# A Game-Based Definition of Coercion Resistance and its Applications*

Ralf Küsters † Tomasz Truderung Andreas Vogt

University of Trier, Germany
{kuesters,truderung,vogt}@uni-trier.de

February 13, 2013

**Abstract**

Coercion resistance is one of the most important and intricate security requirements for voting protocols. Several definitions of coercion resistance have been proposed in the literature, both in cryptographic settings and more abstract, symbolic models. However, unlike symbolic approaches, only very few voting protocols have been rigorously analyzed within the cryptographic setting. A major obstacle is that existing cryptographic definitions of coercion resistance tend to be complex and limited in scope: they are often tailored to specific classes of protocols or are too demanding.

In this paper, we therefore present a simple and intuitive cryptographic definition of coercion resistance, in the style of game-based definitions. This definition allows us to precisely measure the level of coercion resistance a protocol provides. As the main technical contribution of this paper, we apply our definition to two voting systems, namely, the Bingo voting system and ThreeBallot. The results we obtain are out of the scope of existing approaches. We show that the Bingo voting system provides the same level of coercion resistance as an ideal voting system. We also precisely measure the degradation of the level of coercion resistance of the ThreeBallot voting system when the so-called short ballot assumption is not met and show that the level of coercion resistance this system provides is significantly lower than that of an ideal system even in the case of short ballots.

**Keywords:** Coercion resistance, protocol analysis, electronic voting

*This is the full and extended version of work previously published in [18].

†Corresponding author:
ealf Küsters, University of Trier, FB IV - Informatik, Gebäude H/Campus II, D-54286 Trier, Germany
Tel. ++49 (0)651 201-2852
Fax. ++49 (0)651 201-2832
E-Mail: kuesters@uni-trier.de

## 1 Introduction

Coercion resistance is one of the most important and intricate security requirements for voting protocols [16, 24, 3]. Intuitively, a voting protocol is coercion-resistant if it prevents vote buying and voter coercion. Several definitions of coercion resistance have been proposed in the literature (see, e.g., [16, 22, 7, 27, 12, 15, 11, 21, 1]), both based on cryptographic and symbolic models, where symbolic models take an idealized view on cryptography. However, in the cryptographic setting, only very few voting protocols have been analyzed rigorously with respect to coercion resistance. A major obstacle is that existing definitions tend to be complex and limited in scope: they are often tailored to a very specific class of protocols or are too demanding; some otherwise reasonable protocols are deemed insecure or can be shown to be secure only under stronger assumptions or using stronger cryptographic primitives (see Section 3 for more details). Even some relatively simple voting protocols are out of the scope of current cryptographic approaches. The situation is much better for symbolic approaches. Several quite complex voting protocols have been analyzed in this setting (see, e.g., [11, 21, 1]). For example, in [21] coercion resistance of the Civitas voting system [9] was analyzed rigorously. However, in symbolic approaches, an idealized view on cryptography is taken and the level of coercion resistance a protocol provides cannot be measured precisely.

**Contribution of this paper.** We present a definition of coercion resistance in the cryptographic model, in the style of game-based definitions (rather than simulation-based definitions). The main idea is that a coercer should not be able to distinguish whether a coerced voter is following the instructions of the coerced voter (which, for example, could be to vote for the coercer's favorite candidate) or whether the coerced voter is just trying to achieve her own goal (which, for example, could be to vote for *her* favorite candidate), by running a counter-strategy.

The resulting cryptographic definition has the following main features compared to other cryptographic definitions (see Section 3 for a detailed comparison with existing definitions and approaches): i) Our definition is simple and intuitive. ii) It allows us to precisely measure the level of coercion resistance a protocol provides, i.e., the ability of the coercer to tell whether a coerced voter is following the instructions of the coercer or whether she is just following her own goal, where our definition provides a flexible way of specifying a coerced voter's goal. Typically, the level of coercion resistance depends on several parameters, including the number of voters and the number of choices voters have (e.g., the number of candidates in the election) as well as the probability distribution on these choices. This quantitative approach is much preferable over more coarse approaches that merely provide a yes/no answer, as illustrated by our case studies (see below). iii) Our definition is applicable to a wide range of protocols, including protocols that are out of the scope of existing approaches, with less stringent security assumptions and weaker cryptographic primitives than some of the other approaches, as demonstrated by our case studies.

We apply our definition to three different voting protocols, as explained below. Besides demonstrating the applicability of our definition, the results of our analysis are interesting in their own right as they constitute the first rigorous analyses of the considered voting systems and introduce techniques that are applicable beyond our case studies.

