sum_{1<i\leq m_2+1} (\alpha_{i+1} - \alpha_i)\ell'_i$$

and

$$\text{scores}(b_i) - \text{scores}(p) \leq \sum_{1<i\leq m_2+1} (\alpha_1 - \alpha_i)\ell'_i + \sum_{1<i\leq m_1+1} (\alpha_{i+1} - \alpha_i)\ell_i.$$

Since $\text{scores}(a_i) > \text{scores}(p)$ and $\text{scores}(b_i) > \text{scores}(p)$, it follows that

$$\sum_{1<i\leq m_1+1} (\alpha_1 - \alpha_i)\ell_i > \sum_{1<i\leq m_2+1} (\alpha_1 - \alpha_i)\ell'_i$$

and

$$\sum_{1<i\leq m_2+1} (\alpha_1 - \alpha_i)\ell'_i > \sum_{1<i\leq m_1+1} (\alpha_1 - \alpha_i)\ell_i.$$

Since both sides of both inequalities are nonnegative, it follows that

$$\left(\sum_{1<i\leq m_1+1} (\alpha_1 - \alpha_i)\ell_i\right) \cdot \left(\sum_{1<i\leq m_2+1} (\alpha_1 - \alpha_i)\ell'_i\right) >$$

$$\left(\sum_{1<i\leq m_2+1} (\alpha_1 - \alpha_{i+1})\ell'_i\right) \cdot \left(\sum_{1<i\leq m_1+1} (\alpha_1 - \alpha_{i+1})\ell_i\right).$$

Multiplying this out, it follows that

$$\sum_{1<i\leq m_1+1} (\alpha_1 - \alpha_i)(\alpha_1 - \alpha_j)\ell_i\ell'_j > \sum_{1<i\leq m_1+1} (\alpha_1 - \alpha_{i+1})(\alpha_j - \alpha_{j+1})\ell_i\ell'_j.$$

Since $m = m_1 + m_2 + 1$, this contradicts the assumption that for all $i, j > 1$ such that $i + j \leq m + 1$ it holds that

$$(\alpha_1 - \alpha_i)(\alpha_1 - \alpha_j) \leq (\alpha_1 - \alpha_{i+1})(\alpha_j - \alpha_{j+1}).$$

Lemma A.7 handles all $\alpha_2 > 2 \left\lceil \frac{m-1}{2} \right\rceil + 2$ P cases of Theorem 6.2. Note that this lemma is of limited use if $\alpha_2 = \alpha_2 \left\lceil \frac{m-1}{2} \right\rceil + 2$, since in that case the lemma applies only if $m = 2$ or $\alpha_1 = \alpha_2 = \alpha_2 \left\lceil \frac{m-1}{2} \right\rceil + 2$. The $\alpha_2 = \alpha_2 \left\lceil \frac{m-1}{2} \right\rceil + 2$ cases of Theorem 6.2 are handled by the following lemma in combination with the standard observation that the scoring protocol $(\alpha_1, \alpha_2, \ldots, \alpha_m)$ has the same behavior as the scoring protocol $(\alpha_1 - \alpha_m, \alpha_2 - \alpha_m, \ldots, \alpha_{m-1} - \alpha_m, 0)$. 

47
Lemma A.8. Let $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m)$ be a scoring protocol such that $m \geq 2$, $\alpha_m = 0$, and $\alpha_2 = \alpha_1 \lceil \frac{m+1}{2} \rceil + 2$. If $\alpha_2 = 0$ or $\alpha_1 = \alpha_2$ or $(\alpha_2 = \alpha_{m-1}$ and $\alpha_1 \leq 2\alpha_2$) then the constructive coalition weighted manipulation problem for the single-peaked case is in $P$; otherwise, it is NP-complete.

Proof. If $\alpha_2 = 0$, then any vote of the form $p > \cdots$ is an optimal manipulator vote. If $\alpha_1 = \alpha_2$, then $\alpha_1 = \alpha_1 \lceil \frac{m+1}{2} \rceil + 2$. Let $i$ and $j$ be such that $i, j > 1$ and $i + j \leq m + 1$. Then $\min(i, j) \leq \lceil \frac{m+1}{2} \rceil = \lceil \frac{m-1}{2} \rceil + 1$ and so $(\alpha_i - \alpha_j)(\alpha_1 - \alpha_j) = 0$. It follows from Lemma A.7 that this case is in $P$.

For the remainder of the proof, we assume that $\alpha_1 > \alpha_2 > 0$.

Now consider the case that $\alpha_2 = \alpha_{m-1}$ and $\alpha_1 \leq 2\alpha_2$. Consider society’s order. If $p$ is the leftmost or rightmost candidate, then there is exactly one vote that puts $p$ first, and this is an optimal manipulator vote. Otherwise, let $a$ be the leftmost candidate in society’s order, and $b$ the rightmost. Then $a$ and $b$ are the only candidates that can occur last in a vote. Since $\alpha_1 \leq 2\alpha_2$, it follows that $\text{score}_S(a) + \text{score}_S(b) \leq 2\text{score}_S(p)$. Without loss of generality, let $\text{score}_S(a) \leq \text{score}_S(b)$. Then $\text{score}_S(a) \leq \text{score}_S(p)$ and any vote of the form $p > \cdots > b$ is an optimal manipulator vote.

This concludes the $P$ cases. We now turn to the NP-complete cases. In both cases, we will reduce from Partition.

First assume that $\alpha_2 = \alpha_{m-1}$ and that $\alpha_1 > 2\alpha_2$. Let $k_1, \ldots, k_n$ be a nonempty collection of positive integers that sum to $2K$. We construct the following election: Society’s order is $aLpL \cdots Lb$. $S$ consists of two voters, each of weight $(2\alpha_1 - \alpha_2)K$. One voter votes $a > p > \cdots > b$ and the other voter votes $b > \cdots > p > a$. Note that for all candidates $c \not\in \{a, b\}$, $\text{score}_S(c) = \text{score}_S(p)$. The weights of the manipulators are $(\alpha_1 - 2\alpha_2)k_1, \ldots, (\alpha_1 - 2\alpha_2)k_n$. The proof of the correctness of the reduction is similar to (but easier than) the corresponding proof in Lemma A.5. First suppose that there exists a subcollection of $k_1, \ldots, k_n$ that sums to $K$. Then set $\beta K$ vote weight of $T$ to $p > a > \cdots > b$ and $\beta K$ vote weight to $p > \cdots > b > a$. Note that in the resulting election,$$
\text{score}(p) = (2\alpha_2(2\alpha_1 - \alpha_2) + 2\alpha_1(\alpha_1 - 2\alpha_2))K = (2\alpha_1^2 - 2\alpha_2^2)K$$and$$\text{score}(a) = \text{score}(b) = (\alpha_1(2\alpha_1 - \alpha_2) + \alpha_2(\alpha_1 - 2\alpha_2))K = (2\alpha_1^2 - 2\alpha_2^2)K.$$

For the converse, suppose that $p$ can be made a winner. From the observations above, it is immediate that

$\text{score}_T(a) \leq \alpha_2(\alpha_1 - 2\alpha_2)K$ and $\text{score}_T(b) \leq \alpha_2(\alpha_1 - 2\alpha_2)K$.

Let $W_a$ be the total weight of the voters in $T$ that rank $b$ last. It follows that

$\text{score}_T(a) \geq \alpha_2W_a$ and $\text{score}_T(b) \geq (2(\alpha_1 - 2\alpha_2)K - W_a)\alpha_2$. 

48
From these observations, and the assumption that $\alpha_2 > 0$, it follows that $W_a = (\alpha_1 - 2\alpha_2)K$. So, exactly half of the vote weight of $T$ ranks $b$ last. Then the weights of the voters that rank $b$ last correspond to a partition.

