

Protecting Elections by Recounting Ballots*

Edith Elkind¹, Jiarui Gan¹, Svetlana Obraztsova²,
 Zinovi Rabinovich² and Alexandros A. Voudouris¹

¹Department of Computer Science, University of Oxford

²School of Computer Science and Engineering, Nanyang Technological University

Abstract

Complexity of voting manipulation is a prominent topic in computational social choice. In this work, we consider a two-stage voting manipulation scenario. First, a malicious party (an attacker) attempts to manipulate the election outcome in favor of a preferred candidate by changing the vote counts in some of the voting districts. Afterwards, another party (a defender), which cares about the voters' wishes, demands a recount in a subset of the manipulated districts, restoring their vote counts to their original values. We investigate the resulting Stackelberg game for the case where votes are aggregated using two variants of the Plurality rule, and obtain an almost complete picture of the complexity landscape, both from the attacker's and from the defender's perspective.

1 Introduction

Democratic societies use elections to select their leaders. However, in societies without a strong democratic tradition, elections may be used as a way to legitimize the status quo: voters are asked to cast their ballots, but the election authorities do not count these ballots correctly, in order to produce an outcome that favors a specific candidate. There are multiple reports of such cases in Russia¹, Congo² and Colombia³, as well as a number of other countries. Even when the election authorities are trustworthy, election results may be corrupted by an external party, for instance, by means of hacking electronic voting machines [Springall *et al.*, 2014; Halderman and Teague, 2015].

There are several ways to counteract electoral fraud. One approach is to send observers to polling stations, to ensure that only eligible voters participate in the elections and their ballots are counted correctly. However, it may be infeasible for the party that wants to protect the elections (the *defender*) to send observers to all polling stations. Consequently, the election manipulator (the *attacker*) may observe which polling stations remain unprotected, and focus their effort on these stations. Thus, under this approach the attacker benefits from the second-mover advantage.

An alternative approach that the defender can explore is to request recounts in some of the voting districts. While recounts cannot protect from all forms of attacks on election integrity (e.g., a recount is of limited use if voters have been bribed to vote in a specific way, or if the polling station has been burned down), they are feasible in a range of settings and offer the defender the second-mover advantage.

*Edith Elkind and Alexandros A. Voudouris are supported by the European Research Council (ERC) under grant number 639945 (ACCORD). Jiarui Gan is supported by the EPSRC International Doctoral Scholars Grant EP/N509711/1. Svetlana Obraztsova is supported by the MOE AcRF-T1-RG23/18 grant. Zinovi Rabinovich is supported by the NTU SUG M4081985 grant.

¹<https://reut.rs/2Gf2FD5>

²<https://on.ft.com/2SW7ggy>

³<https://colombiareports.com/voting-fraud-in-colombia-how-elections-are-rigged/>

		Plurality over Voters (PV)	Plurality over Districts (PD)	
		Unweighted	Weighted	
REC	NP-c, Thm. 3.1 (i) ③	P, Thm. 4.3	NP-c, Thm. 4.1 (i) ③	
	NP-c, Thm. 3.1 (ii) ④ $O(n^{m+2})$, Thm. 3.2		NP-c, Thm. 4.1 (ii) ④ $O(n^{m+2})$, Thm. 4.2	
MAN	NP-h, Thm. 3.3 (i) ③ ① ⑧	NP-c, Thm. 4.8 ④	Σ_2^P -c, Thm. 4.6 ③	
	NP-h, Thm. 3.3 (ii) ④ ① ⑧		NP-h, Thm. 4.7 ④ ①	

Table 1: Summary of our complexity results. MAN denotes the attacker’s problem, and REC denotes the defender’s problem. Hardness results with ④ hold even when the input is given in unary (the default is binary); with ③ hold even for three candidates; with ① hold even when the defender’s budget is zero; with ⑧ hold even when the attacker can change as many votes as she wants in each district.

Indeed, there are several examples where a recount changed the election outcome. For instance, in the 2008 United States Senate election in Minnesota the Democratic candidate Al Franken won the seat after a recount revealed that 953 absentee ballots were wrongly rejected⁴, and in the 2004 race for governor in Washington the Democratic candidate Gregoire was declared the winner after three consecutive recounts⁵.

However, recounts can be costly. In Gregoire’s case, the Democratic party paid \$730000 for a statewide manual recount, and in the 2016 US Presidential Election the fee to initiate a recount in Wisconsin was \$3.5 million. Thus, a party that would like to initiate a recount in order to rectify the election results should allocate its budget carefully. Of course, the attacker also incurs costs to carry out the fraud: local election officials may need to be bribed or intimidated, and the more districts are corrupted, the higher is the risk that the election results will not be accepted.

Our Contribution. In this paper we analyze the strategic game associated with vote recounting. In our model, there are two players: the attacker, who modifies some of the votes in order to make his preferred candidate p the election winner, and the defender, who observes the attacker’s actions and tries to restore the correct outcome (or, more broadly, to ensure that a candidate who is better than p wins the election) by means of recounting some of the votes. We assume that the set of voters is partitioned into electoral districts, and both the defender and the attacker make their choices at the level of districts rather than individual votes. The attacker selects a subset of at most B_A districts and changes the vote counts in the selected districts, and the defender can then restore the vote counts in at most B_D districts to their original values. We assume that both players have full information about the true votes and each other’s budgets, and the defender can observe the attacker’s actions. While the full information assumption is not entirely realistic, we note that in a district-based model both parties only need to know the vote counts in each district rather than individual votes, and one can get fairly accurate district-level information from independent polls. Also, verifying whether the votes in a district have been tampered with is possible using risk-limiting audits Lindeman and Stark [2012]; Schürmann [2016].

For simplicity, we focus on the Plurality voting rule, where each voter votes for a single candidate. We consider two implementations of this rule: (1) Plurality over Voters, where districts are only used for the purpose of collecting the ballots and the winner is selected among the candidates that receive the largest number of votes in total, and (2) Plurality over Districts, where each district selects a preferred candidate using the Plurality rule, and the overall winner is chosen among the candidates supported by the largest number of districts; we also consider a variant of the latter rule where districts have weights,

⁴<https://bit.ly/2S2PMxY>

⁵<https://bit.ly/2tnO4gG>

and the measure of a candidate’s success is the total weight of districts that support her. Both of these rules are widely used in practice. For example, Plurality over Voters is commonly used in gubernatorial elections in the US, while Plurality over Districts is used in the US Presidential elections.

We provide a detailed analysis of the computational complexity of the algorithmic problems faced by the attacker and the defender. Our main results are summarized in Table 1. Briefly, assuming that the vote counts and the weights of the districts are specified in binary, most of the problems we consider are computationally hard; however, the defender’s problem appears to be easier than that of the attacker, and we also get some tractability results for the former. Towards the end of the paper, we consider a variant of our model where the attacker is limited to only transferring votes to his preferred candidate; we show that, while this assumption reduces the attacker’s ability to achieve his goals, it lowers the complexity of some of the problems we consider.

Related Work. There is a very substantial literature on voting manipulation and bribery; we point the readers to the excellent surveys of Conitzer and Walsh [2016] and Faliszewski and Rothe [2016]. In much of this literature it is assumed that the malicious party can change some of the votes subject to various constraints, and the challenge is to determine whether the attacker’s task is computationally feasible; there is no defender that can counteract the attacker’s actions.

While there is a number of papers that apply game-theoretic analysis to the problem of voting manipulation, they typically consider interactions between several manipulators, with possibly conflicting goals (e.g., see the recent book by Meir [2018]), rather than a manipulator and a socially-minded actor. An important exception, which is similar in spirit to our paper, is the recent work of Yin *et al.* [2018], who investigate a pre-emptive approach to protecting elections. In their model the defender allocates resources to guard some of the electoral districts, so that the votes there cannot be corrupted; notably, in this model the defender has to commit to its strategy first, and the attacker can observe the defender’s actions before deciding on its response. The leader-follower (defender-attacker) structure of this model is in the spirit of a series of successful applications of Stackelberg games to security resource allocation problems [Tambe, 2011]. Li *et al.* [2017] analyze a variant of the model of Yin et al. where the goal is to minimize resource consumption, and Chen *et al.* [2018] study a similar scenario, in which manipulation is achieved through bribing the voters. The key difference between our work and the above papers is the action order of the players: in all prior work on election protection that we are aware of the defender makes the first move.

2 The Model

We consider elections over a *candidate set* C , $|C| = m$. There are n *voters* who are partitioned into k pairwise disjoint *districts* D_1, \dots, D_k , $k \leq n$; for each $i \in [k]$, let $n_i = |D_i|$. For each $i \in [k]$, district D_i has a *weight* w_i , which is a positive integer; we say that an election is *unweighted* if $w_i = 1$ for all $i \in [k]$. Each voter votes for a single candidate in C . For each $i \in [k]$ and each $a \in C$ let v_{ia} denote the number of votes that candidate a gets from voters in D_i ; we refer to the list $\mathbf{v} = (v_{ia})_{i \in [k], a \in C}$ as the *vote profile*.

Let \succ be a linear order over C ; $a \succ b$ indicates that a is favored over b . We consider the following two voting rules, which take the vote profile \mathbf{v} as their input.

- *Plurality over Voters* (PV). We say that a candidate a *beats* a candidate b under PV if $\sum_{i \in [k]} v_{ia} > \sum_{i \in [k]} v_{ib}$ or $\sum_{i \in [k]} v_{ia} = \sum_{i \in [k]} v_{ib}$ and $a \succ b$; the winner is the candidate that beats all other candidates. Note that district weights w_i are not relevant for this rule.
- *Plurality over Districts* (PD). For each $i \in [k]$ the winner a_i in D_i is chosen from the set $\arg \max_{a \in C} v_{ia}$, with ties broken according to \succ . Then, for each $i \in [k]$, $a \in C$, we set $w_{ia} = w_i$

if $a = a_i$ and $w_{ia} = 0$ otherwise. We say that a candidate a *beats* a candidate b under PD if $\sum_{i \in [k]} w_{ia} > \sum_{i \in [k]} w_{ib}$ or $\sum_{i \in [k]} w_{ia} = \sum_{i \in [k]} w_{ib}$ and $a \succ b$; the winner is the candidate that beats all other candidates.

For PV and PD, we define the *social welfare* of a candidate $a \in C$ as the total number of votes that a gets and the total weight that a gets, respectively:

$$\text{SW}^{\text{PV}}(a) = \sum_{i \in [k]} v_{ia}, \quad \text{SW}^{\text{PD}}(a) = \sum_{i \in [k]} w_{ia}.$$

Hence, the winner under each voting rule is a candidate with the maximum social welfare.

We consider scenarios where an election may be manipulated by an *attacker*, who wants to change the election result a^* in favor of his preferred candidate $p \in C$. The attacker has a budget $B_A \in [k]$, which means that he can manipulate at most B_A districts. For each $i \in [k]$, we are given an integer γ_i , $0 \leq \gamma_i \leq n_i$, which indicates how many votes the attacker can change in district i if he chooses to manipulate it. Formally, a *manipulation* is described by a set $M \subseteq [k]$, $|M| \leq B_A$, and a vote profile $\tilde{\mathbf{v}} = (\tilde{v}_{ia})_{i \in [k], a \in C}$ such that $\tilde{v}_{ia} = v_{ia}$ for all $i \notin M$, $a \in C$, and for all $i \in [k]$ it holds that $\sum_{a \in C} \tilde{v}_{ia} = n_i$ and $\sum_{a \in C} \max\{0, \tilde{v}_{ia} - v_{ia}\} \leq \gamma_i$.

After the attack, a *defender* with budget $B_D \in \{0\} \cup [k]$ can demand a recount in at most B_D districts. Formally, a defender's strategy is a set $R \subseteq M$ with $|R| \leq B_D$; after the defender acts, the vote counts in all districts in R are restored to their original values, i.e., the resulting vote profile $\mathbf{u} = (u_{ia})_{i \in [k], a \in C}$ satisfies $u_{ia} = v_{ia}$ for each $i \in R$, $a \in C$ and $u_{ia} = \tilde{v}_{ia}$ for each $i \in [k] \setminus R$, $a \in C$. Then the underlying voting rule $\mathcal{R} \in \{\text{PV}, \text{PD}\}$ is applied to \mathbf{u} with ties broken according to \succ ; let a' denote the candidate selected in this manner. The defender chooses her strategy R so as to maximize $\text{SW}^{\mathcal{R}}(a')$, breaking ties using \succ .

We say that the attacker *wins* if he has a strategy $(M, \tilde{\mathbf{v}})$ such that, once the defender responds optimally, candidate p is the winner in the resulting vote profile \mathbf{u} ; otherwise we say that the attacker *loses*. We note that if $B_D \geq B_A$, the defender can always ensure that $a' = a^*$, i.e., the winner at \mathbf{u} is the winner at the original vote profile \mathbf{v} , so in what follows we assume that the attacker's strategy satisfies $|M| > B_D$.

Example 2.1. Consider an election with five districts D_1, \dots, D_5 over a candidate set $C = \{a, b, p\}$, where p is the attacker's preferred candidate; suppose that ties are broken according to the priority order $p \succ a \succ b$. In each of D_1 and D_2 there are 7 voters who vote for a , and in each of D_3, D_4 and D_5 there are 3 voters who vote for b . Suppose that $\gamma_i = n_i$ and $w_i = (n_i)^2$ for each $i \in [5]$, and $B_A = 2$, $B_D = 1$.