We first provide a detailed analysis of an ideal voting system that reveals only the totals of the election. The level of coercion resistance such a system provides is a function of the number of honest voters in an election, the number of choices (e.g., candidates) voters have in the election, and a probability distribution on choices, which describes how honest voters vote. The analysis of the ideal voting system is a pure combinatorial argument. This analysis is motivated by the discovery that the analysis of certain voting protocols, namely those that provide (almost) ideal coercion resistance, can often be divided into two parts: a combinatorial part corresponding to the ideal case and a cryptographic part. With the results presented in this paper, the combinatorial part does not have be redone.

Based on the analysis of the ideal voting system, we show that the Bingo voting system [4], which has been used in practice [2], provides the same level of coercion resistance as the ideal system (up to forced abstention attacks). This result is shown by a reduction to the ideal case, as explained above. It could not be obtained by previous approaches, as the Bingo voting system is either outside the considered class of voting systems or, in the case of simulation-based definitions, cannot be proven to be coercion resistance, unless stronger security assumptions or more advanced cryptographic primitives are used (see Section 3 and 5 for more details).

We also provide a detailed analysis of the ThreeBallot voting system [25]. This system is known to leak partial information to a coercer. In particular, it is known that coercion resistance cannot be obtained if the number of candidates in the election is high. In other words, coercion resistance can at most be achieved under the so-called *short ballot assumption*. However, this assumption has so far not been defined or quantified within a formal framework. Using our definition, we rigorously measure the degradation of coercion resistance as the number of candidates grows. Surprisingly, already with seven candidates and a few hundred voters the level of coercion resistance ThreeBallot provides is insufficient. With ten candidates and two thousand voters, ThreeBallot provides almost no coercion resistance. (Note that results of elections are often published per polling station and that a polling station often does not have more than a few hundred voters.) We also precisely analyze ThreeBallot in the case where the short ballot assumption is clearly met: we consider the case of two candidates. Even in this case, we find that the level of coercion resistance ThreeBallot provides is significantly less than the ideal protocol. This analysis of ThreeBallot requires non-trivial combinatorial arguments because information can be leaked in subtle ways. As in the case of the Bingo voting system, other approaches are unsuitable for the analysis of ThreeBallot (see Sections 3 and 6 for more details).

**Structure of this paper.** In the following section, we present and discuss our definition of coercion resistance. In Section 3 we provide a detailed comparison with other definitions and approaches. Our case studies are presented in Sections 4 to 6, with detailed proofs provided in the appendix. We conclude in Section 7.

## 2 Defining Coercion Resistance

In this section, we present our definition of coercion resistance. First, we introduce some notation and terminology.

### 2.1 Preliminaries

As usual, a function $f$ from the natural numbers to the real numbers is *negligible* if for every $c > 0$ there exists $\ell_0$ such that $f(\ell) \leq \frac{1}{\ell^c}$ for all $\ell > \ell_0$. The function $f$ is *overwhelming* if the function $1 - f(\ell)$ is negligible. Let $\delta \in [0,1]$. The function $f$ is *$\delta$-bounded* if $f$ is bounded by $\delta$ plus a negligible function, i.e., for every $c > 0$ there exists $\ell_0$ such that $f(\ell) \leq \delta + \frac{1}{\ell^c}$ for all $\ell > \ell_0$.

Our definition of coercion resistance is based on a quite standard computational model, similar to models for simulation-based security (see, e.g., [13, 17]), in which *interactive Turing machines (ITMs)* communicate via tapes. The ITMs may perform probabilistic polynomial-time computations in the length of the security parameter and the input received so far. The details of the model are not essential for the rest of the paper. For concreteness, we use the IITM model ("inexhaustible interactive Turing machine"), introduced in [17]. We now fix some notation. A *system* $\mathscr{S}$ of ITMs is a multi-set of ITMs, which we write as $\mathscr{S} = M_1 \parallel \cdots \parallel M_l$, where $M_1, \dots, M_l$ are ITMs. If $\mathscr{S}_1$ and $\mathscr{S}_2$ are systems of ITMs, then $\mathscr{S}_1 \parallel \mathscr{S}_2$ is a system of ITMs, assuming that the systems are connectible: Two systems are *connectible* if common external tapes, i.e., tapes with the same name, have complementary directions (input/output). As usual, the systems we consider are such that the length of a run is polynomially bounded in the length of the security parameter. Clearly, a run is uniquely determined by the random coins used by the ITMs in $\mathscr{S}$.