Finally, we handle the last case, namely, $\alpha_1 > \alpha_2 > \alpha_{m-1}$. Let $\hat{m}$ be the smallest index such that $\alpha_2 > \alpha_{\hat{m}}$. Note that $\lfloor \frac{m-1}{2} \rfloor + 2 < \hat{m} < m$. Take society’s order to be

$$a_{\lfloor \frac{m-1}{2} \rfloor}L\cdots La_1LpLb_1L\cdots Lb_{\lfloor \frac{m-1}{2} \rfloor}. $$

We are again reducing from Partition. Let $k_1, \ldots, k_n$ be a nonempty collection of positive integers that sum to $2K$. Let $S$ consist of two voters, each of weight $(2\alpha_1 - \alpha_2 - \alpha_{\hat{m}})K$, voting

$$a_{\lfloor \frac{m-1}{2} \rfloor} > \cdots > a_1 > p > b_1 > \cdots > b_{\lfloor \frac{m-1}{2} \rfloor} > \cdots$$

and let the weights of $T$ be $(\alpha_1 - \alpha_2)k_1, \ldots, (\alpha_1 - \alpha_2)k_n$. Since $\lfloor \frac{m-1}{2} \rfloor - (m-\hat{m}) = \hat{m} - \lfloor \frac{m-1}{2} \rfloor + 1 \geq \hat{m} - (\lfloor \frac{m-1}{2} \rfloor + 2) > 0$ and $\lfloor \frac{m-1}{2} \rfloor - (m-\hat{m}) + 1 + \lfloor \frac{m-1}{2} \rfloor - (m-\hat{m}) = 2\hat{m} - m < \hat{m}$, we have the following scores.

$$\text{score}_{S}(p) = 2\alpha_2(2\alpha_1 - \alpha_2 - \alpha_{\hat{m}})K,$$

$$\text{score}_{S}(a_{\lfloor \frac{m-1}{2} \rfloor}) = (\alpha_1 + \alpha_2)(2\alpha_1 - \alpha_2 - \alpha_{\hat{m}})K,$$

$$\text{score}_{S}(b_{\lfloor \frac{m-1}{2} \rfloor}) = \text{score}_{S}(a_{\lfloor \frac{m-1}{2} \rfloor}).$$

If $k_1, \ldots, k_n$ has a partition, then set $(\alpha_1 - \alpha_2)K$ vote weight of $T$ to

$$p > a_1 > \cdots > a_{\lfloor \frac{m-1}{2} \rfloor} > b_1 > \cdots > b_{\lfloor \frac{m-1}{2} \rfloor} > \cdots$$

and set $(\alpha_1 - \alpha_2)K$ vote weight of $T$ to

$$p > b_1 > \cdots > b_{\lfloor \frac{m-1}{2} \rfloor} > a_1 > \cdots > a_{\lfloor \frac{m-1}{2} \rfloor} > \cdots.$$

Note that $a_{\lfloor \frac{m-1}{2} \rfloor}$ and $b_{\lfloor \frac{m-1}{2} \rfloor}$ are the only candidates that can score higher than $p$ and that

$$\text{score}(p) = (2\alpha_2(2\alpha_1 - \alpha_2 - \alpha_{\hat{m}}) + 2\alpha_1(\alpha_1 - \alpha_2))K = (2\alpha_1^2 + 2\alpha_1\alpha_2 - 2\alpha_2\alpha_{\hat{m}} - 2\alpha_2^2)K. $$

And $\text{score}(a_{\lfloor \frac{m-1}{2} \rfloor}) = \text{score}(b_{\lfloor \frac{m-1}{2} \rfloor}) =$

$$((\alpha_1 + \alpha_2)(2\alpha_1 - \alpha_2 - \alpha_{\hat{m}}) + (\alpha_2 + \alpha_{\hat{m}})(\alpha_1 - \alpha_2))K = (2\alpha_1^2 + 2\alpha_1\alpha_2 - 2\alpha_2\alpha_{\hat{m}} - 2\alpha_2^2)K. $$

So, $p$ is a winner of the resulting election.
For the converse, suppose that \( p \) can be made a winner. Assume (using Lemma A.4) that \( p \) is ranked first by every manipulator. From the observations above, it is immediate that
\[
\text{score}_T \left( a \left\lfloor \frac{m-1}{2} \right\rfloor - (m - ˆm) \right) \leq (\alpha_2 + \alpha_{m})(\alpha_1 - \alpha_2) \text{ and }
\text{score}_T \left( b \left\lceil \frac{m-1}{2} \right\rceil - (m - ˆm) \right) \leq (\alpha_2 + \alpha_{m})(\alpha_1 - \alpha_2).
\]
Let \( W_a \) be the total weight of all \( T \) voters such that \( a \left\lfloor \frac{m-1}{2} \right\rfloor - (m - ˆm) > b \left\lceil \frac{m-1}{2} \right\rceil - (m - ˆm) \). It follows that
\[
\text{score}_T \left( a \left\lfloor \frac{m-1}{2} \right\rfloor - (m - ˆm) \right) \geq \alpha_2 W_a + \alpha_{m}(2(\alpha_1 - \alpha_2)K - W_a) \text{ and }
\text{score}_T \left( b \left\lceil \frac{m-1}{2} \right\rceil - (m - ˆm) \right) \geq \alpha_{m} W_a + \alpha_2(2(\alpha_1 - \alpha_2)K - W_a).
\]
From these observations, and the fact that \( \alpha_2 > \alpha_m \), it follows that \( W_a = (\alpha_1 - \alpha_2)K \). Then the weights of the voters such that \( a \left\lfloor \frac{m-1}{2} \right\rfloor - (m - ˆm) > b \left\lceil \frac{m-1}{2} \right\rceil - (m - ˆm) \) correspond to a partition.

\[\square\]

B Additional Definitions and Nonspotlight Proofs

B.1 Additional Definitions

Let us define some additional election systems mentioned in Section 2.

Nanson elections are runoff methods based on Borda’s scoring protocol. In Nanson’s original definition [Nan82], a series of Borda elections is held and all candidates who at any stage have not more than the average Borda score are excluded unless all candidates have identical Borda scores, in which case these candidates are declared the winners of the election. There exist two variants of Nanson due to Fishburn and Schwartz, which exclude candidates with the lowest Borda score and candidates whose Borda score is less than the average score, respectively. All three versions fail to be weakCondorcet-consistent [Nio87].

Maximin (a.k.a. Simpson-Kramer) elections [Sim69, Kra77] choose those candidates that fare best in their worst pairwise comparison against any other candidate. The remaining three election systems are based on the pairwise majority relation. In Schwartz elections (sometimes also called the top cycle) [Sch72], the winners are defined as the maximal elements of the asymmetric part of the transitive closure of the majority relation. The winners in Fishburn elections [Fis77] are the maximal elements of the Fishburn relation \( F \), which is defined by letting \( a F b \) if every candidate that beats \( a \) in a pairwise comparison also beats \( b \) and there exists a candidate that beats \( b \) but not \( a \). Finally, Kemeny [Kem59] elections are based on the smallest number of reversals in the voters’ pairwise preferences such that the majority relation becomes transitive and complete. The Kemeny winners are the maximal elements of such minimally modified majority relations.
B.2 Proofs for Section 2

Theorem B.1. Kemeny, Schwartz, and Fishburn’s and Schwartz’s versions of Nanson are weakCondorcet-consistent for single-peaked electorates.

Proof. All statements rely on the observation that the pairwise majority relation for single-peaked electorates is transitive [Bla48, Bla58]. Further observe that weak Condorcet winners are precisely the maximal elements of the pairwise majority relation. It follows immediately that Schwartz is weakCondorcet-consistent. In the case of Kemeny, we need to modify individual preferences in order to obtain a transitive and complete majority relation. For completeness, we need to modify at least one individual preference for every pair of candidates that tie. This is also sufficient because the original majority relation is already transitive. We can thus turn exactly every weak Condorcet winner into the maximal element of a minimally modified majority relation that is transitive and complete.