If the voting rule is PV, then the attacker does not have a winning strategy. Indeed, consider an attacker's strategy $(M, \tilde{\mathbf{v}})$. If $M \neq \{1, 2\}$, the defender can set $R = M \cap \{1, 2\}$; in the recounted vote profile a gets at least 14 votes, so it is the election winner. If $M = \{1, 2\}$, the defender can set $R = \{1\}$: in the recounted vote profile p gets at most 7 votes, while b gets at least 9 votes, so the winner is a or b (a can win if, e.g., the attacker chooses to transfer exactly 4 votes from a to p in D_2 , in which case a gets 10 votes after the recount). Note that even if the winner in \mathbf{u} is b rather than a , the defender still prefers recounting D_1 to no recounting: even though she cannot restore the correct result, she prefers b to p , since $\text{SW}^{\text{PV}}(b) = 9 > 0 = \text{SW}^{\text{PV}}(p)$.

If the voting rule is PD, then the attacker can win by choosing $M = \{1, 2\}$ and transferring a majority of votes from a to p in both districts. Indeed, even if the defender demands a recount in one of these districts, p still wins the remaining district, leading to a vote weight of 49 in the recounted profile. Since a 's vote weight is 49 and b 's vote weight is 27, p wins by the tie-breaking rule. \square

We assume that both the defender and the attacker have full information about the game. Both parties know the true vote profile \mathbf{v} , the parameters w_i and γ_i for each district $i \in [k]$ and each others' budgets. Moreover, the defender observes the strategy $(M, \tilde{\mathbf{v}})$ of the attacker.

We can now define the following decision problems for each $\mathcal{R} \in \{\text{PV}, \text{PD}\}$:

- \mathcal{R} -MAN: Given a vote profile \mathbf{v} , the attacker's preferred candidate p , budgets B_A and B_D , and district parameters $(w_i, \gamma_i)_{i \in [k]}$, does the attacker have a winning strategy?
- \mathcal{R} -REC: Given a vote profile \mathbf{v} , a distorted vote profile $\tilde{\mathbf{v}}$ with winner b , a candidate $a \neq b$, a budget B_D , and district weights $(w_i)_{i \in [k]}$, can the defender recount the votes in at most B_D districts so that a gets elected?

We will also consider an optimization version of \mathcal{R} -REC, where c is not part of the input and the goal is to maximize the social welfare of the eventual winner.

Unless specified otherwise, we assume that the vote counts v_{ia} and the district weights w_i are given in binary; we explicitly indicate which of our hardness results still hold if these numbers are given in unary. All problems considered in this paper admit straightforward greedy algorithms for $m = 2$, so in what follows we focus on the case $m \geq 3$. When the voting rule $\mathcal{R} \in \{\text{PV}, \text{PD}\}$ is clear from context, we write $\text{SW}(a)$ instead of $\text{SW}^{\mathcal{R}}(a)$.

Next, we give formal definitions of the decision problems that are used throughout the paper to show hardness of \mathcal{R} -REC and \mathcal{R} -MAN for $\mathcal{R} \in \{\text{PV}, \text{PD}\}$, under various constraints.

Definition 2.2 (SUBSET SUM). An instance of SUBSET SUM is given by a multiset X of integers. It is a yes-instance if there exists a non-empty subset $X' \subseteq X$ such that $\sum_{x \in X'} x = 0$, and a no-instance otherwise.

Definition 2.3 (EXACT COVER BY 3-SETS (X3C)). An instance of X3C is given by a set E of size 3ℓ and a collection \mathcal{S} of 3-element subsets of E . It is a yes-instance if there exists a sub-collection $\mathcal{Q} \subseteq \mathcal{S}$ of size ℓ such that $\cup_{S \in \mathcal{Q}} S = E$, and a no-instance otherwise.

Definition 2.4 (INDEPENDENT SET). An instance of INDEPENDENT SET is a graph $G = (V, E)$ and an integer ℓ . It is a yes-instance if there exists a subset $V' \subseteq V$ of size ℓ that forms an independent set, i.e., $\{a, b\} \notin E$ for all $a, b \in V'$, and a no-instance otherwise.

Definition 2.5 (PARTITION). An instance of PARTITION is given by a multiset X of positive integers. It is a yes-instance if there exists a subset $X' \subseteq X$ such that $\sum_{x \in X'} x = \frac{1}{2} \sum_{x \in X} x$, and a no-instance otherwise.

All of these problems are NP-complete [Garey and Johnson, 1979]. However, SUBSET SUM and PARTITION are NP-hard only when the input is given in binary; for unary input, these problems can be solved in time polynomial in the size of the input.

3 Plurality over Voters

In this section we focus on Plurality over Voters. We first take the perspective of the defender, and then the perspective of the attacker.

Unfortunately, the defender's problem turns out to be computationally hard, even if there are only three candidates or if the input vote counts are given in unary.

Theorem 3.1. PV-REC is NP-complete even when

- (i) $m = 3$, or
- (ii) the input vote profile is given in unary.

Proof. This problem is clearly in NP. We give separate hardness proofs for the case $m = 3$ (part (i)) and for the case where the input is given in unary (part (ii)).

Part (i). To prove that PV-REC is NP-hard for $m = 3$, we provide a reduction from SUBSET SUM; see Definition 2.2.

Given an instance X of SUBSET SUM with $|X| = \ell$, we construct an instance of PV-REC as follows. Without loss of generality, we assume that $x \neq 0$ for every $x \in X$ and $\sum_{x \in X} x > 0$, and let $X^+ = \{x \in X : x > 0\}$, $X^- = \{x \in X : x < 0\}$, $y = \sum_{x \in X} 2|x|$. We set $C = \{a, b, p\}$, where p is the attacker's preferred candidate. In what follows, we describe each district D_i by a tuple (v_{ia}, v_{ib}, v_{ip}) . There are $n = 12y\ell + 1$ voters distributed over $\ell + 3$ districts, which are further partitioned into two sets I_1 and I_2 as follows:

- For each $x \in X^+$ there is a district in I_1 with votes $(0, 2x, 0)$, which are distorted to $(0, 0, 2x)$, and for each $x \in X^-$ there is a district in I_1 with votes $(0, 0, -2x)$, which are distorted to $(0, -2x, 0)$. Note that $|I_1| = \ell$.
- I_2 contains three districts with votes $(y+1, 0, 0)$, $(0, y - \sum_{x \in X^+} 2x, 0)$, and $(0, 0, y + \sum_{x \in X^-} 2x)$, respectively. The votes in these districts are not distorted.

Finally, $B_D = \ell - 1$.

Before the manipulation, a gets $y + 1$ votes and b and p get y votes each. After the manipulation, a gets $y + 1$ votes, b gets $y - \sum_{x \in X^+} 2x$ and p gets $y + \sum_{x \in X^-} 2x$ votes; thus, by our assumption that $\sum_{x \in X} x > 0$, candidate p is the winner in the manipulated profile. The goal is to restore the true winner a .

Now, assume that there exists a subset $X' \subseteq X$ with $|X'| \geq 1$ such that $\sum_{x \in X'} x = 0$. Then, by recounting the $\ell - |X'|$ districts of I_1 that correspond to the integers in $X \setminus X'$, the defender can ensure that both b and p get y votes. Since a always gets $y + 1$ votes from the non-manipulated districts, she is successfully restored as the winner.

Conversely, assume that there is no non-empty subset $X' \subseteq X$ such that $\sum_{x \in X'} x = 0$. Then, since the votes of b and p always add up to exactly $2y$, and each of them gets an even number of votes from each district, one of them must get at least $y + 2$ votes. Therefore, a cannot be restored as the winner.

Part (ii). We give a reduction from EXACT COVER BY 3-SETS (X3C); see Definition 2.3. Given an instance of X3C, we construct the following PV-REC instance. Without loss of generality, we assume that $\cup_{S \in \mathcal{S}} S = E$, and let $s = |\mathcal{S}|$.

- Let $C = \{j_e : e \in E\} \cup \{a, b\}$, $|C| = 3\ell + 2$.
- For each subset $S \in \mathcal{S}$, there is a district D_S , where a gets 2 votes, b gets 6 votes, for each $e \notin S$ candidate j_e gets 2 votes, and for each $e \in S$ candidate j_e gets 0 votes. The attacker distorts the votes in D_S by transferring two votes from b to each candidate j_e with $e \in S$, so that in the distorted profile b gets 0 votes in D_S and every other candidate gets 2 votes in D_S .
- There is a district D_0 where a receives $6\ell s$ votes, b receives 0 votes and for every $e \in E$ candidate j_e receives $6\ell s + 1$ votes; the votes in this district are not distorted.
- The budget of the defender is $B_D = \ell$.

Candidate a is the true winner with $2s + 6\ell s$ votes, compared to the $6s$ votes of b and the $2|\{S \in \mathcal{S} : e \notin S\}| + 6\ell s + 1 \leq 2s + 6\ell s - 1$ votes of j_e for every $e \in E$. In the distorted profile $\tilde{\mathbf{v}}$ candidate a gets $2s + 6\ell s$ votes, candidate b gets 0 votes, and each candidate in $C \setminus \{a, b\}$ gets $2s + 6\ell s + 1$ votes.

Recounting a district D_S reduces by 2 the votes of each candidate j_e such that $e \in S$, leading to a getting more votes than these candidates; b cannot get more than $6s$ votes no matter what the defender does. Therefore, a can be restored as the winner by recounting ℓ districts if and only if E can be covered by ℓ sets from \mathcal{S} . \square

If the number of candidates is bounded by a constant and the input is given in unary, an optimal set of districts to recount can be identified in time polynomial in the input size by means of dynamic programming.

Theorem 3.2. PV-REC can be solved in time $O(k \cdot B_{\mathcal{D}} \cdot (n + 1)^m)$.

Proof. Consider an instance of PV-REC with a candidate set C , $|C| = m$, and n voters that are distributed over k districts. For each $i \in [k]$, let $\mathbf{v}_i = (v_{ia})_{a \in C}$ and $\tilde{\mathbf{v}}_i = (\tilde{v}_{ia})_{a \in C}$ denote, respectively, the true and distorted votes in district i . Let $B_{\mathcal{D}}$ be the budget of the defender.

We present a dynamic programming algorithm that given a candidate $c \in C$, decides whether c can be made the election winner by recounting at most $B_{\mathcal{D}}$ districts. Our algorithm fills out a table T containing entries of the form $T(i, \ell, \mathbf{u})$, for each $i \in \{0, 1, \dots, k\}$, $\ell \in \{0, 1, \dots, B_{\mathcal{D}}\}$, and $\mathbf{u} = (u_a)_{a \in C} \in \{0, \dots, n\}^m$; thus, $|T| = O(k \cdot B_{\mathcal{D}} \cdot (n + 1)^m)$. We define $T(i, \ell, \mathbf{u}) = \text{true}$ if we can recount at most ℓ of the first i districts so that the vote count of candidate a equals u_a for each $a \in C$; otherwise we define $T(i, \ell, \mathbf{u}) = \text{false}$. There exists a recounting strategy that restores c if and only if there exists a \mathbf{u} such that $T(k, B_{\mathcal{D}}, \mathbf{u}) = \text{true}$, $u_c \geq u_a$ for all $a \in C$, and for all $a \in C \setminus \{c\}$ such that $u_c = u_a$ the tie-breaking rule favors c over a .

For each $a \in C$, let $\hat{u}_a = \sum_{i \in [k]} \tilde{v}_{ia}$ be the number of votes that candidate a gets after manipulation, and let $\tilde{\mathbf{u}} = (\tilde{u}_a)_{a \in C}$. We fill out T according to the following rule:

$$T(i, \ell, \mathbf{u}) = \begin{cases} \text{true}, & \text{if } \mathbf{u} = \tilde{\mathbf{u}} \\ \text{false}, & \text{if } i = 0 \text{ or } \ell = 0, \text{ and } \mathbf{u} \neq \tilde{\mathbf{u}} \\ T(i - 1, \ell, \mathbf{u}) \vee (\mathbf{u} - \mathbf{v}_i + \tilde{\mathbf{v}}_i \in \{0, \dots, n\}^m \text{ and} \\ & T(i - 1, \ell - 1, \mathbf{u} - \mathbf{v}_i + \tilde{\mathbf{v}}_i)), & \text{otherwise.} \end{cases}$$

This completes the proof. □

We obtain similar hardness results for the attacker's problem. However, it is not clear if PV-MAN is in NP. Indeed, it may belong to a higher level of the polynomial hierarchy: it is not hard to see that PV-MAN is in Σ_2^P , and it is plausible that this problem is hard for this complexity class.

Theorem 3.3. PV-MAN is NP-hard even when $B_{\mathcal{D}} = 0$, $\gamma_i = n_i$ for all $i \in [k]$ and

- (i) $m = 3$, or
- (ii) the input vote profile is given in unary.

Proof. We prove the two claims separately.

Part (i). To prove that PV-MAN is NP-hard for $m = 3$, we provide a reduction from SUBSET SUM; see Definition 2.2.

Given an instance X of SUBSET SUM with $|X| = \ell$, we construct an instance of PV-MAN as follows. We can assume without loss of generality that $\ell \geq 2$ and $x \neq 0$ for every $x \in X$, and let $y = \max_{x \in X} 2|x|$; by our assumptions, $y \geq 2$. We set $C = \{a, b, p\}$, where p is the attacker's preferred candidate. In what follows, we describe each district D_i by a tuple (v_{ia}, v_{ib}, v_{ip}) . There are $n = 12y\ell + 1$ voters distributed over $k = 4\ell + 2$ districts, which are further partitioned into four sets I_1, I_2, I_3, I_4 as follows:

- For each $x \in X$ there is a district in I_1 with votes $(2y + 4x, 2y - 4x, 0)$. Thus, $|I_1| = \ell$.
- Set I_2 consists of $\ell - 1$ districts with votes $(2y, 2y, 0)$ in each district.
- For each $x \in X$ there are two districts in I_3 with votes $(y - 2x, y + 2x, 0)$. Thus, $|I_3| = 2\ell$.

- Set I_4 consists of three districts with votes $(y, y, 0)$, $(y, y, 0)$, and $(0, 0, 1)$.

We set $B_{\mathcal{A}} = \ell$, $B_{\mathcal{D}} = 0$ and $\gamma_i = n_i$ for each $i \in [k]$.