We assume that a system of ITMs has at most one ITM with a special output tape $\mathsf{decision}$. For a system $\mathscr{S}$ of ITMs and a security parameter $\ell$, we write $\Pr[S^{(\ell)} \mapsto 1]$ to denote the probability that $\mathscr{S}$ outputs 1 (on tape $\mathsf{decision}$) in a run with security parameter $\ell$.

A *property* of a system $\mathscr{S}$ is a subset of runs of $\mathscr{S}$. For a property $\gamma$ of $\mathscr{S}$, we write $\Pr[\mathscr{S}^{(\ell)} \mapsto \gamma]$ to denote the probability that a run of $\mathscr{S}$, with security parameter $\ell$, belongs to $\gamma$.

### 2.2 Voting Protocols

A *voting protocol* $P$ specifies the programs (actions) carried out by honest voters and honest voting authorities, such as honest registration tellers, tallying tellers, bulletin boards, etc.

A voting protocol $P$, together with certain parameters, induces an *election system* $S = P(k,m,n,\vec{p})$. The parameters are as follows: $k$ denotes the number of choices an honest voter has in the election apart from abstaining from voting. In the simplest case, these choices can be the candidates the voter can vote for. Choices can also be preference lists of candidates, etc. In what follows, we often use the terms "candidate" and "choice" interchangeably. By $m$ we denote the total number of voters and by $n$, with $n \leq m$, the number of honest voters. Honest voters follow the programs as specified in the protocol. The actions of dishonest voters and dishonest authorities are determined by the coercer, and hence, these participants can deviate from the protocol specification in arbitrary ways. We make the parameter $n$ explicit since it is crucial for the level of coercion resistance a system guarantees; intuitively the level of coercion resistance increases with the number of honest voters. One can also think of $n$ as the minimum number of voters the coercer may not corrupt. The vector $\vec{p} = p_0, \dots, p_k$ is a probability distribution on the possible choices, i.e., $p_0, \dots, p_k \in [0,1]$ and $\sum_{i=0}^{k} p_i = 1$. Honest voters will abstain from voting with probability $p_0$ and vote for candidate $i$ with probability $p_i$, $1 \leq i \leq k$. We make this distribution explicit, because it is realistic to assume that the coercer knows this distribution (e.g., from opinion polls) and may use it in his strategy, and because, as we see later, the specific distribution is crucial for the level of coercion resistance of a system.

An election system $S = P(k,m,n,\vec{p})$ specifies (sets of) ITMs for all participants, honest voters and authorities, the coercer, subsuming dishonest voters and dishonest authorities, and the coerced voter: (i) There are ITMs, say $S_1, \dots, S_l$, for all honest voting authorities. These ITMs run the programs as specified by the voting protocol. (ii) There is an ITM $S_{\mathsf{v}_i}$, $i \in \{1, \dots, n\}$ for each of the honest voters. Every such ITM first makes a choice according to the probability distribution $\vec{p}$. Then, if the choice is not to abstain, it runs the program for honest voters according to the protocol specification with the candidate chosen before. (iii) The coercer is described by a set $C_S$ of ITMs. This set contains all (probabilistic polynomial-time) ITMs, and hence, all possible coercion strategies the coercer can carry out. These ITMs are only constraint by their interface to the rest of the system. In particular, the ITMs can directly connect to the interface of dishonest voters and authorities. They can also communicate with the coerced voter. Moreover, they have access to all public information (e.g., bulletin boards) and possibly (certain parts of) the network. The precise interface of the ITMs in $C_S$ depends on the specific protocol and the assumptions on the power of the coercer. iv) Similarly, the coerced voter is described by a set $V_S$ of ITMs. Again, this set contains all (probabilistic polynomial-time) ITMs. This set represents all the possible programs the coercer can ask the coerced voter to run as well as all counter-strategies the coerced voter can run (see Section 2.3 for more explanation). The interface of these ITMs is typically the interface of an honest voter plus an interface for communication with the coercer. In particular, the set $V_S$ contains what we call a *dummy strategy* dum which simply forwards all the messages between the coercer and the interface the coerced voter has as an honest voter. We note that a program in $V_S$ can represent one coerced voter or a number of cooperating or independent coerced voters (see Section 2.3).