For Nanson, we prove that no weak Condorcet winner is eliminated at any stage of the election. First let us normalize Borda scores by subtracting $(\|V\| \cdot (m-1))/2$ from the Borda score of every candidate. This ensures that the average normalized Borda score is precisely zero. Now, observe that the normalized Borda score of candidate $a$ is simply half of the sum, over all candidates $b$, of the number of voters who prefer $a$ to $b$ minus the number of voters who prefer $b$ to $a$. As a consequence, the normalized Borda score of every weak Condorcet winner is at least zero. Moreover, in the case of a single-peaked electorate, due to the transitivity of the majority relation, there is always a candidate with a negative normalized Borda score unless all candidates have an identical score of zero. In the latter case, all three versions of Nanson will terminate. In the former case, neither Fishburn’s nor Schwartz’s variant will exclude weak Condorcet winners because their score is at least as large as the average score.

The fact that Black, Dodgson, the original version of Nanson, and Copeland$^\alpha$ elections for all $\alpha \in [0,1)$ are not weakCondorcet-consistent for single-peaked electorates is seen by the following universal counterexample. Let there be two voters with preferences $b > a > c$ and $c > b > a$. These preferences are single-peaked with respect to the societal ordering $a L b L c$. Candidates $b$ and $c$ are weak Condorcet winners, but each of the mentioned election systems chooses only $b$. Similarly, strongYoung is not weakCondorcet-consistent for single-peaked electorates because in an election with two voters whose preferences are $a > b > c$ and $c > b > a$, all candidates are weak Condorcet winners, but strongYoung yields only candidates $a$ and $c$.  

51
B.3 Nonspotlight Proofs for Section 3

Theorem B.2. The winner problem for Young elections is \( \Theta_2^p \)-complete, both in the nonunique-winner model and the unique-winner model.

Proof. \( \Theta_2^p \)-completeness of the (nonunique) winner problem for strong Young elections (somewhat confusingly called Young elections there) was shown in [RSV03]. The same proof also establishes \( \Theta_2^p \)-completeness in the unique winner model [HHR09]. The \( \Theta_2^p \) upper bound is easy to show, and the same argument can be used to show a \( \Theta_2^p \) upper bound for the Young winner problem (in both the nonunique-winner model and the unique-winner model). \( \Theta_2^p \)-hardness of the winner problem for strong Young elections was shown as Theorem 2.5 in [RSV03] via a reduction from the \( \Theta_2^p \)-complete problem Maximum Set Packing Compare (MSPC, for short): Given two sets, \( B_1 \) and \( B_2 \), and two collections of sets \( A_1 \) and \( A_2 \), where each \( S \in A_i \) is a nonempty, finite subset of \( B_i \), is it the case that \( \kappa(A_1) > \kappa(A_2) \)? Here, \( \kappa(A_i) \) denotes the maximum number of pairwise disjoint sets in \( A_i \). As in [RSV03], we assume that \( \kappa(A_i) > 2 \).

The proofs of Theorems 2.3 and 2.5 in [RSV03] show how to construct from a given MSPC instance \( I = (B_1, B_2, A_1, A_2) \) an election \( (D, W) \) with two designated candidates, \( c \) and \( d \), such that (a) if \( \kappa(A_1) > \kappa(A_2) \) then \( c \) and \( d \) are strong Young winners of \( (D, W) \), and (b) if \( \kappa(A_2) > \kappa(A_1) \) then \( d \) is the unique strong Young winner of \( (D, W) \).

We now show how this proof can be adapted to work for Young elections. Please refer to [RSV03] for definitions and details of the construction, as we will only point out the differences here. Given an MSPC instance \( I = (B_1, B_2, A_1, A_2) \), we construct an election \( (C', V') \) such that \( c \) and \( d \) are designated candidates in \( C' \), and it holds that YoungScore\((C', c, V') = 2 \cdot \kappa(A_1) \) and YoungScore\((C', d, V') = 2 \cdot \kappa(A_2) \).

Let \( C' = C \) and \( V' = V - \{v_{(2,4)}, v_{(2,7)}\} \), where \( (C, V) \) is the election constructed in the proof of Theorem 2.3 of [RSV03], \( v_{(2,4)} \) is one of the two voters in \( V \) referred to as “voters of the form (2.4)” in [RSV03] and \( v_{(2,7)} \) is one of the two voters in \( V \) referred to as “voters of the form (2.7)” in [RSV03]. One can then define a submultiset \( V' \) of the voters \( V' \) as \( V' = V - \{v_{(2,4)}\} \), where \( V' \) is defined on page 381 of [RSV03]. Then, \( ||V'|| = 2 \cdot \kappa(A_1) \) and \( c \) is a weak Condorcet winner in \( (C', V') \), implying that YoungScore\((C', c, V') \geq 2 \cdot \kappa(A_1) \).

To show that YoungScore\((C', c, V') \leq 2 \cdot \kappa(A_1) \), we adapt Lemma 2.4 of [RSV03] in the following way. (The proof of Lemma B.3 is similar to the proof of Lemma 2.4 in [RSV03] and we omit it.)

---

\[\text{Note that in this proof, we follow Rothe et al.’s convention for Young scores and define the YoungScore}(C, c, V) \text{ of a candidate } c \text{ in an election } (C, V) \text{ as the size of the maximal submultiset of voters for which } c \text{ is a weak Condorcet winner. Hence, the set of Young winners consists of all candidates whose YoungScore is at least as large as the YoungScore of each other candidate.} \]
Lemma B.3. For any \( \lambda \) with \( 3 < \lambda \leq \| \mathcal{X}_1 \| + 1 \), let \( V_\lambda \) be a submultiset of \( V' \) such that \( V_\lambda \) contains exactly \( \lambda - 1 \) voters of the form (2.4) or (2.5) and \( c \) is a weak Condorcet winner in \((C', V')\). Then \( V_\lambda \) contains exactly \( \lambda - 1 \) voters of the form (2.3) and no voters of the form (2.6), (2.7), or (2.8). Moreover, the \( \lambda - 1 \) voters of the form (2.3) in \( V_\lambda \) represent pairwise disjoint sets from \( \mathcal{X}_1 \).

Let \( \hat{V}' \subseteq V' \) be such \( \| \hat{V}' \| \geq 2 \cdot \kappa(\mathcal{X}_1) \) and such that \( c \) is a weak Condorcet winner in \((C', \hat{V}')\). Suppose there are exactly \( \lambda - 1 \) voters of the form (2.4) or (2.5) in \( \hat{V}' \). Since \( \kappa(\mathcal{X}_1) > 2 \), it follows that \( \| \hat{V}' \| \geq 6 \), and thus \( \lambda > 3 \) in order for a to not beat \( c \). Also, \( \lambda - 1 \leq \| \mathcal{X}_1 \| \), since there are exactly \( \| \mathcal{X}_1 \| \) voters of the form (2.4) or (2.5) in \( V' \). Lemma B.3 then implies that there are exactly \( \lambda - 1 \) voters of the form (2.3) in \( \hat{V}' \), those voters represent pairwise disjoint sets from \( \mathcal{X}_1 \), and \( \hat{V}' \) contains no voters of the form (2.6), (2.7), or (2.8). Hence, \( \| \hat{V}' \| = 2 \cdot (\lambda - 1) \leq 2 \cdot \kappa(\mathcal{X}_1) \).

We thus have YoungScore\((C', c, V') = 2 \cdot \kappa(\mathcal{X}_1)\). Analogously, one can show that YoungScore\((C', d, V') = 2 \cdot \kappa(\mathcal{X}_2)\). Thus

\[
\kappa(\mathcal{X}_1) \geq \kappa(\mathcal{X}_2) \quad \text{if and only if} \quad \text{YoungScore}(C', c, V') \geq \text{YoungScore}(C', d, V'),
\]

which proves the \( \Theta_2^p \)-hardness of comparing Young scores. To show that the winner problem for Young elections is \( \Theta_2^p \)-hard, we alter the election \((C', V')\) in exactly the same way as \((C, V)\) is altered in Theorem 2.5 of [RSV03]. Let \((D', W')\) be the altered election. One can then show that the Young scores of \( c \) and \( d \) in \((D', W')\) are the same as in \((C', V')\), and that all other candidates have a Young score in \((D', W')\) of at most 2. Thus, since the Young scores of \( c \) and \( d \) are at least 6, comparing the Young scores of \( c \) and \( d \) in \((D', W')\) is tantamount to deciding if \( c \) is a Young winner.