We have $\text{SW}(a) = \text{SW}(b) = 6y\ell$ and $\text{SW}(p) = 1$. Hence, the true winner is a or b , depending on the tie-breaking rule. We claim that the attacker can make p the winner if and only if there exists a non-empty subset $X' \subseteq X$ such that $\sum_{x \in X'} x = 0$.

To see this, assume first that there exists a subset $X' \subseteq X$ such that $|X'| \geq 1$ and $\sum_{x \in X'} x = 0$. Then the attacker can distort the votes in the $|X'|$ districts of I_1 corresponding to the elements of X' , and in arbitrary $\ell - |X'|$ districts of I_2 , by transferring all votes to p in each of these districts. In the resulting election, p gets $4y\ell + 1$ votes, while a and b get $4y\ell$ votes each, so p becomes the winner.

Conversely, suppose that the attacker has a successful manipulation $(M, \tilde{\mathbf{v}})$ with $|M| \leq \ell$. For each $c \in C$, let s_c denote the number of votes that c receives in $\tilde{\mathbf{v}}$. For p to be the winner in $\tilde{\mathbf{v}}$, it must hold that $s_p \geq n/3 = (12y\ell + 1)/3$; since s_p is an integer and $\text{SW}(p) = 1$, this means that the manipulation transfers at least $4y\ell$ votes to p . On the other hand, in every district there are at most $4y$ voters who vote for a or b , so p can gain at most $4y\ell$ votes from the manipulation. It follows that $s_p = 4y\ell + 1$, $s_a + s_b = 8y\ell$. If these $8y\ell$ votes are not split evenly between a and b , at least one of these candidates would get strictly more than $4y\ell$ points; since each district allocates an even number of votes to both a and b , this further means that one of them would get at least $4y\ell + 2$ votes, a contradiction with p being the winner at $\tilde{\mathbf{v}}$. Thus, it must be the case that $s_a = s_b = 4y\ell$.

Further, $s_p = 4y\ell + 1$, $|M| = \ell$ implies that $M \subseteq I_1 \cup I_2$ and $M \cap I_1 \neq \emptyset$. Moreover, we have $\tilde{v}_{ia} = \tilde{v}_{ib} = 0$ for every district $i \in M$. Hence,

$$s_a = 4y\ell - 4 \sum_{i \in M \cap I_1} x_i, \quad s_b = 4y\ell + 4 \sum_{i \in M \cap I_1} x_i,$$

where x_i is the integer in X that corresponds to district D_i . Thus, $\sum_{i \in M \cap I_1} x_i = 0$, and hence $X' = \{x_i \in X : i \in M \cap I_1\}$ is a witness that X is a yes-instance of SUBSET SUM.

Part (ii). To prove that PV-MAN is NP-hard when the input is given in unary, we provide a reduction from X3C; see Definition 2.3.

Given an instance $\langle E, \mathcal{S} \rangle$ of X3C with $|E| = 3\ell$, $|\mathcal{S}| = s$, we construct an instance of PV-MAN as follows. We set $C = \{j_e : e \in E\} \cup \{p\}$, where p is the attacker's preferred candidate. The districts are partitioned into three sets I_1, I_2, I_3 :

- For each subset $S \in \mathcal{S}$ the set I_1 contains a district D_S . In this district each candidate j_e such that $e \in S$ gets 3ℓ votes, and all other candidates get no votes. Thus, $|D_S| = 9\ell$.
- For each element $e \in E$, the set I_2 contains $3\ell s + 9\ell^2 - 3\ell \cdot |\{S \in \mathcal{S} : e \in S\}|$ districts; each of these districts consists of a single voter who votes for j_e .
- The set I_3 contains a single district D^* that consists of $3\ell s - 2\ell$ voters who vote for p .

We set $B_{\mathcal{A}} = \ell$, $B_{\mathcal{D}} = 0$ and $\gamma_i = n_i$ for all $i \in [k]$.

We have $\text{SW}(j_e) = 3\ell s + 9\ell^2$ for all $e \in E$ and $\text{SW}(p) = 3\ell s - 2\ell$. Hence, the true winner is the candidate in $C \setminus \{p\}$ who is favored by the tie-breaking rule. We show that the attacker is able to make p the winner if and only if E admits an exact cover by sets from \mathcal{S} .

Suppose that $\mathcal{Q} \subseteq \mathcal{S}$ is an exact cover for E ; note that $|\mathcal{Q}| = \ell$. The attacker can manipulate the ℓ districts in I_1 that correspond to sets in \mathcal{Q} by reassigning all the 9ℓ votes in each of them to p . In the resulting election, p gets $3\ell s + 9\ell^2 - 2\ell$ votes, while every other candidate j_e gets $3\ell s + 9\ell^2 - 3\ell$ votes, as every e is covered by exactly one set in \mathcal{Q} .

Conversely, suppose the attacker has a successful manipulation $(M, \tilde{\mathbf{v}})$ with $|M| \leq \ell$. For each $c \in C$, let s_c denote the number of votes that c receives in $\tilde{\mathbf{v}}$. As p can gain at most 9ℓ votes for each

district in M , we have $s_p \leq 3\ell s + 9\ell^2 - 2\ell$. Let $\mathcal{Q} = \{S \in \mathcal{S} : D_S \text{ is manipulated}\}$; note that $|\mathcal{Q}| \leq \ell$. We claim that \mathcal{Q} is a cover for E . Indeed, if for some $e \in E$ no district in $\{D_S : e \in S\}$ is manipulated, the manipulation lowers the score of j_e by at most ℓ , so $s_{j_e} \geq 3\ell s + 9\ell^2 - \ell > s_p$, a contradiction. \square

In the hardness reductions in the proof of Theorem 3.3 the defender’s budget is 0. This indicates that the attacker’s problem remains NP-hard even if the defender is known to use a heuristic (e.g., a greedy algorithm) to compute her response.

We remark that PV-REC and PV-MAN with $B_D = 0$ are very similar in spirit to combinatorial (shift) bribery [Bredereck *et al.*, 2016]. In both models, a budget-constrained agent needs to select a set of vote-changing actions, with each action affecting a group of voters. However, there are a few technical differences between the models. For instance, in our model different actions are associated with non-overlapping groups of voters, which is not the case in combinatorial shift bribery. On the other hand, in shift bribery under the Plurality rule votes can only be transferred to/from the manipulator’s preferred candidate p , while our model does not impose this constraint (see, however, Section 5). Consequently, it appears that the technical results in our paper cannot be derived from known results for combinatorial shift bribery.

4 Plurality over Districts

In this section we study Plurality over Districts. For the defender’s problem, we can replicate the results we obtain for Plurality over Voters, by using similar techniques.

Theorem 4.1. PD-REC is NP-complete even when

- (i) $m = 3$, or
- (ii) the input vote profile and district weights are given in unary.

Proof. This problem is clearly in NP. We give separate hardness proofs for the case $m = 3$ (part (i)) and for the case where the input is given in unary (part (ii)).

Part (i). We use the same reduction as in the proof of the first part of Theorem 3.1. An important feature of this reduction is that all voters in each district vote for the same candidate. Thus, if we set the weight of each district to be equal to the number of voters therein, the proof goes through without change.

Part (ii). We provide a reduction from INDEPENDENT SET; see Definition 2.4. Given an instance $\langle G, \ell \rangle$ of INDEPENDENT SET, where $G = (V, E)$, we construct an instance of PD-REC as follows. Let $\nu = |V|$, $\mu = |E|$; we can assume without loss of generality that $\mu \geq 1$. We set $C = \{j_u : u \in V\} \cup \{j_e : e \in E\} \cup \{a, p\}$, where p is the attacker’s preferred candidate; thus, $|C| = \nu + \mu + 2$. We create the following districts. For our argument, the district sizes and the values of γ_i do not matter; for concreteness, we assume that each district consists of a single voter, whose vote can be changed by the manipulator.

- For each edge $e = \{x, y\} \in E$, there are two districts $D_{e,x}$ and $D_{e,y}$ with weight 2 each. In each such district $D_{e,u}$ the winner before manipulation is j_e , and the winner after manipulation is j_u .
- For each node $u \in V$, there is a district D_u with weight 2μ ; in this district the winner before manipulation is j_u , and the winner after manipulation is p .

- There is a set I of $2(\nu + \mu) + 1$ districts with weight $\frac{2}{2(\nu + \mu) + 1}$ each⁶; in each such district the winner before manipulation is a , and the winner after manipulation is p .
- There is a district of weight $2(\nu - \ell)\mu + 3$ with winner a ; this district is not manipulated.
- For each $e \in E$, there is a district of weight $2(\nu - \ell)\mu$ with winner j_e ; this district is not manipulated.
- For each $u \in V$, there is a district of weight $2(\nu - \ell)\mu - 2\mu + 2$ with winner j_u ; this district is not manipulated.

The budget of the defender is $B_D = \nu + \mu$. The candidates' weights before and after manipulation are given in the following table:

	true weight	distorted weight
a	$2(\nu - \ell)\mu + 5$	$2(\nu - \ell)\mu + 3$
p	0	$2\nu\mu + 2$
$j_e, e \in E$	$2(\nu - \ell)\mu + 4$	$2(\nu - \ell)\mu$
$j_u, u \in V$	$2(\nu - \ell)\mu + 2$	$\leq 2(\nu - \ell)\mu + 2$

Hence, the true winner is candidate a and the winner after manipulation is p .

If $V' \subseteq V$ is an independent set of size ℓ in G , the defender can proceed as follows. For each $u \in V'$, she demands a recount in D_u and in each district $D_{e,u}$ such that e is incident to u . Since V' forms an independent set, this requires recounting at most $\nu + \mu$ districts. Moreover, after the recount the weight of p is $2(\nu - \ell)\mu + 2$, the weight of a is $2(\nu - \ell)\mu + 3$, the weight of each candidate j_u such that $u \in V'$ is $2(\nu - \ell)\mu + 2$, the weight of each candidate j_u such that $u \in V \setminus V'$ is at most $2(\nu - \ell)\mu + 2$, and the weight of each candidate j_e such that $e \in E$ is at most $2(\nu - \ell)\mu + 2$. Thus, this recounting strategy successfully restores a as the election winner.

Conversely, suppose that the defender has a recounting strategy R that results in making a the election winner. Since $|R| \leq B_D$, at most $\nu + \mu$ districts in I can be recounted, so a 's weight after the recount is at most $2(\nu - \ell) + 3 + \frac{2(\nu + \mu)}{2(\nu + \mu) + 1} < 2(\nu - \ell) + 4$. Now, if R contains at most $\ell - 1$ districts in $\{D_u : u \in V\}$, then p 's weight after the recount is at least $2(\nu - \ell + 1)\mu + 2 \geq 2(\nu - \ell) + 4$, a contradiction with a becoming the winner after the recount. Hence, R contains at least ℓ districts in $\{D_u : u \in V\}$; let V' be the subset of nodes corresponding to these districts. We claim that V' forms an independent set in G .

Indeed, consider a node $u \in V'$. If the defender does not recount some district $D_{e,u}$ such that u is incident to e then after the recount the weight of j_u is at least $2(\nu - \ell)\mu + 4$, a contradiction with a becoming the winner after the recount. Thus $D_{e,u}$ is necessarily recounted. Now, suppose that $e = \{x, y\} \in E$ for some $x, y \in V$. We have just argued that both $D_{e,x}$ and $D_{e,y}$ have to be recounted. But this means that the score of j_e is at least $2(\nu - \ell)\mu + 4$ after the recount, a contradiction again. Thus, V' is an independent set. \square

Theorem 4.2. PD-REC can be solved in time $O(k \cdot B_D \cdot (n + 1)^m)$.

Proof. The algorithm is a simple adaptation of the dynamic program presented in the proof of Theorem 3.2. \square

We also obtain a positive result that does not have an analogue in the PV setting; if all districts have the same weight, the recounting problem can be solved efficiently.

⁶For convenience, we use fractional weights. We can turn all weight into integers, by multiplying them by $2(\nu + \mu) + 1$.

Theorem 4.3. PD-REC can be solved in polynomial time if $w_i = 1$ for all $i \in [k]$.

Proof. We reduce our problem to nonuniform bribery [Faliszewski, 2008]. An instance of nonuniform bribery under the Plurality rule is given by a set of voters and a set of candidates; for each voter i and each candidate c there is a price π_{ic} for making voter i vote for c , and the briber's goal is to make her preferred candidate the Plurality winner⁷ while staying within a budget B . This problem is known to be in P [Faliszewski, 2008]. To reduce PD-REC to nonuniform bribery, we map each district D_i to a single voter i ; if the true winner in D_i is x , and in the distorted profile the winner in D_i is y , we set $\pi_{iy} = 0$, $\pi_{iz} = +\infty$ for $z \in C \setminus \{x, y\}$, and if $x \neq y$ (i.e., if the attacker has changed the outcome in D_i), we set $\pi_{ix} = 1$. Then for any candidate $c \in C$ it holds that in PD-REC the defender can make c win by recounting at most B_D districts if and only if in our instance of nonuniform bribery the briber can make c win by spending at most B_D . \square

We now consider the attacker's problem. It turns out that for the PD rule we can obtain a stronger hardness result than for PV: we will now argue that when weights and vote counts are given in binary, PD-MAN is Σ_2^P -complete even for $m = 3$. Our reduction uses a variant of the SUBSET SUM problem, which we term SUB-SUBSET SUM (SSS); this problem may be of independent interest.

Definition 4.4 (SUB-SUBSET SUM). An instance of SUB-SUBSET SUM is a set $X \subseteq \mathbb{Z}$ and a positive integer ℓ . It is a yes-instance if there is a subset $X' \subseteq X$ with $|X'| = \ell$ such that $\sum_{x \in X''} x \neq 0$ for every non-empty subset $X'' \subseteq X'$, and a no-instance otherwise.