Given an election system $S = P(k,m,n,\vec{p})$, we denote by $\mathsf{e}_S$ the system of ITMs containing all honest participants, i.e., $\mathsf{e}_S = (S_{\mathsf{v}_1} \parallel \dots \parallel S_{\mathsf{v}_n} \parallel S_1 \parallel \dots \parallel S_l)$, where, as explained above, $S_{\mathsf{v}_1} \parallel \dots \parallel S_{\mathsf{v}_n}$ are the ITMs modeling honest voters and $S_1 \parallel \dots \parallel S_l$ are the honest authorities. A system $(c \parallel v \parallel \mathsf{e}_S)$ of ITMs, with $c \in C_S$ and $v \in V_S$, is called an *instance of S*. We often implicitly assume a scheduler (modeled as an ITM) to be part of a system. Its role is to make sure that all components of the system are scheduled in a fair way, e.g., all voters get a chance to vote. For simplicity of notation, we do not state the scheduler explicitly. We define a *run of S* to be a run of some instance of $S$.

For an election system $S = P(k,m,n,\vec{p})$, we denote by $\Omega_1 = \{0, \dots, k\}^n$ the set of all possible combinations of choices made by the honest voters, with the corresponding probability distribution $\mu_1$ derived from $\vec{p} = p_0, p_1, \ldots, p_k$. All other random bits used by ITMs in an instance of $S$, that is, all other random bits used by honest voters as well as all random bits used by honest authorities, the coercer, and the coerced voter, are uniformly distributed. We take $\mu_2$ to be this distribution over the space $\Omega_2$ of random bits. Formally, this distribution depends on the security parameter. We can, however, safely ignore it in the notation without causing confusion. We define $\Omega = \Omega_1 \times \Omega_2$ and $\mu = \mu_1 \times \mu_2$, i.e., $\mu$ is the product distribution obtained from $\mu_1$ and $\mu_2$. For an event $\varphi$, we write $\mathsf{Pr}_{\omega_1,\omega_2 \leftarrow \Omega}[\varphi]$, $\mathsf{Pr}_{\omega_1,\omega_2}[\varphi]$, or simply $\mathsf{Pr}[\varphi]$ to denote the probability $\mu(\{(\omega_1,\omega_2) \in \Omega : \varphi(\omega_1,\omega_2)\})$. Similarly, $\mathsf{Pr}_{\omega_1 \leftarrow \Omega_1}[\varphi]$ or simply $\mathsf{Pr}_{\omega_1}[\varphi]$ stand for $\mu_1(\{\omega_1 \in \Omega_1 : \varphi(\omega_1)\})$; analogously for $\mathsf{Pr}_{\omega_2 \leftarrow \Omega_2}[\varphi]$.

A *property* of an election system $S = P(k,m,n,\vec{p})$ is defined to be a class $\gamma$ of properties of instances of $S$, containing one property $\gamma_T$ for each instance $T$ of $S$. We write $\mathsf{Pr}\,[T \mapsto \gamma]$ to denote the probability $\mathsf{Pr}\,[T \mapsto \gamma_T]$.

### 2.3 Coercion Resistance

We can now present our definition of coercion resistance. For now, we concentrate on the case that only a single voter is coerced. The case of multi-voter coercion resistance is discussed later. In what follows, let $P$ be a voting protocol and $S = P(k,m,n,\vec{p})$ be an election system for $P$.

In the definition of coercion resistance we imagine that the coercer demands full control over the voting interface of the coerced voter: the coercer wants the coerced voter to forward all messages from/to the interface of the coerced voter. In other words, the coercer wants the coerced voter to run the dummy strategy $\mathsf{dum}$ (see Section 2.2) instead of the program an honest voter would run. If the coerced voter indeed runs $\mathsf{dum}$, the coercer can effectively vote on behalf of the coerced voter or decide to abstain from voting. Of course, the coercer is not bound to follow the specified voting procedure; he can perform arbitrary coercion strategies. For example, the coercer could send fake messages and depend his decisions on the information he has gathered so far. The intention of the coercer might even be to merely test whether the coerced voter follows his instructions, e.g., to find out whether this voter is "reliable", and hence, is a good candidate for coercion in later elections. Also, the coercer is not necessarily bound to use the interface of the coerced voter in his coercion strategy. There may be other ways to vote on behalf of the coerced voter. (However, for a protocol to be coercion-resistant, there will always be at least one step in the protocol that the coercer cannot do all by himself, e.g., register, perform operations on a security token, or vote in a voting booth. For such actions, the coercer has to consult the coerced voter.)