Altogether, we have that (a) if \( \kappa(\mathcal{X}_1) \geq \kappa(\mathcal{X}_2) \) then \( c \) is a Young winner of \((D', W')\), and (b) if \( \kappa(\mathcal{X}_2) > \kappa(\mathcal{X}_1) \) then \( d \) is the unique Young winner of \((D', W')\). It follows that an MSPC-instance \( I \) is in MSPC if and only if \( c \) is a Young winner of \((D', W')\), implying \( \Theta_2^p \)-hardness of the Young winner problem in the nonunique-winner model. For the unique-winner model, we follow the argument from the proof of [HHR09, Theorem 2.3]: Observe that \( I \) is in MSPC if and only if \( d \) is not the unique winner of \((D', W')\). Thus the complement of the unique-winner problem for Young elections is \( \Theta_2^p \)-hard. Since \( \Theta_2^p \) is closed under complement, this proves that the unique-winner problem for Young elections is \( \Theta_2^p \)-hard as well.

\[\square\]

Theorem B.4. The winner problem for weakDodgson elections is \( \Theta_2^p \)-complete, both in the nonunique-winner model and the unique-winner model.
Proof. \( \Theta_2^p \)-completeness of the (nonunique) winner problem for Dodgson elections was shown in [HHR97]. The same proof also establishes \( \Theta_2^p \)-completeness in the unique winner model [HHR09]. The \( \Theta_2^p \) upper bound is easy to show, and the same argument can be used to show a \( \Theta_2^p \) upper bound for the weak-Dodgson winner problem (in both the nonunique-winner model and the unique-winner model). \( \Theta_2^p \)-hardness of the winner problem for Dodgson was shown by a reduction from the \( \Theta_2^p \)-hard problem Two Election Ranking (2ER, for short) [HHR97]: Given two Dodgson triples \((C, c, V)\) and \((D, d, W)\), where both \(|V|\) and \(|W|\) are odd and \(c \neq d\), is the Dodgson score of \(c\) in \((C, V)\) less than or equal to the Dodgson score of \(d\) in \((D, W)\)? Here, a Dodgson triple \((C, c, V)\) is an election \((C, V)\) with a designated candidate \(c \in C\).

The reduction from 2ER to the winner problem for Dodgson elections works by merging the elections \(E_1 = (C, V)\) and \(E_2 = (D, W)\) into a new election \(E_3 = (C', V')\) such that \(C \cup D \subseteq C'\) and the following properties are satisfied:

(i) Dodgson-Score\(_{E_3}(c) = \) Dodgson-Score\(_{E_1}(c) + 1\),

(ii) Dodgson-Score\(_{E_3}(d) = \) Dodgson-Score\(_{E_1}(d) + 1\), and

(iii) Dodgson-Score\(_{E_3}(x) \) > Dodgson-Score\(_{E_1}(c)\) for all \(x \in C' - \{c, d\}\).

Here, Dodgson-Score\(_E(x)\) denotes the minimal number of switches required to make candidate \(x\) a Condorcet winner in election \(E\). Thus, we immediately have that \(c\) is a Dodgson winner in \(E_3\) if and only if Dodgson-Score\(_{E_1}(c) \geq\) Dodgson-Score\(_{E_2}(d).\)

\(\text{14}\)We have noted a fixable problem in the construction from [HHR97]. The problem is at the end of the proof of [HHR97, Lemma 3.7 (page 822)], where after proving that Dodgson-Score\(_{E_1}(d_{[|D|-1]}) >\) Dodgson-Score\(_{E_1}(c)\), it is claimed that “The same argument applies to each candidate in \((C \cup D) \setminus \{c, d\}\).” Though this claim is clearly true for each candidate in \(D - \{d\}\), it is not true for each candidate in \(C - \{c\}\), since in the preference orders of voter groups (b) and (c), candidates in \(C - \{c\}\) are not separated from candidates in \(D\) by a set of separating candidates. As a consequence, it may be possible to make a candidate in \(C - \{c\}\) a Condorcet winner by less than \(|S|\)/2 switches in those voter groups.

This problem can be avoided by changing the \(t_1 < \cdots < t_{|T|} < c < s_1 < \cdots < s_{|S|} < c_1 < \cdots < c_{|C| - 1}\) prefix in the preference orders of the voters in groups (b) and (c) to \(c < t_1 < \cdots < t_{|T|} < c_1 < \cdots < c_{|C| - 1} < s_1 < \cdots < s_{|S|}\). Intuitively, the changed construction is more symmetrical than the original one, as the preferences of voter groups (a) and (b) are defined analogously, with the roles of \(C\) and \(D\) (and those of \(S\) and \(T\)) interchanged.

Using the proof of [HHR97, Lemma 3.7], it is easy to see that the three properties mentioned above hold. In particular, \(c\) is preferred to each \(t \in T\) by more than half of the voters, and so the Dodgson score of \(c\) does not change. For every candidate \(d' \in D\), \(d'\) does not occur in the prefix of the preference order that is changed and so the Dodgson score of \(d'\) does not change. It remains to show that Dodgson-Score\(_{E_1}(x) >\) Dodgson-Score\(_{E_1}(c)\) for all \(x \in T \cup S \cup (C - \{c\})\). From the proof of [HHR97, Lemma 3.7] we know that Dodgson-Score\(_{E_1}(c) \leq |S|/2\) and that \(|S| = |T|\). In order for \(t \in T\) to become a Condorcet winner, \(t\) needs to gain at least one vote over \(d\). Note that in the changed preferences, we need more than \(|S|\) switches to switch \(t\) beyond \(d\), and so the change in preferences will not lower the Dodgson score of \(t\) below
We now show how the approach from [HHR97] can be adapted to work for weakDodgson. First, observe that the problem Weak Two Election Ranking (w2ER), which is defined like 2ER except with “Dodgson score” replaced by “weakDodgson score,” inherits $\Theta^P_2$-hardness from 2ER because Dodgson scores and weakDodgson scores coincide for all instances with an odd number of voters. Also observe, by inspection of the $\Theta^P_2$-hardness proof from [HHR97], that 2ER and w2ER are still $\Theta^P_2$-hard if we restrict the problem to Dodgson triples with at least three voters. Let w2ER’ be the thus-restricted version of of w2ER.

Here is a reduction from w2ER’ to the winner problem for weakDodgson elections. Given two Dodgson triples $(C, d, V)$ and $(D, d, W)$, denote the corresponding elections by $E_1 = (C, V)$ and $E_2 = (D, W)$. For convenience, denote $v = \|V\|$ and $w = \|W\|$. Recall that both $v$ and $w$ are odd and at least three and assume without loss of generality that $C \cap D = \emptyset$ and that $v \geq w$. Define $m = v \cdot \|C\| + w \cdot \|D\|$ and observe that $m$ is a strict upper bound for all weakDodgson scores in $E_1$ and $E_2$: Even in the worst case (a candidate is least preferred by all voters), $\lceil \frac{m}{2}\rceil \cdot (\|C\| - 1) < m$ switches suffice to make a candidate a weak Condorcet winner in $E_1$ (by making it the top choice of $\lceil \frac{m}{2}\rceil$ voters), and an analogous statement holds for $E_2$.

We now define the new election $E_3 = (C', V')$. The set $C'$ of candidates of $E_3$ is defined as $C' = C \cup D \cup S \cup T \cup U$, where $C$ and $D$ are the candidates from elections $E_1$ and $E_2$ and $S = \{s_1, \ldots, s_m\}$, $T = \{t_1, \ldots, t_m\}$, and $U = \{u_1, \ldots, u_m\}$ are $3m$ new candidates.