Our proof proceeds by establishing that SSS is Σ_2^P -complete (Lemma 4.5; the proof can be found in the appendix), and then reducing this problem to PD-MAN.

Lemma 4.5. SSS is Σ_2^P -complete.

Theorem 4.6. PD-MAN is Σ_2^P -complete, even when $m = 3$.

Proof. Clearly, PD-MAN is in Σ_2^P . To prove hardness, we reduce from SSS. Given an instance $\langle X, \ell \rangle$ of SSS, we construct an instance of PD-MAN with three candidates $\{a, b, p\}$. Let $X^+ = \{x \in X : x > 0\}$ and $X^- = X \setminus X^+$. Set $y = \sum_{x \in X} 3|x|$. In what follows we describe the votes in each district D_i as a list (v_{ia}, v_{ib}, v_{ip}) . The districts are partitioned into three sets I_1 , I_2 and I_3 :

- I_1 has a district with votes $(0, 3x, 0)$ for each $x \in X^+$, and a district with votes $(0, 0, -3x)$ for each $x \in X^-$.
- I_2 consists of a single district with votes $(0, y + 3, 0)$.
- I_3 consists of three districts with votes $(2y + 5, 0, 0)$, $(0, y - \sum_{x \in X^+} 3x, 0)$, and $(0, 0, 2y + 4 + \sum_{x \in X^-} 3x)$.

For every district D_i we set $w_i = n_i$. The attacker is allowed to change all votes in each district in I_1 and I_2 , but none in I_3 . Finally, let $B_A = \ell + 1$ and $B_D = \ell$. The true winner in this profile is candidate a with weight $2y + 5$, compared to the weight $2y + 3$ of b and $2y + 4$ of p .

Given a set of integers $Y \subseteq X$, let $I_1(Y)$ be the corresponding set of districts in I_1 . Assume that there is a subset $X' \subseteq X$ with $|X'| = \ell$ such that no $X'' \subseteq X'$ has sum equal to 0. The attacker can then exchange the weights of b and p in the districts in $I_1(X')$ and the district in I_2 . This way, p becomes the winner with weight $3y + 7 + \sum_{x \in X'} 3x \geq 2y + 7$, compared to the weight $2y + 5$ of a and the weight $y - \sum_{x \in X'} 3x \leq 2y + 4$ of b .

Since $\text{SW}(p) > \text{SW}(b)$, to defeat the attacker, the defender needs to restore a as the winner. To this end, she must recount the district in I_2 , as otherwise p 's weight will remain at least $2y + 7$. Hence

⁷Faliszewski [2008] assumes that ties are broken in favor of the briber, but his results extend to lexicographic tie-breaking.

she can recount at most $\ell - 1$ manipulated districts in I_1 . Let the set of non-recounted districts in I_1 be $I_1(X'')$ for some $X'' \subseteq X'$; note that $X'' \neq \emptyset$, so by assumption, $\sum_{x \in X''} x \neq 0$. Then, the weight of b is $2y + 3 - \sum_{x \in X''} 3x$ and the weight of p is $2y + 4 + \sum_{x \in X''} 3x$. At least one of these numbers is greater than or equal to $2y + 6$; thus, a cannot be restored as the winner.

Conversely, suppose that for every subset $X' \subseteq X$ of size ℓ there exists a non-empty $X'' \subseteq X'$ such that $\sum_{x \in X''} x = 0$. Then, the attacker cannot win. Indeed, let M be the set of manipulated districts. If a district is changed in favor of a , the defender can recount all other districts in M . On the other hand, if all districts in M are won by b or p , the defender can identify a non-empty subset of $M \cap I_1$ such that the corresponding integers sum up to 0, and request a recount of all other districts in M . Such a recount recovers the correct weights of b and p , and a is restored as the winner. \square

We conjecture that PD-MAN remains Σ_2^P -complete when the input is given in unary; however, for this setting we are only able to prove that this problem is NP-hard.

Theorem 4.7. PD-MAN is NP-hard, even when $B_D = 0$ and the input vote profile and district weights are given in unary.

Proof. To show that PD-MAN is NP-hard even when the input votes and district weights are given in unary, we provide a reduction from INDEPENDENT SET; see Definition 2.4.

Given an instance $\langle G, \ell \rangle$ of INDEPENDENT SET with $G = (V, E)$, we construct the following instance of PD-MAN. Let $\nu = |V|$, $\mu = |E|$. We set $C = \{j_u : u \in V\} \cup \{j_e : e \in E\} \cup \{a, p\}$, where p is the attacker's preferred candidate; thus, $|C| = \nu + \mu + 2$. Then, we create the following districts; the weight of each district is equal to the number of voters therein.

- For every edge $e = \{x, y\} \in E$, we create two districts $D_{e,x}$ and $D_{e,y}$ with 5 voters each; thus, $w_{e,x} = w_{e,y} = 5$. In each such district $D_{e,u}$ there are two voters who vote for j_e and three voters who vote for j_u . We set $\gamma_{e,u} = 1$; thus, the attacker can change the winner in this district from j_u to j_e .
- For every node $u \in V$, we create a district D_u with 5μ voters; thus, $w_u = 5\mu$. In each such district there are 2μ voters who vote for j_u and 3μ voters who vote for a . We set $\gamma_u = \mu$; thus, the attacker can change the winner in this district from a to j_u .
- There are also some districts that cannot be manipulated (i.e., $\gamma = 0$). We specify the weights and the winners of these districts.
 - For each $e \in E$, there is a district with weight $5\mu(\nu - \ell) - 5$ and winner j_e .
 - For each $u \in V$, there is a district with weight $5\mu(\nu - \ell - 1)$ and winner j_u .
 - Finally, there is a district with weight $5\mu(\nu - \ell) + 1$ and winner p .

The budgets are $B_A = \nu + \mu$ and $B_D = 0$.

We have $\text{SW}(a) = 5\mu\nu$, $\text{SW}(p) = 5\mu(\nu - \ell) + 1$, $\text{SW}(j_e) = 5\mu(\nu - \ell) - 5$ for each $e \in E$, and $\text{SW}(j_u) = 5\mu(\nu - \ell - 1) + 5|\{e \in E : u \in e\}| \leq 5\mu(\nu - \ell)$ for each $u \in V$. Hence, the true winner of the election is candidate a . We show that the attacker can make p the winner if and only if $\langle G, \ell \rangle$ is a yes-instance of INDEPENDENT SET, i.e., there is an independent set of size ℓ in G .

Suppose first that there is an independent set $V' \subseteq V$, $|V'| = \ell$, in G . The following manipulation strategy makes p the winner. For every $u \in V'$, change the winner of district D_u from a to j_u , and for every $e \in E$ such that $u \in e$, change the winner of district $D_{e,u}$ from j_u to j_e . Note that since V' is an independent set, the weight of each candidate j_e , $e \in E$, increases by at most 5. Let ω_c denote the weight of each candidate $c \in C$ after manipulation. We have $\omega_a = 5\mu(\nu - \ell)$, $\omega_p = 5\mu(\nu - \ell) + 1$, $\omega_{j_e} \in \{5\mu(\nu - \ell) - 5, 5\mu(\nu - \ell)\}$ for each $e \in E$, and $\omega_{j_u} = 5\mu(\nu - \ell)$ for each $u \in V$; thus, candidate p becomes the winner of the election.

Conversely, suppose that the attacker has a manipulation that makes p the election winner; for each $c \in C$, let ω_c be the weight of candidate c after this manipulation. Since p cannot be made the winner in any additional district, we have $\omega_p = 5\mu(\nu - \ell) + 1$. Let V' be the set of all nodes $u \in V$ such that the attacker changes the winner of D_u from a to j_u . Since $\omega_a \leq \omega_p$, we have $|V'| \geq \ell$; we will now argue that V' is an independent set. Indeed, consider a node $u \in V'$. Changing the winner in D_u from a to j_u increases the weight of j_u by 5μ . As we have $\omega_{j_u} \leq \omega_p$, the manipulation needs to reduce the weight of j_u by $5|\{e \in E : u \in e\}|$. The only way to do so is to change the winner from j_u to j_e in all districts $D_{e,u}$ with $u \in e$, thereby increasing the weight of j_e by 5. Now, suppose that $x, y \in V'$ and $e = \{x, y\} \in E$. Then the manipulation increases the weight of j_e by 10, so we have $\omega_{j_e} = 5\mu(\nu - \ell) + 5 > \omega_p$, a contradiction. Thus, V' is an independent set. \square

PD-MAN remains NP-hard even if all districts have the same weight; however, under this assumption this problem can be placed in NP, i.e., the unweighted variant of PD-MAN is strictly easier than its weighted variant unless $\text{NP} = \Sigma_2^P$ (which is believed to be highly unlikely).

Theorem 4.8. PD-MAN is NP-complete when $w_i = 1$ for all $i \in [k]$.

Proof. To see that PD-MAN is in NP when $w_i = 1$ for all $i \in [k]$, it suffices to note that PD-REC is in P under this assumption (Theorem 4.3). To prove that PD-MAN remains NP-hard even in this case, we again provide a reduction from INDEPENDENT SET; see Definition 2.4.

Given an instance $\langle G, \ell \rangle$ of INDEPENDENT SET, where $G = (V, E)$, we construct an instance of PD-MAN as follows. Let $\nu = |V|$, $\mu = |E|$, and for each $u \in V$ let $\deg(u)$ denote the degree of vertex u in G ; without loss of generality, we can assume that $\mu > 0$ and $\deg(u) > 0$ for all $u \in V$. Let $A_V = \{a_u : u \in V\}$, $A'_V = \{b_u : u \in V\}$, $A_E = \{a_e : e \in E\}$, and set $C = A_V \cup A'_V \cup A_E \cup \{p\}$, where p is the attacker's preferred candidate; thus, $|C| = 2\nu + \mu + 1$. The tie-breaking order \succ is defined so that $p \succ c$ for all $c \in C \setminus \{p\}$, and $c \succ c'$ for all $c \in A_V, c' \in A_E$. We create the following districts (note that the weight of each district is 1).

- For every edge $e = \{x, y\} \in E$, we create two districts $D_{e,x}$ and $D_{e,y}$ with 5 voters each. In each such district $D_{e,u}$ there are two voters who vote for a_u and three voters who vote for a_e . We set $\gamma_{e,u} = 1$; thus, the attacker can change the winner in this district from a_e to a_u .
- For every vertex $u \in V$, we create a district D_u with two voters who vote for a_u and three voters who vote for b_u . We set $\gamma_u = 1$; thus, the attacker can change the winner in this district from b_u to a_u .
- There are also some districts that cannot be manipulated (i.e., $\gamma = 0$); for concreteness, we assume that each such district has five voters, and they all vote for the same candidate:
 - For each $e \in E$, there are $\mu - 1$ districts where the winner is a_e ;
 - For each $u \in V$, there are $\mu - \deg(u)$ districts where the winner is a_u ;
 - There are μ districts where the winner is p .

The budgets are $B_A = 2\mu + \ell$ and $B_D = \ell$. Thus, we have $\text{SW}(p) = \mu$, $\text{SW}(a_e) = \mu + 1$ for each $e \in E$, $\text{SW}(a_u) = \mu - \deg(u) < \mu$ and $\text{SW}(b_u) = 1$ for each $u \in V$. Consequently, the true winner is one of the candidates in A_E .

We will now argue that G admits an independent set of size ℓ if and only if there is a winning strategy for the attacker.

Suppose first that $V' \subseteq V$ is an independent set of size ℓ . Consider the following strategy for the attacker, which changes votes in exactly B_A districts:

- For each $e = \{x, y\} \in E$, change the winner of $D_{e,x}$ from a_e to a_x , and the winner of $D_{e,y}$ from a_e to a_y .

- For each $u \in V'$, change the winner of D_u from b_u to a_u .

Let ω_c denote the weight of each candidate $c \in C$ after this manipulation. We have $\omega_p = \mu$, $\omega_{a_e} = \mu - 1$ for each $e \in E$, $\omega_{a_u} = \mu$ for each $u \in V \setminus V'$, $\omega_{a_u} = \mu + 1$ for each $u \in V'$, and $\omega_{b_u} = 0$ each $u \in V$. Hence, in the manipulated instance the winner is chosen from $\{a_u : u \in V'\}$ according to the tie-breaking rule.

Even though p does not win the election at this point, we will now show that p becomes the winner once the defender respond optimally to this manipulation.

First, we show that the defender can make p win. To this end, for each $u \in V'$ the defender can pick one edge e^u such that $u \in e^u$ and demand a recount in district $D_{e^u,u}$; altogether, this strategy requires recounting $\ell = B_D$ districts. Since V' is an independent set, after the recount the weight of each candidate a_e , $e \in E$, is at most μ , and also the weight of each candidate a_u , $u \in V$, is at most μ . Since $\omega_p = \mu$ and p is favored by the tie-breaking rule, p becomes the election winner.

We will now argue that for every candidate a that can be made the election winner by recounting at most ℓ districts we have $\text{SW}(a) \leq \text{SW}(p)$; since defender breaks ties according to \succ , this proves that the defender will choose a recounting strategy that makes p win. To see this, suppose for the sake of contradiction that there is a recounting strategy that results in a candidate a with $\text{SW}(a) > \text{SW}(p)$ becoming the election winner. Note that $\text{SW}(a) > \text{SW}(p)$ implies that $a \in A_E$ and hence $\omega_a = \mu - 1$. Let ω'_c denote the weight of each candidate $c \in C$ after the recount. The attacker does not transfer any district to a , which implies that $\omega'_a \leq \text{SW}(a) = \mu + 1$. On the other hand, since $\omega'_p = \mu$, and the tie-breaking rule favors p over all other candidates, we have $\omega'_a \geq \mu + 1$. Thus, $\omega'_a = \mu + 1$. This means that $\omega'_a - \omega_a = 2$, i.e., if $a = a_e$ and $e = \{x, y\}$, both $D_{e,x}$ and $D_{e,y}$ are recounted. We will now argue that $x, y \in V'$. Indeed, for each $u \in V'$ we have $\omega_{a_u} = \mu + 1$; on the other hand, $a_u \succ a$ and hence $\omega'_{a_u} < \omega'_a = \mu + 1$. Thus, the defender must demand that for each $u \in V'$ the district D_u is recounted; since $B_D = \ell$, the set of recounted districts is exactly V' , and hence $x, y \in V'$, as claimed. But this is a contradiction, since $\{x, y\} \in E$, and V' is an independent set. This proves that if $\langle G, \ell \rangle$ is a yes-instance of INDEPENDENT SET, there is a winning strategy for the attacker.