Our definition of coercion resistance is parameterized by a set $\tilde{V} \subseteq V_S$, which we call a set of *goal achieving (counter-)strategies*. The intuition is that the set $\tilde{V}$ is defined in such a way that by running a program in $\tilde{V}$ (instead of running $\mathsf{dum}$), called a *counter-strategy*, the coerced voter achieves her own goal, e.g., votes for her favorite candidate, despite of what the coercer tells her to do. The concrete definition of $\tilde{V}$ depends on the specific goals one wants the coerced voter to be able to achieve, which in turn can depend on the protocol and the security assumptions (e.g., the power of the coercer). We therefore do not fix this set up front. However, below and in our case studies we provide several examples of how $\tilde{V}$ can be defined.

Now, for a protocol to be coercion-resistant our definition requires that there exists a *counter-strategy* $\tilde{v} \in \tilde{V}$ that the coerced voter can run instead of $\mathsf{dum}$ such that the coercer is not able to distinguish whether the coerced voter runs $\mathsf{dum}$ or $\tilde{v}$. More precisely, we will measure the ability of the coercer to distinguish between these two cases. Note that, since $\tilde{v} \in \tilde{V}$, by running $\tilde{v}$ the coerced voter achieves her own goal. So, in a nutshell, our definition demands that there exists a counter-strategy such that the coercer cannot tell (or can tell with only small probability) whether the coerced voter is in fact following the coercer's instructions or whether she is just running the counter-strategy, and hence, achieves her own goal. If such a counter-strategy exists, then it indeed does not make sense for the coercer to try to influence a voter in any way, e.g., by offering money and/or threatening the voter, at least not from a technical point of view:[1] Even if the coerced voter tries to sell her vote (and hence, follows the instructions of the coercer by running $\mathsf{dum}$), the coercer is not able to tell whether she was actually following the coercer's instructions (i.e., run $\mathsf{dum}$) or whether she was just trying to achieve her own goal (i.e., run the counter-strategy). For the same reason, the coerced voter is safe even if she wants to achieve her goal and therefore runs the counter-strategy.

Our formal definition of coercion resistance is the following:

**Definition 1.** *Let $P$ be a protocol and $S = P(k,m,n,\vec{p})$ be an election system. Let $\delta \in [0,1]$, and $\tilde{V} \subseteq V_S$. The system $S$ is $\delta$-*coercion-resistant w.r.t. $\tilde{V}$*, if there exists $\tilde{v} \in \tilde{V}$ such that for all $c \in C_S$ we have*

$$\mathsf{Pr}[(c \parallel \mathsf{dum} \parallel \mathsf{e}_S)^{(\ell)} \mapsto 1] - \mathsf{Pr}[(c \parallel \tilde{v} \parallel \mathsf{e}_S)^{(\ell)} \mapsto 1] \text{ is } \delta\text{-bounded, as a function of the security parameter } \ell. \quad (1)$$

Following the intuition explained above, since $\tilde{v} \in \tilde{V}$, by running $\tilde{v}$ the coerced voter achieves her own goal in runs of $c \parallel \tilde{v} \parallel \mathsf{e}_S$. Condition (1) captures that the coercer is unable to distinguish whether the coerced voter ran $\mathsf{dum}$ or $\tilde{v}$. More precisely, the coercer accepts a run (i.e., outputs 1 on tape $\mathsf{decision}$) with almost the same probability no matter whether the coerced voter performs $\mathsf{dum}$ or $\tilde{v}$, where "almost the same" is formalized as "$\delta$-bounded", for some reasonably small $\delta$ (see below for more explanation). If the two probabilities are far apart, say for example, for some $c$, the probability of $c$ accepting the run is 60% higher if the coerced voter performs $\mathsf{dum}$, this would give strong incentive for the coerced voter to follow the instructions of the coercer, i.e., run $\mathsf{dum}$: Imagine, for example, that if the coercer outputs 1 (i.e., thinks the coerced voter is following his instructions), then he pays $50 to the coerced voter and if he outputs 0 (i.e., thinks the coerced voter is not following his instructions) forces the coerced voter to pay $20 or damnifies the coerced voter in some other way. Now, by running $\mathsf{dum}$, the coerced voter significantly increases her chance of getting paid and her chance of not being punished.

In the rest of this section, we discuss further aspects and features of the definition.