In order to define the voters of $E_3$, we introduce the following abbreviations for notational convenience. For a (ordered) set $M = \{m_1, \ldots, m_k\}$, $a < M < b$ is shorthand for $a < m_1 < \cdots < m_k < b$. Furthermore, let $c_1, \ldots, c_{\|C\| - 1}$ and $d_1, \ldots, d_{\|D\| - 1}$ be arbitrary enumerations of $C_- = C - \{c\}$ and $D_- = D - \{d\}$, respectively. The voters $V'$ of $E_3$ consist of the following subgroups:

(a) For every voter in $E_1$ with preference order $C_j$ over $C$ (written from least preferred to most preferred), there is one voter with preference order

\[ U < d < S < D_- < T < C_i. \]

In order for $s \in S$ to become a Condorcet winner, it is shown in [HHR97] that $s$ needs to gain at least one vote over $c$ and that we need more than $\|S\|/2$ switches to do that. Since $s$ is preferred to $c$ by the voters in groups (b) and (c) in both the original and the changed election, these voters can not be used for $s$ to gain a vote over $c$. It follows that in the changed election, $s$ needs more than $\|S\|/2$ switches to become a Condorcet winner. Finally, let $c' \in C - \{c\}$. The argument from [HHR97] that every $d' \in D - \{d\}$ needs to gain at least one vote over $c$ in order to become a Condorcet winner, and that we need more than $\|S\|/2$ switches to do that can be used (by symmetry of the construction) to show that every $c' \in C - \{c\}$ needs to gain at least one vote over $d$ in order to become a Condorcet winner, and that we need more than $\|S\|/2$ switches to do that.

15If the number of voters is odd, being a weak Condorcet winner is tantamount to being a Condorcet winner, since there are no ties in pairwise comparisons.
(b) For every voter in $E_2$ with preference order $D_j$ over $D$, there is one voter with preference order

$$U < c < T < C_− < S < D_j.$$ 

(c) There are $\left\lceil \frac{v}{2} \right\rceil - \left\lceil \frac{w}{2} \right\rceil$ voters with preference order

$$U < c < T < C_− < S < D_− < d.$$ 

d) There are $\left\lceil \frac{v}{2} \right\rceil$ voters with preference order

$$T < C_− < D_− < S < U < c < d.$$ 

e) There are $\left\lceil \frac{w}{2} \right\rceil - 1$ voters with preference order

$$T < C_− < D_− < S < U < d < c.$$ 

(f) There is one voter with preference order

$$U < d < c < T < C_− < D_− < S.$$ 

Since $v$ is odd, the total number of voters is $\|V\| = 2v + w + 1$. As both $v$ and $w$ are odd, $\|V\|$ is even and a weak Condorcet winner has to be preferred to every other candidate by at least $\|V\|_2 = v + \left\lceil \frac{w}{2} \right\rceil$ voters. We now show that the following three properties are satisfied:

(i) weakDodgson-Score$_{E_3}(c) =$ weakDodgson-Score$_{E_1}(c)$,

(ii) weakDodgson-Score$_{E_3}(d) =$ weakDodgson-Score$_{E_1}(d)$, and

(iii) weakDodgson-Score$_{E_3}(x) >$ weakDodgson-Score$_{E_1}(c)$ for all $x \in C' - \{c,d\}$.

For (i), observe that $c$ is preferred to every candidate in $C' - C$ by at least $\|V\|_2$ of the voters. Thus, in computing the weakDodgson score of $c$, we only have to take care of candidates in $C_− = \{c_1, \ldots, c_{\|C\|−1}\}$. Let $x_i$ be the number of voters in group (a) that prefer $c$ to $c_i$. Then, the number of voters in $V'$ that prefer $c$ to $c_i$ is $x_i + \left\lceil \frac{v}{2} \right\rceil + \left\lceil \frac{w}{2} \right\rceil - 1$. Candidate $c$ is a weak Condorcet winner in $E_3$ if and only if this number is greater than or equal to $\|V\|_2 = v + \left\lceil \frac{w}{2} \right\rceil$, and this is the case if and only if $x_i \geq \left\lceil \frac{v}{2} \right\rceil$ for each $i \in \{1, \ldots, \|C\|−1\}$. But this means that $c$ is a Condorcet winner in $E_1$. By definition, this can be achieved by $k$ switches, where $k =$ weakDodgson-Score$_{E_1}(c)$. We have thus shown the upper bound weakDodgson-Score$_{E_3}(c) \leq$ weakDodgson-Score$_{E_1}(c)$. Now assume that weakDodgson-Score$_{E_3}(c) <$ weakDodgson-Score$_{E_1}(c)$. Due to the construction, all the switches in
an optimal sequence occur in voters of group (a), as making \( c \) beat any candidate in \( C_- \) would require more than \( m \) switches in all the other relevant voter groups (b), (c), and (f). This means that there is a way to make \( c \) a weak Condorcet winner in \( E_1 \) with less than weakDodgson-Score\(_{E_1}(c) \) switches, a contradiction. We have thereby shown that weakDodgson-Score\(_{E_1}(c) = \) weakDodgson-Score\(_{E_1}(c) \). The proof of property (ii) is analogous.

For (iii), recall that \( m \) was chosen sufficiently large to be a strict upper bound on weakDodgson-Score\(_{E_1}(c) \) and thus, by (i), on weak Dodgson-Score\(_{E_1}(c) \). We now show that all candidates in \( C' \) other than \( c \) and \( d \) have a weak Dodgson score of at least \( m \) in \( E_3 \). Consider a candidate \( s \in S \). In order to become a weak Condorcet winner, \( s \) must in particular beat or tie \( c \) in their pairwise contest. As \( s \) is preferred to \( c \) by only \( w + \left[ \frac{v}{2} \right] - \left[ \frac{w}{2} \right] + 1 = \left[ \frac{v}{2} \right] + \left[ \frac{w}{2} \right] \) voters in \( V' \), it needs to gain at least \( \left[ \frac{v}{2} \right] \) extra votes over \( c \) in voter groups (a), (d), and (e). But gaining just one extra vote over \( c \) would require more than \( m \) switches, because \( s \) is separated from \( c \) by at least \( m \) other candidates in all these voter groups.

A similar argument applies to all other candidates: Candidates in \( T \) need \( \left[ \frac{v}{2} \right] \) extra votes over \( d \) in (b), (c), (d), and (e), but one extra vote requires more than \( m \) switches in each of these voters. Candidates in \( U \) need \( \left[ \frac{v}{2} \right] \) extra votes over \( d_i \in D_- \) in (a), (b), (c), and (f), but one extra vote requires more than \( m \) switches. Candidates in \( D_- \) need \( \left[ \frac{v}{2} \right] \) extra votes over \( c \) in (a), (d), and (e), but one extra vote requires more than \( m \) switches. Candidates in \( C_- \) need \( \left[ \frac{w}{2} \right] \) extra votes over \( d \) in (b), (c), (d), and (e), but one extra vote requires more than \( m \) switches. Thus, we have shown that weakDodgson-Score\(_{E_3}(x) > m > \) weakDodgson-Score\(_{E_3}(c) \) for all \( x \in C' - \{c, d\} \).

It is now easy to see that (1) if weakDodgson-Score\(_{E_1}(c) \) \( \leq \) weakDodgson-Score\(_{E_2}(d) \), then \( c \) is a weak Dodgson winner in \( E_3 \), and (2) if weakDodgson-Score\(_{E_1}(c) \) \( > \) weakDodgson-Score\(_{E_2}(d) \), then \( d \) is the unique weak Dodgson winner in \( E_3 \). Let \( I = ((C, c, V), (D, d, W)) \) be an instance of w2ER'. We have just argued that \( I \) is in w2ER' if and only if \( c \) is a weak Dodgson winner in \( E_3 \), which immediately implies \( \Theta_2^n \)-hardness of the weak Dodgson winner problem in the nonunique-winner model. For the unique-winner model, we follow the argument from the proof of [HHR09, Theorem 2.3]: Observe that \( I \) is in w2ER' if and only if \( d \) is not the unique weak Dodgson winner in \( E_3 \). Thus the complement of the unique-winner problem for weak Dodgson elections is \( \Theta_2^n \)-hard. Since \( \Theta_2^n \) is closed under complement, this proves that the unique-winner problem for weak Dodgson elections is also \( \Theta_2^n \)-hard.
B.4 Nonspotlight Proofs for Section 4

Proof of Theorem 4.3. The polynomial-time algorithms witnessing P-membership of approval-negative-bribery and approval-strongnegative-bribery have the same flavor as the algorithm for approval-bribery (Theorem 4.2), and we present them first.