Conversely, suppose that G has no independent set of size ℓ . Consider an attack that changes votes in at most B_A districts. For each $c \in C$, let ω_c denote the weight of candidate c after the attack. Note that $\omega_p = \mu$; moreover, any attack can only increase the weight of candidates in A_V , and the weight of any such candidate after the attack is at most $\mu + 1$. Let $V' = \{u \in V : \omega_{a_u} = \mu + 1\}$ and $C' = \{a_u \in C : u \in V'\}$. We consider three cases:

- $|V'| > B_D$. Since recounting a district only reduces the weight of one candidate, the weight of some candidate $a_u \in C'$ will still be $\mu + 1$ after the recount, so p will be beaten by a_u .
- $|V'| \leq B_D$, V' is not an independent set. Pick an edge $e^* = \{x, y\}$ such that $x, y \in V'$, and consider the following recounting strategy. For each $u \in V' \setminus \{x, y\}$, the defender picks one edge e^u such that $u \in e^u$, and demands a recount in districts $D_{e^u,u}$ for each $u \in V' \setminus \{x, y\}$ as well as in $D_{e^*,x}$ and in $D_{e^*,y}$. This recounting strategy requires recounting $|V'| \leq B_D$ districts, reduces the weight of every candidate $c \in C'$ by 1 and increases the weight of a_{e^*} by 2. Thus, after the recount the weight of a_{e^*} is $\mu + 1$, whereas the weights of all candidates in $C \setminus A_E$ do not exceed μ , so the winner is a candidate $a \in A_E$. Since $\text{SW}(a) > \text{SW}(p)$, this means that p cannot win after the recount.
- $|V'| \leq B_D$, V' is an independent set. Then by our assumption $|V'| < \ell = B_D$. Consider an edge $e^* = \{x, y\}$ with $x \in V', y \notin V'$. For each $u \in V' \setminus \{x\}$, the defender can pick one edge e^u such that $u \in e^u$, and demand a recount in districts $D_{e^u,u}$ for each $u \in V' \setminus \{x\}$ as well as in $D_{e^*,x}$ and in $D_{e^*,y}$. This strategy requires recounting $|V'| + 1 \leq B_D$ districts and ensures that after the recount the weight of e^* is $\mu + 1$, whereas the weights of all candidates in $C \setminus A_E$ are at most μ ,

so the winner is a candidate $a \in A_E$. Since $\text{SW}(a) > \text{SW}(p)$, this means that p cannot win after the recount.

Hence, the attacker cannot win in any case. This completes the proof. \square

Theorem 4.7 holds even for $B_D = 0$, but for Theorems 4.6 and 4.8 this is not the case. Indeed, PD-MAN is in NP when $B_D = 0$, since the attacker simply needs to guess a manipulation and check whether it makes p the winner. The unweighted problem (Theorem 4.8) can be shown to be in P when $B_D = 0$; the argument uses a reduction to nonuniform bribery similar to the one in the proof of Theorem 4.3. Thus, recounting has a clear impact on the complexity of the attacker's problem.

5 Regular Manipulations

In our model, the attacker does not have to transfer votes to his preferred candidate p in the manipulated districts; indeed, he may even choose to transfer votes *from* p to another candidate. However, manipulations that give additional votes to candidates other than p are counter-intuitive and may be difficult to implement in practice. Therefore, in this section we study what happens if the attacker is limited to transferring votes (in case of PV) or vote weight (in case of PD) to his preferred candidate p .

Definition 5.1 (Regular manipulation). Let p be the preferred candidate of the attacker. A manipulation $(M, \tilde{\mathbf{v}})$ is said to be *regular* if for every district $i \in M$ it holds that

- the voting rule is PV and $\tilde{v}_{ia} \leq v_{ia}$ for all $a \in C \setminus \{p\}$;
- the voting rule is PD and in $\tilde{\mathbf{v}}$ candidate p is the winner in each district in M .

The difference between our general model and the one where the attacker is limited to using regular manipulations is similar to the difference between swap bribery and shift bribery [Elkind *et al.*, 2009]: in swap bribery the attacker can change the vote in any way he likes subject to budget constraints, while in shift bribery he is limited to shifting his preferred candidate in voters' rankings.

One may expect that the restriction to regular manipulations is without loss of generality: indeed, why would the attacker want to transfer votes to candidates other than p ? However, our next example shows that this intuition is incorrect.

Example 5.2. We show an example for PV; the example also works for PD by setting $w_i = n_i$ for every $i \in [k]$. Consider an instance with 3 candidates $\{a, b, p\}$ and 19 voters who are distributed to 12 districts. The vote profile is as follows:

Candidate	D_1	D_2	D_3, \dots, D_8	D_9, \dots, D_{12}
a	0	3	1	0
p	6	0	0	0
b	0	0	0	1

Also, $B_A = 2$, $B_D = 1$, and $\gamma_i = n_i$ for all $i \in \{1, \dots, 12\}$. The true winner is candidate a with 9 votes, compared to the 6 votes of p and the 4 votes of b . No regular manipulation can make p win: no matter what the attacker does, by recounting at most one district the defender can ensure that a gets at least 8 votes and p gets at most 7 votes.

Now, consider a non-regular manipulation that distorts all votes in D_1 in favor of b , and all votes in D_2 in favor of p . Then in the distorted profile a has 6 votes and p has 3 votes, and b wins with 10 votes. If the defender does not recount D_1 , b remains the winner after recounting, and if she does recount it, p becomes the winner. Crucially, since $\text{SW}(b) < \text{SW}(p)$, the defender prefers the latter option, so p wins after the recount. \square

Example 5.2 shows that only considering regular manipulations may be suboptimal for the attacker. However, the attacker may be limited to regular manipulations by practical considerations. For instance, the election officials in the manipulated districts may find it difficult to follow complex instructions. Thus, it is interesting to understand if focusing on regular manipulations affects the complexity of the problems we consider.

The following observation will be useful for our analysis.

Proposition 5.3. *Let $\mathcal{R} \in \{\text{PV}, \text{PD}\}$, and let $(M, \tilde{\mathbf{v}})$ be a winning regular manipulation. Then for every recounting strategy $R \subseteq M$ it holds that after the recount p is the election winner.*

Proof. Let $B = \{b \in C \setminus \{p\} : \text{SW}^{\mathcal{R}}(b) < \text{SW}^{\mathcal{R}}(p) \text{ or } \text{SW}^{\mathcal{R}}(b) = \text{SW}^{\mathcal{R}}(p), p \succ b\}$. Since M is a winning manipulation, the winner after recounting is either p or some candidate in B ; we will show that, since M is regular, the latter case is, in fact, impossible. For each $c \in C$, let s_c denote the number of votes/vote weight of c after the recount. Since M is a regular manipulation, for each candidate $b \in B$

$$s_b \leq \text{SW}^{\mathcal{R}}(b) \leq \text{SW}^{\mathcal{R}}(p) \leq s_p,$$

and if $b \succ p$, the second inequality is strict. Thus, p beats every candidate in B after recounting, so no such candidate can be the election winner. \square

By setting $R = \emptyset$ in Proposition 5.3, we observe that p is the winner at $\tilde{\mathbf{v}}$, i.e., the situation described in Example 5.2, where p does not win after the manipulation, but the defender is forced to make p the election winner, cannot occur if the attacker is limited to regular manipulations.

In what follows, we consider the complexity of \mathcal{R} -REC and \mathcal{R} -MAN for $\mathcal{R} \in \{\text{PV}, \text{PD}\}$ under the assumption that the attacker is limited to regular manipulations; we denote these versions of our problems by \mathcal{R} -REC-REG and \mathcal{R} -MAN-REG, respectively. We first consider the defender's problem (\mathcal{R} -REC-REG) and then the attacker's problem (\mathcal{R} -MAN-REG).

5.1 The Defender's Problem

Let $\mathcal{R} \in \{\text{PV}, \text{PD}\}$, and consider a regular manipulation $(M, \tilde{\mathbf{v}})$. Recall that we assume that $|M| > B_{\mathcal{D}}$. Note that if p is not a winner at $\tilde{\mathbf{v}}$, the attacker necessarily By Proposition 5.3, we can assume that p is the winner at $\tilde{\mathbf{v}}$. The defender can then try the following greedy strategy. Initially, it defines the set of *provisional winners* to consist of p . Then, for each $a \in C \setminus \{p\}$ such that $\text{SW}^{\mathcal{R}}(a) > \text{SW}^{\mathcal{R}}(p)$ or $\text{SW}^{\mathcal{R}}(a) = \text{SW}^{\mathcal{R}}(p)$ and $a \succ p$ the algorithm sorts the districts in M in non-increasing order in terms of the quantity $(v_{ia} - v_{ip}) - (\tilde{v}_{ia} - \tilde{v}_{ip})$ for PV, and the quantity $(w_{ia} - w_{ip}) - (\tilde{w}_{ia} - \tilde{w}_{ip})$ for PD; ties are broken arbitrarily. Next, it checks what happens if the first $B_{\mathcal{D}}$ districts in this order are recounted; if this results in a candidate $b \in C \setminus \{p\}$ with $\text{SW}^{\mathcal{R}}(b) > \text{SW}^{\mathcal{R}}(p)$ or $\text{SW}^{\mathcal{R}}(b) = \text{SW}^{\mathcal{R}}(p)$, $b \succ p$, winning the election, the defender adds b to the set of *provisional winners*. Finally, it outputs the provisional winner with the maximum social welfare, breaking ties according to \succ . We refer to this algorithm as *greedy recounting*; note that its running time is polynomial in the input size.

Lemma 5.4. *Let $\mathcal{R} \in \{\text{PV}, \text{PD}\}$. Suppose that the attacker uses a regular manipulation $(M, \tilde{\mathbf{v}})$. Then greedy recounting outputs p if and only if $(M, \tilde{\mathbf{v}})$ is a winning strategy for the attacker.*

Proof. We provide the proof for $\mathcal{R} = \text{PV}$; the proof for $\mathcal{R} = \text{PD}$ is obtained by replacing candidates' vote counts with weights. Consider the attacker's strategy $(M, \tilde{\mathbf{v}})$. Given a set of districts $R \subseteq M$ and a candidate $a \in C$, let $s_a(R)$ denote the number of votes that candidate a gets after the attacker manipulates according to $(M, \tilde{\mathbf{v}})$ and the defender recounts the districts in R . Given two candidates $a, b \in C$ and a subset of districts $R \subset M$, we write $s_a(R) \succ s_b(R)$ if $s_a(R) > s_b(R)$ or $s_a(R) = s_b(R)$ and $a \succ b$.

Since $(M, \tilde{\mathbf{v}})$ is a regular manipulation, for every $R \subseteq M$ we have

$$s_p(R) \geq \text{SW}(p) \quad \text{and} \quad s_c(R) \leq \text{SW}(c) \text{ for all } c \in C \setminus \{p\}.$$

Suppose that $(M, \tilde{\mathbf{v}})$ is not a winning strategy for the attacker. Then there exists a subset R^* of at most B_D districts such that after recounting R^* the winner is a candidate $a \in C \setminus \{p\}$ such that $\text{SW}(a) > \text{SW}(p)$ or $\text{SW}(a) = \text{SW}(p)$, $a \succ p$. We can assume without loss of generality that $|R^*| = B_D$. Indeed, suppose that $|R^*| < B_D$ and a wins after recounting the districts in R^* . Let Q be an arbitrary set of B_D districts such that $R^* \subset Q \subseteq M$, and suppose that once the votes in Q are recounted, the winner is c . Since $(M, \tilde{\mathbf{v}})$ is a regular manipulation, we have

$$\text{SW}(c) \geq s_c(Q) \geq s_a(Q) \geq s_a(R^*) \geq s_p(R^*) \geq \text{SW}(p).$$

If any of these inequalities is strict, we have $\text{SW}(c) > \text{SW}(p)$. Otherwise, the second inequality implies $c \succ a$ and the fourth inequality implies $a \succ p$, so we have $\text{SW}(c) = \text{SW}(p)$, $c \succ p$. In either case, recounting the districts in Q results in an outcome that the defender prefers to p .

We will argue that the greedy recounting algorithm does not output p . Let R_a be the set of districts recounted by this algorithm when it considers candidate a (i.e., R_a contains the first B_D districts in non-increasing order of the quantity $(v_{ia} - v_{ip}) - (\tilde{v}_{ia} - \tilde{v}_{ip})$), and let b be the winner after recounting the districts in R_a . Consider the following possibilities.

- $b \neq p$. If $\text{SW}(b) > \text{SW}(p)$ or $\text{SW}(b) = \text{SW}(p)$, $b \succ p$, the algorithm adds b to the set of provisional winners and thus does not output p . Otherwise, we have

$$s_p(R_a) \geq \text{SW}(p) \geq \text{SW}(b) \geq s_b(R_a);$$

if $b \succ p$, the second inequality is strict. Consequently, $s_p(R_a) > s_b(R_a)$, so b cannot be the winner after recounting the districts in R_a , a contradiction.