Let \((C, V)\) be an instance of a single-peaked approval election and \(L\) a valid single-peaked ordering of the candidates. We have to decide whether a designated candidate \(p \in C\) can be made a winner by bribing \(\leq k\) voters. As in the proof of Theorem 4.2, we define \(V_c\) as the multiset of voters who approve of candidate \(c\). \(V_+ = V_p\) is the multiset of voters approving \(p\) and \(V_- = V - V_+\) is the multiset of voters disapproving \(p\).

Approval-strongnegative-bribery is like approval-bribery, except that bribed voters after the bribe are not allowed to approve of \(p\). Consequently, bribing a voter who approves of \(p\) is always pointless and we will bribe only voters from \(V_-\). Also, we can without loss of generality assume that all bribed voters disapprove of all candidates after the bribe, as this is clearly the best possible action with regard to our goal to make \(p\) a winner. The algorithm is similar to the one presented in the proof of Theorem 4.2. Define the surplus \(n(c)\) of a candidate \(c \in C - \{p\}\) as \(n(c) = \|V_c\| - \|V_p\|\) and consider the rightmost candidate \(c'\) to the right of \(p\) that has a positive surplus. In order to make \(p\) a winner, we obviously have to bribe at least \(n(c')\) voters from \(V_{c'} \cap V_-\). By definition, all candidates to the right of \(c'\) have a nonpositive surplus, and that is why—in deciding which \(n(c')\) voters from \(V_{c'} \cap V_-\) to bribe—we can solely focus on candidates to the left of \(c'\) and choose the \(n(c')\) voters from \(V_{c'} \cap V_-\) whose approval range extends the furthest to the left. As mentioned above, we bribe these voters to disapprove of everyone, thereby making \(n(c') = 0\). We then recalculate the surpluses of all candidates to the left of \(c'\) (note that a candidate’s surplus never grows and we can thus ignore the candidates to the right of \(c'\)) and repeat the process with the now rightmost candidate to the right of \(p\) that has a positive surplus.

By the time when all candidates to the right of \(p\) have a nonpositive surplus, we will mirror the societal order \(L\) and repeat the same procedure for all nonpositive-surplus candidates that were originally to the left of \(p\) with respect to \(L\). If we can make all surpluses nonpositive without exceeding the bribe limit of \(k\), we have found a successful bribery action. Otherwise, a successful bribery action does not exist, as all our decisions (concerning which voters to bribe) were provably optimal.

In the case of approval-negative-bribery, bribed candidates may approve of \(p\) after the bribe only if they approved of \(p\) before the bribe. In this model, it does sometimes make sense to bribe voters from \(V_+\). But that does not pose a problem, as the following observation shows. The approval score
of \( p \) remains unchanged for every optimal bribe. Here, an optimal bribe is defined as either bribing a voter from \( V_+ \) to disapprove of everyone except \( p \), or bribing a voter from \( V_- \) to disapprove of everyone. Again, as in the case of strongnegative-bribery, we can without loss of generality assume that all our bribes are optimal. Thus, the observation tells us that we do not have to differentiate between bribing voters from \( V_+ \) and \( V_- \), as in both cases the only thing we care about is the removal of approvals of candidates other than \( p \). Hence, the algorithm is the same as for strongnegative bribery, except that when considering a positive-surplus candidate \( c \) and deciding which voters from \( V_c \) to bribe, we not only consider voters that disapprove of \( p \) but also voters that approve of \( p \).

We now go on to show that both bribery problems are NP-complete in the general (i.e., not necessarily single-peaked) case. Membership in NP is obvious for both problems.

The hardness proof of approval-strongnegative-bribery is an adaptation from the proof that approval-bribery is NP-hard (Theorem 4.2 of [FHH09]). Please refer to that proof as we here only point out the differences. In the reduction from the NP-hard problem Exact Cover by 3-sets (X3C), define the bribery instance as in Theorem 4.2 of [FHH09], except that the number of voters who approve only of \( p \) is changed from \( m - t \) to instead \( m \). Then, if there exists a cover \( A \) with \( \|A\| = t \) and \( \bigcup_{i \in A} S_i = B \), we bribe each \( v_i \) with \( i \in A \) to approve of zero candidates (this is a slight additional change). So all candidates tie at \( m \) approvals and all win. Looking at the other direction, if there exists a successful bribe of \( t \) voters, then since \( p \) has \( m \) approvals and each of the 3\( t \) candidates in \( B \) has \( m + 1 \) approvals, and no bribe can increase the number of approvals of \( p \) (in both the negative and the strongnegative bribery setting), then clearly \( p \) still has \( m \) approvals after the bribes. So, each of the 3\( t \) candidates in \( B \) (that started with \( m + 1 \) approvals) must have lost exactly one approval (if any lost more than one approval, one would have lost zero approvals and would beat \( p \); if any lost zero approvals, it would beat \( p \)). So, we have an exact cover by 3-sets.

Due to the construction, one can use the same proof to show NP-hardness of approval-negative-bribery: the only voters approving \( p \) are voters who approve only \( p \) and are obviously never bribed.

\[\square\]

**Proof of Theorem 4.4.** The general setting is the same in all of the eight bribery problems and we describe it here. Let \((C, V)\) be an election instance and let \( L \) be a linear order over the candidates such that the electorate is single-peaked with respect to \( L \). The question is whether a designated candidate \( p \in C \) can be made a weakCondorcet winner by bribing at most \( k \) voters.

If \( p \) is a weakCondorcet winner of the election \((C, V)\), a successful bribery action is obviously possible as we do not have to bribe any voter. We can thus focus on the case where \( p \) is not a
weakCondorcet winner of \((C,V)\). In this case, the definition of weakCondorcet implies that \(p\) is not contained in the “median interval” (i.e., \(p \notin [m_L, m_r]_L\), where \(m_L\) and \(m_r\) are the top choices of the median voters). Assume without loss of generality that \(p\) lies strictly to the right of the median interval, i.e., \(m_r \leq p \leq m_L\) (otherwise, we can just mirror the societal order \(L\)).

Identify each voter with his or her most preferred candidate. Define \(V_L = \{ v \in V \mid vLp \}\) as the multiset of voters lying to the left of \(p\) with respect to \(L\) and \(V_r = V - V_L = \{ v \in V \mid v = p \text{ or } pLv \}\) as the multiset of voters lying on \(p\) or to the right of \(p\). In settings where voters have no weights, \(m_r \leq p \leq m_L\) immediately implies that \(\|V_L\| > \|V_r\|\) and that, in order to make \(p\) a weakCondorcet winner, we need to bribe \(\left\lceil \frac{\|V_L\| - \|V_r\|}{2} \right\rceil\) voters from \(V_L\) to move to \(p\) (or, in the negative-bribery settings, to the candidate to the right of \(p\)). If voters have weights, we have to shift a total weight of at least \(\left\lceil \frac{w(V_L) - w(V_r)}{2} \right\rceil\), where \(w(V')\) for a submultiset \(V' \subseteq V\) of voters is defined in the natural way as the sum of the weights of voters contained in \(V'\), i.e., \(w(V') = \sum_{v \in V'} w(v)\).

Observe that all eight bribery problems are easily seen to be in NP. We now prove the assertions of Theorem 4.4 in the order in which they are mentioned in the statement of the theorem.

(i) weakCondorcet-weighted-$bribery is NP-complete.

Define \(k\) as the weight that needs to be shifted from \(V_L\) to \(p\), i.e.,

\[ k = \left\lceil \frac{w(V_L) - w(V_r)}{2} \right\rceil. \]

The problem weakCondorcet-weighted-$bribery can now be stated as follows. We are given a collection of objects (voters), each with a positive integer weight and a positive integer price, and bounds \(k\) and \(k\), and the question is whether there exists a subset whose price is \(\leq k\) and whose weight is \(\geq k\). If so, bribe them all to first-choose \(p\). If this holds (or if \(p\) initially wins), we succeed, else we fail. It is straightforward to show that the NP-hard problem Knapsack (see, e.g., [GJ79]) reduces directly to this problem, establishing the NP-hardness of weakCondorcet-weighted-$bribery.

(ii) weakCondorcet-negative-weighted-$bribery is NP-complete.

We give a reduction from the NP-complete problem Partition (see Section A.4). Given a collection \((k_1, k_2, \ldots, k_n)\) of positive integers that sum to \(2K\), define the single-peaked election \((C,V)\) with \(C = \{a, p, c\}, aLpLc,\) and for each \(k_i\) we have one voter \(v_i\) whose first choice is \(a\) and whose weight is \(k_i\). Set the budget \(k\) equal to \(n\). It is easy to see that \(p\) can win via negative bribery if and only if \((k_1, k_2, \ldots, k_n)\) can be partitioned into two equal-sum parts.

(iii) weakCondorcet-negative-weighted-$bribery is NP-complete.
This follows from (ii).

(iv) weakCondorcet-bribery is in P.

The algorithm is easy: Bribe $k$ voters chosen arbitrarily from $V$ to make $p$ their top choice. If, after the bribe, $p$ is a weakCondorcet winner, we were successful. Otherwise, no successful bribery action exists.

(v) weakCondorcet-$b$ribery is in P.

The algorithm is the same as in (iv), except that we bribe voters in the order of their price tags, starting with the cheapest voter.

(vi) weakCondorcet-weighted-bribery is in P.

The algorithm is the same as in (iv), except that we bribe voters in the order of their weights, starting with the voter that has the highest weight.

(vii) weakCondorcet-negative-bribery is in P.

This follows from (viii).

(viii) weakCondorcet-negative-$b$ribery is in P.

Recall that $p$ lies strictly to the right of the median interval. In the case of negative bribery, bribed voters must not have $p$ as their top choice. Thus, $p$ can never be made a weakCondorcet winner if either (a) no voter has $p$ as his or her first choice and $|V|$ is odd, or (b) $p$ is on the right end of the societal order $L$.

Otherwise, let $\hat{p}$ be the candidate to the right of $p$ with respect to $L$. Successful bribery is possible if and only if, by greedily bribing voters from $V$ (starting with the cheapest voter) to have $\hat{p}$ as their top choice, we can make $p$ a weakCondorcet winner.

B.5 Nonspotlight Proofs for Section 5

Let $T$ be one of the 22 control types defined in [FHHR09a], i.e., the eleven types (adding/deleting candidates/voters (4 types); “unlimited-adding” of candidates (1 type); and three partition types each with TP and TE (6 types)), each for both the constructive cases (making $c$ win or uniquely win) and the destructive cases (making $c$ not win or not be a unique winner). Let $E$ be any election system. The following holds for both the unique-winner model and the nonunique-winner model.
Theorem B.5. The T control problem for E is in P for single-peaked electorates in our default model (in which the societal order L is part of the input) if and only if the T control problem for E is in P for single-peaked electorates in the “exists-L” model in which we ask if there exists some order L relative to which the entire input is single-peaked and is such that the goal can be achieved by a type T control action.

Proof. Note that for any control problem instance and any L′ and L′′ that are valid societal orders relative to the instance (and here it is crucial that our notion is that all of the instance—not just the voters/candidates left in at the end—must respect an order for it to be valid), then the set of successful (resp., unsuccessful) control actions under L′ is exactly the same as under L′′. Thus, we have the following key observation:

(⋆) There exists a valid single-peaked order relative to which the control problem instance can be successfully controlled if and only if the control problem instance can be successfully controlled for every valid single-peaked order.

So, if a control problem is in P for the model in which L is part of the input via algorithm A, then for the exists-L model, we can on a given instance I compute a valid L (e.g., using the algorithm by Escoffier, Lang, and Öztürk [ELÖ08]) and hand it to A. By (⋆) and the correctness of A, we know that A gives the correct answer.

As to the other direction, if a control problem is in P for the exists-L model, then thanks to (⋆), we can simply strip L off the input in the model in which L is part of the input and can safely (knowing the answer is correct for our original issue) ask the existential question to the hypothetical P algorithm for the exists-L version.

Some particular instances of Theorem B.5 were argued directly for particular cases of E and T by Faliszewski et al. [FHR11], but Theorem B.5 provides a tool that removes the need for case-specific arguments.

Proof of Theorem 5.4. We here give algorithms for the nonunique-winner model. For the unique-winner model, see the proof of Theorem 5.6 (by Fact 3.1 on page 10, being a unique weakCondorcet winner is tantamount to being a Condorcet winner for single-peaked electorates).

Associate each voter with his or her most preferred candidate. Our goal is to make p a weak-Condorcet winner, i.e., we want to end up in one of the following two situations:

1. ||V|| is odd and the median voter has p as his or her top choice.
2. \( ||V|| \) is even and \( p \) lies in the (inclusive) interval \([m_\ell,m_r]_L\), where \( m_\ell \) and \( m_r \) are the median voters. (This includes the case where \( m_\ell = m_r \).) In the following, we will refer to \([m_\ell,m_r]_L\) as the “median interval.”

Here is an easy algorithm for the case of control by addition of voters: See where the current median (or median interval) is. If \( p \) is the median (or if \( p \) lies in the median interval), we are done. Otherwise, assume without loss of generality that the median (interval) lies to the left of \( p \). Then, add unregistered voters whose top choice is either \( p \) or some candidate to the right of \( p \) until \( p \) is a weak Condorcet winner or we hit our addition bound or run out of voters to add of this sort. If at this point we have not succeeded, success is impossible.

The algorithm for the control-by-deletion-of-voters case is similar: See where the current median (or median interval) is. If \( p \) is the median (or if \( p \) lies in the median interval), we are done. Otherwise, we need to shift the location of the median voter(s) towards \( p \). Without loss of generality assume that the median (interval) lies to the left of \( p \). Then, we start deleting voters at the left end of \( L \) until we make \( p \) a weak Condorcet winner or we hit our deletion bound without success.

**Proof of Theorem 5.5.** Let \((C,V)\) be an instance of a Condorcet election and let \( L \) be a linear order with respect to which the electorate is single-peaked. Without loss of generality assume \( ||C|| \geq 2 \) (otherwise, the problem is trivial). Furthermore, \( p \in C \) is a designated candidate and the question is whether there exists some partition \((V_1,V_2)\) such that \( p \) is a Condorcet winner of the overall election. It is clear that this is the case if and only if there exists a partition \((V_1,V_2)\) such that \( p \) is a Condorcet winner in \((C,V_1)\) and \((C,V_2)\) has no Condorcet winner, a Condorcet winner that \( p \) beats in a pairwise comparison, or \( p \) itself as a Condorcet winner.