- $b = p$. By our choice of R_a and the fact that $|R_a| = |R^*|$ we have

$$\begin{aligned} 0 &\leq s_a(R^*) - s_p(R^*) \\ &= \sum_{i \notin R^*} (\tilde{v}_{ia} - \tilde{v}_{ip}) + \sum_{i \in R^*} (v_{ia} - v_{ip}) \\ &= \sum_{i \in [k]} (\tilde{v}_{ia} - \tilde{v}_{ip}) + \sum_{i \in R^*} (v_{ia} - v_{ip} - \tilde{v}_{ia} + \tilde{v}_{ip}) \\ &\leq \sum_{i \in [k]} (\tilde{v}_{ia} - \tilde{v}_{ip}) + \sum_{i \in R_a} (v_{ia} - v_{ip} - \tilde{v}_{ia} + \tilde{v}_{ip}) \\ &= \sum_{i \notin R_a} (\tilde{v}_{ia} - \tilde{v}_{ip}) + \sum_{i \in R_a} (v_{ia} - v_{ip}) \\ &= s_a(R_a) - s_p(R_a). \end{aligned}$$

Combining this with the fact that $s_p(R_a) > s_a(R_a)$, we conclude that $s_p(R_a) = s_a(R_a)$ and $p \succ a$. Thus, all inequalities above are, in fact, equalities, so in particular $s_p(R^*) = s_a(R^*)$. Together with $p \succ a$ this implies that $s_p(R^*) > s_a(R^*)$, a contradiction with our assumption that a is the winner after recounting R^* .

This completes the proof. \square

Notably, greedy recounting does not constitute an algorithm for $\mathcal{R}\text{-REC-REG}$: it is unable to decide whether there is a recounting strategy that results in a specific candidate becoming the election winner. However, it serves as a $1/2$ -approximation algorithm for the defender: it outputs a candidate a such that for every candidate a' that can be made a winner by recounting at most B_D districts it holds that $\text{SW}(a) \geq \text{SW}(a')/2$.

Theorem 5.5. *Greedy recounting is a $1/2$ -approximation algorithm for the optimization versions of PV-REC-REG and PD-REC-REG.*

Proof. We focus on PV; the analysis can be adapted for PD, by modifying the notation so as to take into account the weights of the candidates rather than their vote counts.

Consider an instance with a set of candidates C , $|C| = m$, and let p be the attacker's preferred candidate. Suppose that the attacker uses a regular manipulation $(M, \tilde{\mathbf{v}})$; we will assume that p is the winner in the manipulated instance, as otherwise the attacker does not have an incentive to manipulate. For each $c \in C$, let s_c denote the vote count of candidate c in the manipulated instance. If p is the winner before the manipulation or if no recounting strategy can change the outcome, then greedy recounting is trivially optimal. Hence, in the remainder of the proof we assume that there is a candidate $b \neq p$ such that $\text{SW}(b) > \text{SW}(p)$ or $\text{SW}(b) = \text{SW}(p)$, $b \succ p$, such that the defender can make b win; let c be the defender's most preferred candidate with this property, and let R be a recounting strategy that results in c becoming the winner after a recount. We consider the round in which greedy recounting examines candidate c ; suppose that greedy recounting selects a subset of districts G . Let $A = C \setminus \{c, p\}$ denote the set of the remaining $m - 2$ candidates.

We define the following pairwise disjoint sets of districts:

- $I_G = G \setminus R$;
- $I_O = R \setminus G$;
- $I_{OG} = R \cap G$;
- $I_{\overline{OG}} = M \setminus (R \cup G)$.

Given a set of districts $I \subseteq M$ and a subset of candidates $J \subseteq C \setminus \{p\}$, let

$$\Delta(I, J) = \sum_{i \in I} \sum_{a \in J} (v_{ia} - \tilde{v}_{ia})$$

denote the total number of votes in I that are transferred by the attacker from candidates in J to p ; if J or I is a singleton, we omit the curly braces and write $\Delta(I, j)$ or $\Delta(i, J)$, respectively. Since $(M, \tilde{\mathbf{v}})$ is a regular manipulation, we have

$$s_p = \text{SW}(p) + \Delta(M, c) + \Delta(M, A), \quad (1)$$

$$s_c = \text{SW}(c) - \Delta(M, c), \quad (2)$$

$$s_a = \text{SW}(a) - \Delta(M, a) \text{ for each } a \in A. \quad (3)$$

Since recounting the districts in $R = I_O \cup I_{OG}$ ensures that c becomes the winner, we obtain

$$s_c + \Delta(I_O \cup I_{OG}, c) \geq s_p - \Delta(I_O \cup I_{OG}, c) - \Delta(I_O \cup I_{OG}, A); \quad (4)$$

if $p \succ c$, this inequality is strict.

Next, let us focus on the behavior of the greedy recounting. Let $g \in I_G$ and $o \in I_O$. Since the greedy algorithm selects g , but not o , we have

$$(v_{gc} - v_{gp}) - (\tilde{v}_{gc} - \tilde{v}_{gp}) \geq (v_{oc} - v_{op}) - (\tilde{v}_{oc} - \tilde{v}_{op}).$$

Since $v_{ic} - \tilde{v}_{ic} = \Delta(i, c)$ and $\tilde{v}_{ip} - v_{ip} = \Delta(i, c) + \Delta(i, A)$ for every $i \in M$, we then obtain

$$2\Delta(g, c) + \Delta(g, A) \geq 2\Delta(o, c) + \Delta(o, A).$$

Since $|G| = B_{\mathcal{D}}$, $|R| \leq B_{\mathcal{D}}$, we have $|I_G| \geq |I_O|$. Pick a subset of districts $I'_G \subseteq I_G$ with $|I'_G| = |I_O|$. We can pair each $o \in I_O$ with a unique $g \in I'_G$, and add all corresponding inequalities to get

$$2\Delta(I'_G, c) + \Delta(I'_G, A) \geq 2\Delta(I_O, c) + \Delta(I_O, A);$$

since $\Delta(i, b) \geq 0$ for all $b \in A \cup \{c\}$ and all $i \in I_G \setminus I'_G$, we get

$$2\Delta(I_G, c) + \Delta(I_G, A) \geq 2\Delta(I_O, c) + \Delta(I_O, A). \quad (5)$$

By adding inequalities (4) and (5), we obtain

$$\begin{aligned} s_c + \Delta(I_O \cup I_{OG}, c) + 2\Delta(I_G, c) + \Delta(I_G, A) \\ \geq s_p - \Delta(I_O \cup I_{OG}, c) - \Delta(I_O \cup I_{OG}, A) + 2\Delta(I_O, c) + \Delta(I_O, A), \end{aligned}$$

or, simplifying,

$$s_c + \Delta(I_G \cup I_{OG}, c) \geq s_p - \Delta(I_G \cup I_{OG}, c) - \Delta(I_G \cup I_{OG}, A); \quad (6)$$

if $p \succ c$, then Inequality (4) is strict and hence Inequality (6) is strict as well. Inequality (6) means that after recounting the districts in $G = I_G \cup I_{OG}$, c beats p , i.e., the winner is either c or another candidate $a \in A$. To conclude the proof, it suffices to show that if the winner in the recounted instance is some candidate $a \in A$ then $\text{SW}(a) \geq \frac{1}{2}\text{SW}(c)$.

By substituting expressions for s_c and s_p from (2) and (1), we can write Inequality (6) as

$$\text{SW}(c) - \Delta(I_O \cup I_{\overline{OG}}, c) \geq \text{SW}(p) + \Delta(I_O \cup I_{\overline{OG}}, c) + \Delta(I_O \cup I_{\overline{OG}}, A).$$

Since $\text{SW}(p) \geq 0$ and $\Delta(I_O \cup I_{\overline{OG}}, A) \geq 0$, it follows that

$$\Delta(I_O \cup I_{\overline{OG}}, c) \leq \frac{1}{2}\text{SW}(c). \quad (7)$$

By our assumption, recounting the districts in $G = I_G \cup I_{OG}$ results in a getting at least as many votes as c , so we obtain

$$s_a + \Delta(I_G \cup I_{OG}, a) \geq s_c + \Delta(I_G \cup I_{OG}, c).$$

By substituting expressions for s_a and s_c from (3) and (2), we can rewrite this inequality as

$$\text{SW}(a) - \Delta(I_O \cup I_{\overline{OG}}, a) \geq \text{SW}(c) - \Delta(I_O \cup I_{\overline{OG}}, c).$$

Finally, since $\Delta(I_O \cup I_{\overline{OG}}, a) \geq 0$, from Inequality (7) we obtain

$$\text{SW}(a) \geq \frac{1}{2}\text{SW}(c),$$

as desired. \square

In fact, the bound on the approximation ratio provided by Theorem 5.5 is essentially tight.

Theorem 5.6. *For any constant $\varepsilon > 0$, neither PV-REC-REG nor PD-REC-REG admits a polynomial-time $(\frac{1}{2} + \varepsilon)$ -approximation algorithm unless P = NP, even when $m = 3$.*

Proof. We focus on PV; the proof for PD follows by setting the weight of each district in the reduction below to be equal to the number of voters therein.

We will show that if there is a $(\frac{1}{2} + \varepsilon)$ -approximation algorithm for PV-REC-REG, it can be used to solve PARTITION; see Definition 2.5.

Given an instance X of PARTITION with $|X| = \ell$, we construct an instance of PV-REC-REG with a set of candidates $C = \{a, b, p\}$, where p is the attacker's preferred candidate, as follows. Let $y = \sum_{x \in X} x$, and $z = \lceil y/\varepsilon \rceil$. Without loss of generality, we assume that all integers in X are divisible by 4 and hence $y \geq 4$. In what follows, we describe each district D_i by a tuple (v_{ia}, v_{ib}, v_{ip}) . The districts are partitioned into the following three sets I_1, I_2 and I_3 :

- For each $x \in X$, there is a district in I_1 with votes $(0, 2x\ell, 0)$, which are distorted to $(0, 0, 2x\ell)$.
- I_2 consists of $2z\ell$ districts with votes $(1, 0, 0)$, which are distorted to $(0, 0, 1)$.
- I_3 consists of two districts with votes $(2z\ell + y\ell + 2\ell, 0, 0)$ and $(0, 2z\ell, 0)$, which are not distorted.

Finally, the budget of the defender is $B_D = \ell - 1$.

Since votes are transferred to p only, the manipulation is regular. The vote counts of the candidates before and after the manipulation are as follows:

	True vote counts (SW)	Distorted vote counts
a	$4z\ell + y\ell + 2\ell$	$2z\ell + y\ell + 2\ell$
b	$2z\ell + 2y\ell$	$2z\ell$
p	0	$2z\ell + 2y\ell$

Therefore, before the manipulation the winner is a , and the manipulation makes p the election winner. Since

$$\frac{\text{SW}(c)}{\text{SW}(a)} \leq \frac{2z\ell + 2y\ell}{4z\ell + y\ell + 2\ell} < \frac{1}{2} + \varepsilon \quad \text{for each } c \in \{b, p\},$$

any $(\frac{1}{2} + \varepsilon)$ -approximation algorithm can decide whether a can be restored as the winner. We will now argue that this is equivalent to deciding whether the given instance of PARTITION is a yes-instance.

Suppose that X is a yes-instance of PARTITION, i.e., there exists a subset $X' \subseteq X$ such that $\sum_{x \in X'} x = y/2$; note that $|X'| \leq \ell - 1$. Then, by recounting the $|X'|$ districts of I_1 that correspond to the integers in X' , the defender lowers the vote count of p by $y\ell$ and increases the vote count of b by $y\ell$. As a result, a gets $2z\ell + y\ell + 2\ell$ votes, b gets $2z\ell + y\ell$ votes, and p gets $2z\ell + y\ell$ votes. Therefore, a is restored as the election winner.

Conversely, suppose that there is no subset $X' \subseteq X$ such that $\sum_{x \in X'} x = y/2$. Since all integers in X are divisible by 4, $y/2$ is even and hence for any $X' \subseteq X$ we have $|\sum_{x \in X'} x - y/2| \geq 2$. Suppose that the defender recounts districts in I_1 that correspond to a subset $X' \subseteq X$ as well as q districts in I_2 ; let $u = \sum_{x \in X'} x$. Since $q < \ell$, the vote count of candidate a after the recount is

$$2z\ell + y\ell + 2\ell + q < 2z\ell + y\ell + 3\ell.$$

If $u \geq y/2 + 2$, then the vote count of b after the recount is

$$2z\ell + 2u\ell \geq 2z\ell + y\ell + 4\ell.$$

Otherwise, $u \leq y/2 - 2$. Since $q < \ell$, the vote count of p after the recount is

$$\begin{aligned} 2z\ell - q + 2\ell \sum_{x \in X \setminus X'} x &= 2z\ell - q + 2y\ell - 2u\ell \\ &> 2z\ell + 2y\ell - 2u\ell - \ell \\ &\geq 2z\ell + y\ell + 3\ell. \end{aligned}$$

Therefore, in either case one of b or p gets more votes than a , and the theorem follows. \square

5.2 The Attacker's Problem

Greedy recounting also plays an important role in our analysis of \mathcal{R} -MAN-REG. Indeed, even though greedy recounting does not constitute an algorithm for \mathcal{R} -REC-REG, Lemma 5.4 suffices to establish that \mathcal{R} -MAN-REG is in NP: the attacker can guess a regular manipulation and use greedy recounting to verify whether it is successful. For PV, this complexity upper bound is tight: one can check that in the hardness proofs in Theorem 3.3 the attacker's successful manipulation strategy is regular, and hence PV-MAN-REG is NP-complete. We summarize these observations in the following theorem.

Theorem 5.7. *PV-MAN-REG is NP-complete. The hardness result holds even if $m = 3$ or if the input vote profile and district weights are given in unary.*

We cannot use the same approach to show that PD-MAN-REG is NP-hard: the hardness proofs in Theorems 4.6–4.8 rely on the attacker using a non-regular strategy. In fact, it turns out that PD-MAN-REG is polynomial-time solvable, i.e., for PD focusing on regular manipulations brings down the complexity of the attacker's problem from Σ_2^P to P.