We show that Algorithm 3 returns a partition with this property whenever one exists. Algorithm 3 loops over all candidates \( c \) that \( p \) beats in a pairwise comparison and over all possible sizes of \( V_1 \) (line 2). Then, the set \( C \) of candidates is divided into five regions \( R_1,R_2,\ldots,R_5 \) that are defined as follows. Without loss of generality assume that \( p <_L c \), i.e., \( c \) lies to the right of \( p \) with respect to the societal order \( L \) (otherwise, just mirror everything). Region \( R_1 \) consists of all candidates to the left of \( p \), i.e., \( R_1 = \{ x \in C \mid x <_L p \} \), and region \( R_2 \) consists of \( p \) only. Similarly, \( R_5 \) consists of all candidates to the right of \( c \) and \( R_4 \) consists of \( c \) only. Finally, \( R_3 \) consists of all remaining candidates, namely, the candidates that lie between \( p \) and \( c \) with respect to \( L \). Note that all regions except \( R_2 \) and \( R_4 \) may be empty. We have the following picture:
Algorithm 3 Condorcet Control by Partition of Voters

1: for all candidates $c$ that lose against $p$ in a pairwise comparison do
2:     for all $k \in \{1, 2, \ldots, \|V\|\}$ do
3:         for all $\ell = (\ell_1, \ell_2, \ldots, \ell_5) \in \mathbb{N}^5$ with $\sum_{i=1}^5 \ell_i = k$ do
4:             define $\mathcal{P}_\ell$ as the set of partitions $(V_1, V_2)$ of $V$ such that for each $i$, $\ell_i$ is the number of voters in $V_1$ whose top choice is in $R_i$
5:             if $\mathcal{P}_\ell \neq \emptyset$ then
6:                 let $(V_1, V_2)$ be an arbitrary partition from $\mathcal{P}_\ell$
7:                     if $p$ is a Condorcet winner of $(C, V_1)$ then
8:                         if either $c$ or $p$ is a Condorcet winner of $(C, V_2)$ then
9:                             return $(V_1, V_2)$
10:                        else if $\exists (V_1, V_2) \in \mathcal{P}_\ell$: $(C, V_2)$ has no Condorcet winner then
11:                             return $(V_1, V_2)$
12:             return “there is no partition that makes $p$ an overall Condorcet winner”

Associate each voter with the candidate at the top of that voter’s preference order. We now define $\mathcal{P}_\ell$, a set of partitions of $V$ with respect to the regions just defined. Let $\ell = (\ell_1, \ell_2, \ldots, \ell_5)$ be a five-dimensional vector of natural numbers. Define $\mathcal{P}_\ell$ as the set of all partitions $(V_1, V_2)$ of $V$ that have the property that, for each $i$, $\ell_i$ is the number of voters in $V_1$ whose top choice is in $R_i$.

For $\mathcal{P}_\ell \neq \emptyset$, the key observation is the following: For $x \in \{p, c\}$ it holds that $x$ is a Condorcet winner in some election $(C, V_1)$ with $(V_1, V_2) \in \mathcal{P}_\ell$ if and only if $x$ is a Condorcet winner in every election $(C, V_1)$ with $(V_1, V_2) \in \mathcal{P}_\ell$. That is, if we want to check whether $x$ is a Condorcet winner in any of the primary elections $(C, V_1)$ induced by any of the partitions $(V_1, V_2) \in \mathcal{P}_\ell$, it suffices to check just one of them to obtain the answer. By symmetry, the same statement holds for Condorcet winners of the elections $(C, V_2)$.

The reason why this is true is that, given the number of voters in each region, it is easy to compute the region(s) of the median voter(s) (just by counting). Since both $p$ and $c$ constitute a region on their own, it is equally easy to tell whether they are Condorcet winners (using Fact 3.1).

We have just shown that the queries in lines 7 and 8 of Algorithm 3 can be efficiently answered. We now go on to show how the query in line 10 can be efficiently answered, i.e., given $\ell$, is there
a partition \((V_1, V_2) \in \mathcal{P}_t\) such that \((C, V_2)\) has no Condorcet winner. Clearly, this cannot happen if there are an odd number of voters in \(V_2\). So assume \(\|V_2\|\) to be even and let \(m_1\) and \(m_2\) be the median voters. For each region \(R_i\), we know the number of \(V_2\)-voters whose top choice is in \(R_i\) (this number is \(\|R_i\| - \ell_i\)). Thus we know in which regions the median voters fall (again just by counting). Now, if at least one of \(m_1\) or \(m_2\) fall in \(R_2\) or \(R_4\) (i.e., are \(p\) or \(c\)), then \((C, V_2)\) cannot possibly have a Condorcet winner other than \(p\) or \(c\) (there may be no Condorcet winners). In any of these three cases, our partition action was successful\(^\text{16}\) and we can return an arbitrary partition in \(\mathcal{P}_t\).

The only remaining cases are that \(m_1\) and \(m_2\) both fall in \(R_1 \cup R_3 \cup R_5\). If \(m_1\) and \(m_2\) fall in different regions, there can obviously be no Condorcet winner and we are done. Assume, therefore, that \(m_1\) and \(m_2\) both fall in \(R_i\) with \(i \in \{1, 3, 5\}\). Our goal is to assign \(q_i = \|R_i\| - \ell_i\) voters with top choice in \(R_i\) to \(V_2\) in such a way that the median pair in \(V_2\) does not fall on the same candidate.

Let the median pair be the \(r\)th and \((r + 1)\)st \(V_2\)-voter in \(R_i\). Here, \(r\) \((1 \leq r \leq q_i - 1)\) is known (by the numbers of \(V_2\)-voters to the left and right of \(R_i\)) and we count from left to right with respect to the societal order \(L\). We will try to accomplish our goal by brute force, namely, for each pair of candidates \((d_r, d_l)\) in \(R_i\) with \(d_l \leq L d_r\), let us try to ensure that the \(r\)th \(V_2\)-voter in \(R_i\) from the left falls on or left of \(d_l\) and the \((r + 1)\)st falls on or right of \(d_r\). We can do this if and only if \(\|\{x \in V \mid x \text{ falls in } R_i \land x \leq L d_l\}\| \geq r\) and \(\|\{x \in V \mid x \text{ falls in } R_i \land d_r \leq L x\}\| \geq q_i - r\). The cost of this check is in \(\mathcal{O}(\|C\|^2)\), as for each pair of candidates in \(R_i\), we do some easy counting.

Summing up, the running time of Algorithm 3 can be bounded as follows. The number of iterations for the loops in lines 1, 2, and 3, are bounded by \(\|C\|\), \(\|V\|\), and \(\|V\|^5\), respectively. The cost of one iteration of the inner loop is clearly dominated by the cost of answering the query in line 10. This cost is bounded by \(\mathcal{O}(\|C\|^2)\), as argued in the last paragraph. Altogether, this yields a running time that is obviously polynomial in the size of the input.

Correctness of Algorithm 3 should be clear from the explanations above, as we have argued that we find a partition that makes \(p\) an overall Condorcet winner if and only if such a partition exists. In particular, observe that we never need to consider cases with \(k = 0\) as \(p\) can never be a Condorcet winner of \((C, \emptyset)\). And the case where \(p\) is already a Condorcet winner of the original election \((C, V)\) is handled by setting \(k\) to \(\|V\|\) in line 2 (and \(c\) to some arbitrary candidate from \(C - \{p\}\) in line 1).

\[\square\]

**Proof of Theorem 5.6.** Associate each voter with his or her most preferred candidate. Our goal is to make \(p\) a Condorcet winner, i.e., we want to end up in a situation where \(p\) is the most preferred

\(^{16}\)In fact, if either \(p\) or \(c\) is a Condorcet winner in \((C, V_2)\), this would already have been detected in line 8.
candidate of the median voter(s).

Here is an easy algorithm for the case of control by addition of voters: See where the current median (or median pair) is. If it is on $p$, we are done. Otherwise, add unregistered voters whose top choice is $p$ until we run out of those or succeed or hit our addition bound. If we succeed or hit our addition bound, we are done (with or without success). If we have not yet hit our addition bound, move on as follows. If $p$ is strictly between two median voters at this point, success is impossible. But if it is one of two distinct median voters (without loss of generality say it is the rightmost of the two median voters) or the median interval does not contain $p$ (without loss of generality say the interval lies to the left of $p$), then add unregistered voters to the right of $p$ until we either make $p$ a Condorcet winner or hit our addition bound or run out of voters to add of this sort. If at this point we have not succeeded, success is impossible.

The algorithm for the control-by-deletion-of-voters case is similar: See where the current median (or median pair) is. If it is on $p$, we are done. Otherwise, we need to shift the location of the median voter(s) towards $p$. Without loss of generality assume that the median (or median interval) lies to the left of $p$. Then, we start deleting voters at the left end of $L$ until we make $p$ a Condorcet winner or we hit our deletion bound without success. Note that if $p$ initially lies in between the two median voters, it can never be made a Condorcet winner by deleting voters.