Theorem 5.8. *PD-MAN-REG can be solved in polynomial time.*

Proof. Let p be the attacker's preferred candidate, and let $A = \{c \in C : \text{SW}(c) > \text{SW}(p) \text{ or } \text{SW}(c) = \text{SW}(p), c \succ p\}$ be the set of candidates that are preferred to p by the defender. For each $c \in C \setminus \{p\}$, we denote by S_c the set of districts that have c as their true winner and can be manipulated in favor of p . Let $S = \bigcup_{c \in C \setminus \{p\}} S_c$ denote the set of all districts that can be manipulated in favor of p . Note that for every $c \in C$ the set S_c can be computed efficiently: the problem of deciding if the winner of district D_i can be changed to p can be viewed as an instance of nonuniform bribery under Plurality with prices in $\{0, 1\}$ and budget γ_i , and nonuniform bribery is in P for the Plurality rule (see the proof of Theorem 4.3 for the definition of nonuniform bribery and references).

Since the manipulation is regular, the attacker's strategy can be identified with a subset $M \subseteq S$. Let $\ell = \min\{B_A, |S|\}$ be the maximum number of districts that can be manipulated. For any set $Q \subseteq S$, $|Q| \leq \ell$, let $f(Q)$ be the set that consists of $\ell - |Q|$ heaviest districts in $S \setminus Q$, with ties broken arbitrarily; thus, $|f(Q) \cup Q| = \ell$. Our algorithm is based on the following lemma.

Lemma 5.9. *Consider a subset $Q \subset S$ such that there exists a winning regular manipulation M , $|M| \leq \ell$, with $Q \subset M$. Suppose that when the attacker manipulates the districts in $Q \cup f(Q)$, there is a candidate $a \in A$ such that the defender can make a beat p by recounting at most B_D districts. Let $S_a^{\max} = \arg \max_{j \in S_a \setminus Q} w_j$. Then*

- (i) $S_a \setminus Q \neq \emptyset$, and
- (ii) for each $i \in S_a^{\max}$ there is a winning regular manipulation M' , $|M'| \leq \ell$, with $Q \cup \{i\} \subseteq M'$.

Before we prove this lemma, we will explain how to use it to find a winning regular manipulation if it exists. The algorithm proceeds as follows.

1. Set $Q = \emptyset$.
2. Apply greedy recounting to $Q \cup f(Q)$ to check whether $Q \cup f(Q)$ is a winning regular manipulation. If yes, terminate and return $Q \cup f(Q)$. Otherwise greedy recounting returns a candidate $a \in A$ such that the defender can make a beat p by recounting at most B_D districts.
3. If $S_a \setminus Q = \emptyset$ or $|Q| = \ell$, then output \emptyset . Otherwise, select an arbitrary $i \in S_a^{\max}$, set $Q \leftarrow Q \cup \{i\}$, and go back to Step 2.

By Lemma 5.4, if the algorithm returns $Q \cup f(Q)$ at the end of Step 2, then $Q \cup f(Q)$ is a winning regular manipulation. Otherwise, by Lemma 5.9, there is no winning strategy. This shows that our algorithm is correct. To see that it runs in polynomial time, note that every execution of Step 2 increases $|Q|$ by 1, and $|Q|$ is bounded from above by ℓ .

To complete the proof, it remains to prove Lemma 5.9

Proof of Lemma 5.9. Suppose that there exist Q , M and a that satisfy the conditions in the statement of the lemma. For each candidate $c \in C$ and each $X \subseteq S$, let $s_c(X)$ denote the weight of c after the districts in X have been manipulated in favor of p . We prove each claim of the lemma separately.

Proof of claim (i). We will prove a stronger claim, namely, that $M \cap (S_a \setminus Q) \neq \emptyset$.

Suppose for the sake of contradiction that $M \cap (S_a \setminus Q) = \emptyset$. We will argue that in this case $Q \cup f(Q)$ is a winning manipulation, thereby obtaining a contradiction with the assumptions of the lemma. To this end, we consider an arbitrary recounting strategy $R' \subseteq Q \cup f(Q)$, $|R'| \leq B_D$, transform it into a recounting strategy $R \subseteq M$, and use the fact that M is a winning manipulation.

Since $Q \subseteq M$, we have $M \cap S_a \subseteq Q$ and $(M \setminus Q) \cap S_a = \emptyset$. Hence,

$$\sum_{i \in M \setminus Q} w_{ia} = \sum_{i \in (M \setminus Q) \cap S_a} w_i = 0. \quad (8)$$

Fix a recounting strategy $R' \subseteq Q \cup f(Q)$, $|R'| \leq B_D$. Let R'' be the set of $\min\{|R' \cap f(Q)|, |M \setminus Q|\}$ heaviest districts in $M \setminus Q$, and set $R = (R' \cap Q) \cup R''$. Note that $R \subseteq M$ and $|R| \leq B_D$: we have $|R| \leq |R' \cap Q| + |R' \cap f(Q)| \leq |R'| \leq B_D$. Moreover, $Q \setminus R = Q \setminus R'$, and

$$\begin{aligned} s_a((Q \cup f(Q)) \setminus R') &= \text{SW}(a) - \sum_{i \in (Q \cup f(Q)) \setminus R'} w_{ia} \\ &\leq \text{SW}(a) - \sum_{i \in Q \setminus R'} w_{ia} \\ &= \text{SW}(a) - \sum_{i \in Q \setminus R} w_{ia} - \sum_{i \in M \setminus Q} w_{ia} \\ &\leq \text{SW}(a) - \sum_{i \in Q \setminus R} w_{ia} - \sum_{i \in (M \setminus Q) \setminus R} w_{ia} \\ &= s_a(M \setminus R), \end{aligned} \quad (9)$$

where the third transition follows by (8).

Next, we claim that

$$\sum_{i \in f(Q) \setminus R'} w_i \geq \sum_{i \in (M \setminus Q) \setminus R} w_i.$$

Indeed, if $|R''| = |M \setminus Q|$, then $M \setminus Q \subseteq R$, so the right-hand side of this inequality is 0, and our claim is immediate. Otherwise, $|R''| = |f(Q) \cap R'| = |(M \setminus Q) \cap R|$ and $|f(Q)| = |M \setminus Q|$, i.e., both sums have the same number of summands. Moreover, $f(Q)$ contains the heaviest $\ell - |Q|$ districts in $S \setminus Q$, $M \setminus Q \subseteq S \setminus Q$, and $(M \setminus Q) \cap R$ consists of $|(M \setminus Q) \cap R|$ heaviest districts in $M \setminus Q$, so the claim follows.

We can now write

$$s_p((Q \cup f(Q)) \setminus R') = \text{SW}(p) + \sum_{i \in Q \setminus R'} w_i + \sum_{i \in f(Q) \setminus R'} w_i$$

$$\begin{aligned}
&\geq \text{SW}(p) + \sum_{i \in Q \setminus R} w_i + \sum_{i \in (M \setminus Q) \setminus R} w_i \\
&= s_p(M \setminus R).
\end{aligned} \tag{10}$$

Combining inequalities (9) and (10), we obtain

$$s_a((Q \cup f(Q)) \setminus R') - s_p((Q \cup f(Q)) \setminus R') \leq s_a(M \setminus R) - s_p(M \setminus R).$$

Since M is a winning manipulation, we have $s_a(M \setminus R) \leq s_p(M \setminus R)$, and if $a \succ p$, this inequality is strict. As this holds for any defender's strategy R' , it follows that $Q \cup f(Q)$ is a winning manipulation, too, which contradicts the assumptions of the lemma.

Proof of claim (ii). We have established that $S_a \setminus Q \neq \emptyset$ and hence $S_a^{\max} \neq \emptyset$. Now, suppose for the sake of contradiction that for some $i \in S_a^{\max}$ there is no winning regular manipulation M' with $|M'| \leq \ell$, $Q \cup \{i\} \subseteq M'$. Since all districts in S_a^{\max} are identical from both the attacker's and the defender's perspective, it holds that, in fact, for every $i \in S_a^{\max}$ there is no winning regular manipulation M' with $|M'| \leq \ell$, $Q \cup \{i\} \subseteq M'$.

We have argued that $M \cap (S_a \setminus Q) \neq \emptyset$; pick some $j \in M \cap (S_a \setminus Q)$. Since $Q \cup \{j\} \subseteq M$ and M is a winning regular manipulation, it follows that $j \notin S_a^{\max}$. Pick some $i \in S_a^{\max}$ and set $M' = (M \setminus \{j\}) \cup \{i\}$. We will now obtain a contradiction by showing that M' is a winning regular manipulation. Consider an arbitrary recounting strategy $R' \subseteq M'$, $|R'| \leq B_D$.

- (a) If $i \in R'$, let $R = (R' \setminus \{i\}) \cup \{j\}$ so that $|R| = |R'| \leq B_D$, and $M' \setminus R' = M \setminus R$. Since M is a winning strategy, for every $c \in A$ we have

$$s_c(M' \setminus R') - s_p(M' \setminus R') = s_c(M \setminus R) - s_p(M \setminus R) \leq 0;$$

if $c \succ p$, this inequality is strict.

- (b) If $i \notin R'$, let $R = R'$. Then for every $c \in C \setminus \{a, p\}$ we have

$$s_c(M' \setminus R') = s_c(M \setminus R),$$

and, since $w_j < w_i$, we obtain

$$\begin{aligned}
s_a(M' \setminus R') &= s_a(M \setminus R) + w_j - w_i < s_a(M \setminus R), \\
s_p(M' \setminus R') &= s_p(M \setminus R) - w_i + w_j > s_p(M \setminus R).
\end{aligned}$$

Combining these facts, for every $c \in A$ we have

$$s_c(M' \setminus R') - s_p(M' \setminus R') < s_c(M \setminus R) - s_p(M \setminus R) \leq 0.$$

Thus, both in case (a) and in case (b), p remains the winner after recounting.

This completes the proof of the lemma. \square

We have described an algorithm that finds a winning regular manipulation (and returns \emptyset if no such manipulation exists) in polynomial time. Thus, PD-MAN-REG is in P. \square

6 Conclusion and Open Problems

We have studied the problem of protecting elections by means of recounting votes in the manipulated districts. Our results offer an almost complete picture of the worst-case complexity of the problems faced by the defender and the attacker. Perhaps the most obvious open question is whether we can strengthen the NP-hardness results for PV-MAN and for PD-MAN under unary representation to Σ_2^P -completeness results. The next challenge is to extend our results beyond Plurality; e.g., leadership elections are often conducted using Plurality with Runoff, and it would be interesting to understand if similar results hold for this rule.

Our model is quite expressive: districts may have different weights, and an attacker may only be able to corrupt a fraction of votes in a district. These features of the model are intended to capture the challenges of real-world scenarios; in particular, it is typically infeasible for the attacker to change *all* votes in a district. However, it is important to understand their impact on the complexity of the problems we consider. We tried to indicate which of our hardness results hold for special cases of the model, and proved some easiness results under simplifying assumptions, but it would be good to obtain a more detailed picture, possibly using the tools of parameterized complexity. A concrete open question is whether our Σ_2^P -hardness result holds if $\gamma_i = n_i$ for all $i \in [k]$.

We contrasted our model with that of Yin *et al.* [2018], where the defender moves first and protects some of the districts from manipulation. In practice, the defender can use a variety of protective measures at different points in time, and an exciting direction for future work is to analyze what happens when the defender can split her resources among different activities, with some activities preceding the attack, and others (such as recounting) undertaken in the aftermath of the attack.

References

- P. Berman, M. Karpinski, L. L. Larmore, W. Plandowski, and W. Rytter. On the complexity of pattern matching for highly compressed two-dimensional texts. *Journal of Computer and System Sciences*, 65(2):332–350, 2002.
- R. Bredereck, P. Faliszewski, R. Niedermeier, and N. Talmon. Large-scale election campaigns: Combinatorial shift bribery. *Journal of Artificial Intelligence Research*, 55:603–652, 2016.
- L. Chen, L. Xu, S. Xu, Z. Gao, N. Shah, Y. Lu, and W. Shi. Protecting election from bribery: New approach and computational complexity characterization. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1894–1896, 2018.
- V. Conitzer and T. Walsh. Barriers to manipulation in voting. In *Handbook of Computational Social Choice*, pages 127–145. 2016.
- E. Elkind, P. Faliszewski, and A. Slinko. Swap bribery. In *Proceedings of the 2nd International Symposium on Algorithmic Game Theory (SAGT)*, pages 299–310, 2009.
- P. Faliszewski and J. Rothe. Control and bribery in voting. In *Handbook of Computational Social Choice*, pages 146–168. 2016.
- P. Faliszewski. Nonuniform bribery. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems (AAMAS)*, pages 1569–1572, 2008.
- M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

- J. A. Halderman and V. Teague. The New South Wales iVote system: Security failures and verification flaws in a live online election. In *Proceedings of the 5th International Conference on E-Voting and Identity (VoteID)*, pages 35–53, 2015.
- Y. Li, Y. Jiang, and W. Wu. Protecting elections with minimal resource consumption. In *Proceedings of the 16th Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1595–1597, 2017.
- M. Lindeman and P. B. Stark. A gentle introduction to risk-limiting audits. *IEEE Security & Privacy*, 10(5):42–49, 2012.
- R. Meir. *Strategic Voting*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2018.
- C. Schürmann. A risk-limiting audit in Denmark: A pilot. In *Proceedings of the 1st International Joint Conference on Electronic Voting (E-Vote-ID)*, pages 192–202, 2016.
- D. Springall, T. Finkenauer, Z. Durumeric, J. Kitcat, H. Hursti, M. MacAlpine, and J. A. Halderman. Security analysis of the Estonian internet voting system. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 703–715, 2014.
- M. Tambe. *Security and game theory: algorithms, deployed systems, lessons learned*. Cambridge University Press, 2011.
- Y. Yin, Y. Vorobeychik, B. An, and N. Hazon. Optimal defense against election control by deleting voter groups. *Artificial Intelligence*, 259:32–51, 2018.

	1	\dots	$2i$	$2i+1$	\dots	$2 B +2$	
x_i	$\bar{1}$					\bar{a}_i	(one copy for each $i \in \{1, \dots, A \}$)
x_0	$\bar{1}$					$\bar{0}$	(q copies)
y_i			$\bar{1}$			\bar{b}_i	($q+1$ copies for each $i \in \{1, \dots, B \}$)
y'_i			$\bar{1}$			$\bar{0}$	($q+1$ copies for each $i \in \{1, \dots, B \}$)
w_i				$\bar{1}$			$\left. \begin{array}{l} (q \text{ copies of } y_{-i} = w_i - z_i \\ \text{for each } i \in \{1, \dots, B \} \end{array} \right\}$
z_i						\bar{b}_i	

s	\bar{q}	\dots	$\bar{2q+1}$	\bar{q}	\dots	\bar{t}	(the goal)
-----	-----------	---------	--------------	-----------	---------	-----------	------------

Figure 1: Reduction from BI-SS to SSS⁺ (all blank sections are $\bar{0}$ s).

A Appendix: Proof of Lemma 4.5

In order to prove that SSS is Σ_2^P -complete, we will first show (Lemma A.1) that a variant of this problem, which we call SSS⁺, is Σ_2^P -complete. We will then explain how to reduce SSS⁺ to SSS.

An instance of SSS⁺ is given by a set of positive integers Y and two positive integers r and f . It is a yes-instance if there exists a subset $Y' \subseteq Y$ with $|Y'| = r$ such that for all $Y'' \subseteq Y'$ it holds that $\sum_{x \in Y''} x \neq f$, and a no-instance otherwise.

Lemma A.1. SSS⁺ is Σ_2^P -complete.

Proof. It is easy to see that SSS⁺ is in Σ_2^P . In the remainder of the proof, we show that this problem is Σ_2^P -hard. Consider an instance $\langle Y, r, f \rangle$ of SSS⁺. Let $q = |Y| - r$, $s = \sum_{x \in Y} x - f$. Note that $\langle Y, r, f \rangle$ is a yes-instance of SSS⁺ if and only if there exists a subset $Y' \subseteq Y$ with $|Y'| = q$ such that for all $Y'' \subseteq Y \setminus Y'$ it holds that $\sum_{y \in Y'} y + \sum_{y \in Y''} y \neq s$; thus, this instance of SSS⁺ can be equivalently described by the triple $\langle Y, q, s \rangle$.

To prove hardness of SSS⁺, we show a reduction from the BILEVEL SUBSET SUM (BI-SS) problem, which is known to be Σ_2^P -complete [Berman *et al.*, 2002].

Definition A.2 (BILEVEL SUBSET SUM (BI-SS)). An instance of BI-SS is given by a positive integer t and two sets of positive integers A and B . It is a yes-instance if there exists a set $A' \subseteq A$ such that for all $B' \subseteq B$ it holds that $\sum_{a \in A'} a + \sum_{b \in B'} b \neq t$, and a no-instance otherwise.

It is convenient to think of both BI-SS and SSS⁺ as leader-follower games. The leader acts first by selecting a subset; his aim is to prevent the sum of the integers chosen by both players from reaching a given target. The follower acts second; her aim is to select a subset so that the sum of the chosen integers equals the target. The difference between these two games is that in the former game the leader and the follower select from two different sets and there is no limit of the number of integers each of them can

choose, whereas in the latter game the leader is limited to q integers and both parties choose from the same base set.

Given an instance $\langle A, B, t \rangle$ of BI-SS, where $A = \{a_1, \dots, a_{|A|}\}$, $B = \{b_1, \dots, b_{|B|}\}$, we proceed as follows. We will represent a positive integer x as a vector of bits of length $L = \lfloor \log_2 x \rfloor + 1$, denoted $\bar{x} = \overline{x^1 \dots x^L}$: we have $\sum_{i \in [L]} x^i \cdot 2^{L-i} = x$. We will consider numbers that correspond to bit vectors consisting of $2|B| + 2$ sections, with each section consisting of

$$\left\lceil \log_2 \left(\sum_{a \in A} a + (|A| + 1) \sum_{b \in B} b + 1 \right) \right\rceil + \lceil \log_2(2|A| + 2) \rceil$$

bits; this value, which is polynomial in the size of the input, is chosen so that addition operations do not carry bits across sections. For $h = 1, \dots, 2|B| + 2$, let $\bar{x}(h)$ denote the h -th section of \bar{x} .

We now construct an instance of SSS^+ described by a triple $\langle Y, q, s \rangle$. Let $q = |A|$. The set Y consists of the following integers (see also Figure 1):

- For each $i = 1, \dots, q$, there is an integer x_i such that $\bar{x}_i(1) = \bar{1}$, $\bar{x}_i(2|B| + 2) = \bar{a}_i$, and $\bar{x}_i(h) = \bar{0}$ for each section $h \neq 1, 2|B| + 2$.
- There are q copies of integer x_0 such that $\bar{x}_0(1) = \bar{1}$, and $\bar{x}_0(h) = \bar{0}$ for each section $h \neq 1$.
- For each $i = 1, \dots, |B|$, there are:
 - $q + 1$ copies of integer y_i such that $\bar{y}_i(2i) = \bar{1}$, $\bar{y}_i(2|B| + 2) = \bar{b}_i$, and $\bar{y}_i(h) = \bar{0}$ for each section $h \neq 2i, 2|B| + 2$.
 - $q + 1$ copies of integer y'_i such that $\bar{y}'_i(2i) = \bar{1}$ and $\bar{y}'_i(h) = \bar{0}$ for every section $h \neq 2i$.
 - q copies of integer $y_{-i} = w_i - z_i$, where w_i is such that $\bar{w}_i(2i + 1) = \bar{1}$ and $\bar{w}_i(h) = \bar{0}$ for every $h \neq 2i + 1$, while z_i is such that $\bar{z}_i(2|B| + 2) = \bar{b}_i$ and $\bar{z}_i(h) = \bar{0}$ for every $h \neq 2|B| + 2$.

Also, we set the goal s so that $\bar{s}(1) = \bar{q}$, $\bar{s}(2|B| + 2) = \bar{t}$, and $\bar{s}(2h) = \overline{2q+1}$, $\bar{s}(2h+1) = \bar{q}$ for each $h \in \{1, \dots, |B|\}$.

To verify the correctness of the reduction, we first make the following observation. In the SSS^+ instance, the follower can achieve the goal s only if, for each $i = 1, \dots, |B|$, all copies of y_{-i} and exactly $2q + 1$ out of the $2q + 2$ copies of y_i and y'_i are included in the set $Y' \cup Y''$, which is chosen by the joint efforts of the leader and the follower: otherwise, the $2i$ -th and the $(2i + 1)$ -th sections of the sum would not match the corresponding sections in s . The follower can decide whether $Y' \cup Y''$ will contain $q + 1$ copy of y_i and q copies of y'_i or vice versa, since the leader's choice is restricted to q integers, while the follower's choice is unrestricted. Therefore, for each $i = 1, \dots, |B|$, the $(2|B| + 2)$ -th section of the sum of the selected copies of y_i , y'_i and y_{-i} will be either $\bar{0}$ or \bar{b}_i ; effectively, the follower chooses whether to include b_i in the sum.

Now, suppose that in the given BI-SS instance there exists a subset $A' \subseteq A$ such that for all $B' \subseteq B$ it holds that $\sum_{a \in A'} a + \sum_{b \in B'} b \neq t$. Then, in the corresponding instance of SSS^+ the leader can choose the subset Y' containing all x_i such that $a_i \in A'$ and $q - |A'|$ copies of x_0 . Given this choice of the leader, the follower can only choose integers from the copies of y_i , y'_i , and y_{-i} since any other choice will make the first section of the sum different from \bar{q} . However, since $\sum_{a \in A'} a + \sum_{b \in B'} b \neq t$ for all $B' \subseteq B$, no matter which integers the follower chooses, the last section of the sum cannot be \bar{t} . Thus, this instance of SSS^+ is a yes-instance.

Conversely, suppose that the BI-SS INSTANCE is such that for every $A' \subseteq A$ there exists a $B' \subseteq B$ such that $\sum_{a \in A'} a + \sum_{b \in B'} b = t$. We will now argue that in the corresponding instance of SSS^+ the follower can always achieve the goal s . Indeed, suppose the leader chooses a set Y' . Let $A' = \{a_i :$

$x_i \in Y'$, and, for each $i \in \{1, \dots, |B|\}$, let α_i be the number of copies of y_i in Y' , let α'_i be the number of copies of y'_i in Y' , and let β_i be the number of copies of y_{-i} in Y' . Fix some set $B' \subseteq B$ such that $\sum_{a \in A'} a + \sum_{b \in B'} b = t$. To achieve the goal s , the follower can include the following integers in Y'' :

- $q - |A'|$ copies of x_0 , so that the first section of the sum is exactly \bar{q} ;
- $q + 1 - \alpha_i$ copies of y_i , $q - \alpha'_i$ copies of y'_i , and $q - \beta_i$ copies of y_{-i} for each i such that $b_i \in B'$, so that the last sections of the copies of y_i, y'_i , and y_{-i} in $Y' \cup Y''$ sum up to \bar{b}_i ;
- $q - \alpha_i$ copies of y_i , $q + 1 - \alpha'_i$ copies of y'_i , and $q - \beta_i$ copies of y_{-i} for each i such that $b_i \notin B'$, so that the last sections of the copies of y_i, y'_i , and y_{-i} in $Y' \cup Y''$ sum up to $\bar{0}$.

This completes the proof. \square

We are now ready to show that SSS is Σ_2^P -complete. This problem is obviously in Σ_2^P . We show that it is Σ_2^P -hard via a reduction from SSS^+ . Given an instance $\langle Y, r, f \rangle$ of SSS^+ , we construct an instance $\langle X, \ell \rangle$ of SSS as follows. Let $q = |Y| - r$, $z = \sum_{x \in Y} x + 1$ and $z' = -f - (q + 1)z$.

- Let X consist of all integers in Y , $2q+1$ copies of z and $q+1$ copies of z' . Thus, $|X| = |Y| + 3q + 2$.
- Set $\ell = |Y| + 2q + 2$.

Observe that any subset $X' \subseteq X$ of size ℓ must contain: (1) at least r integers from Y , (2) at least one copy of z' , and (3) at least $q + 1$ copies of z . Note also that $z' + (q + 1)z = -f$. We will show that $\langle Y, r, f \rangle$ is a yes-instance of SSS^+ if and only if $\langle X, \ell \rangle$ is a yes-instance of SSS.

Suppose that $\langle X, \ell \rangle$ is a yes-instance of SSS; thus, there exists $X' \subseteq X$ such that $|X'| = \ell$ and $\sum_{x \in X''} x \neq 0$ for all $X'' \subseteq X'$ with $X'' \neq \emptyset$. By our observation, X' contains at least r integers from Y . Consider a set Y' obtained by picking exactly r elements from $X' \cap Y$. We claim that $\sum_{x \in Y''} x \neq f$ for all $Y'' \subseteq Y'$. Indeed, pick an arbitrary subset $Y'' \subseteq Y'$, and consider a set X'' containing all integers in Y'' , one copy of z' and $q + 1$ copies of z . If $\sum_{x \in Y''} x = f$, then $\sum_{x \in X''} x = 0$, contradicting our assumption that $\langle X, \ell \rangle$ is a yes-instance of SSS. Therefore, Y' witnesses that $\langle Y, r, f \rangle$ is a yes-instance of SSS^+ .

Conversely, suppose that $\langle Y, r, f \rangle$ is a yes-instance of SSS^+ , i.e., there exists a set $Y' \subseteq Y$ such that $|Y'| = r$ and $\sum_{x \in Y''} x \neq f$ for all $Y'' \subseteq Y'$. Consider the set X' containing all integers in Y' , all $q + 1$ copies of z' , and $q + 1$ copies of z ; hence, $|X'| = r + 2q + 2 = \ell$. Suppose towards a contradiction that $\sum_{x \in X''} x = 0$ for some $X'' \subseteq X'$ with $X'' \neq \emptyset$.

Let n_z and $n_{z'}$ be the number of copies of z and z' in X'' , respectively; we have $n_z, n_{z'} \leq q + 1$. We have

$$0 = \sum_{x \in X''} x = n_{z'} \cdot z' + n_z \cdot z + \sum_{x \in X'' \cap Y'} x. \quad (11)$$

Since z and all numbers in Y' are positive, it holds that $n_{z'} > 0$. Substituting $z' = -f - (q + 1)z$, we obtain

$$n_{z'} f + [(q + 1)n_{z'} - n_z]z = \sum_{x \in X'' \cap Y'} x \leq \sum_{x \in Y'} x < z$$

and hence

$$(q + 1)n_{z'} - n_z < 1. \quad (12)$$

Since $n_z, n_{z'} \leq q + 1$ and $n_{z'} > 0$, Eq. (12) implies that $n_{z'} = 1$ and $n_z = q + 1$. Substituting these values into Eq. (11), we obtain $\sum_{x \in X'' \cap Y'} x = f$, which contradicts the assumption that $\sum_{x \in Y''} x \neq f$ for all $Y'' \subseteq Y'$. We conclude that $\sum_{x \in X''} x \neq 0$ for all $X'' \subseteq X'$ with $X'' \neq \emptyset$, so $\langle X, \ell \rangle$ is a yes-instance of SSS. \square