**Negligible vs. $\delta$-bounded.** The reader might wonder why we require the difference in (1) to be $\delta$-bounded, rather than negligible. The reason is that negligibility is too

[1] Of course, voters can be influenced psychologically.

strong. The difference in (1), even for an ideal protocol, which merely reveals the result of the election, does not decrease with an increasing security parameter, but may depend on the number of choices, the distribution $\vec{p}$ on these choices, and the number of honest voters: Imagine, for example, that a candidate did not get any vote in an election. Now, if the coercer asked the coerced voter to vote for this candidate, it is clear that the coerced voter did not follow the coercer's instruction. The probability for this to happen is non-negligible and depends on $\vec{p}$ and the number of honest voters; the larger the number of honest voters is, the more likely it is that a candidate gets a vote. In fact, in our examples (see Section 5 and 6), $\delta$ will depend on the number of candidates, the probability distribution $\vec{p}$, and the number of honest voters. As will be illustrated in our case studies, the bound $\delta$ indeed provides for a precise level of coercion resistance and is of practical relevance: it might, for example, indicate that a voting protocol does not have a sufficient level of coercion resistance if the number of voters is below a certain threshold or the number of candidates is too big.

**Typical voter goals.** The set $\tilde{V}$ of goal achieving strategies provides for great flexibility in defining the goal of the coerced voter. As we discuss later in this section, this flexibility is very convenient and often necessary in order to specify the exact kind of coercion resistance protocols provide. However, we now point out two important and particularly desirable goals, which we call *favorite candidate goal* and *favorite candidate up to forced abstention goal*:

**Favorite candidate:** The favorite candidate goal requires that, when following her counter-strategy, the coerced voter always successfully votes for the candidate of her choice, where "successfully" means that the coerced voter's vote is in fact counted.

We refer to coercion resistance with respect to this goal as **strong coercion resistance**.

**Favorite candidate up to forced abstention:** This goal requires that the coerced voter successfully votes for her favorite candidate if instructed by the coercer to vote at all (perhaps for some other candidate). If the coerced voter is not instructed to vote, the coerced voter may abstain from voting. In other words, forced abstention attacks, where the coercer can somehow force a voter not to vote, are tolerated.

We refer to coercion resistance with respect to this goal as **coercion resistance up to forced abstention**.

In the following paragraph (trace-based voter goals), we will provide a more detailed definition of these goals.

Election systems should be designed in such a way that strong coercion resistance is achieved. If it is impossible to prevent forced abstention attacks (for example, because a coercer can observe who enters a polling station), then at least coercion resistance up to forced abstention, as defined above, should be achieved by a voting system. In the case studies presented in this paper, we always use the latter notion, which is the best the systems in our case studies accomplish.

Voting systems that provide less than strong coercion resistance or coercion resistance up to forced abstention are undesirable (see, however, the remark in example (c) in the following paragraph). Nevertheless, a definition of coercion resistance should not completely rule out such voting systems but should allow for precisely measuring the level of coercion resistance of such systems. This in fact is the reason why our definition of coercion resistance is parameterized by the set $\tilde{V}$ of goal achieving strategies: with $\tilde{V}$ we can precisely capture the set of goals achievable by a coerced voter. (Similarly, with $\delta$ we can precisely measure how well a coercer can tell whether a coerced voter followed his instructions or not.)

**Defining trace-based voter goals.** We now define a specific class of goal achieving strategies, which in particular, includes the two goals sketched above. Let $\gamma$ be a property of an election system $S = P(k,m,n,\vec{p})$ (recall the definition of properties at the end of Section 2.2). For example, if the goal of the coerced voter is to vote for some specific candidate, then $\gamma$ would contain all the runs in which the coerced voter successfully voted for this candidate. Given $\gamma$, the corresponding set of goal achieving strategies is defined as follows: *The set $\tilde{V}(\gamma)$ of goal achieving strategies corresponding to $\gamma$* is defined by:

$$\tilde{V}(\gamma) = \{v \in V_S \mid \text{ for all } c \in C_S\text{: } \mathsf{Pr}\left[(c \parallel v \parallel \mathsf{e}_S)^{(\ell)} \mapsto \gamma\right] \text{ is overwhelming as a function of } \ell\}. \quad (2)$$

That is, the set $\tilde{V}(\gamma)$ contains all counter-strategies such that the goal $\gamma$ is achieved with overwhelming probability independent of the behavior of the coercer. Let us give some examples of concrete goals that can be expressed in this way, starting with the two most important and most desirable goals sketched in the previous paragraph.

(a) *Favorite candidate*: To express this goal, $\gamma$ contains exactly those runs in which the coerced voter indeed successfully voted for her favorite candidate.

(b) *Favorite candidate up to forced abstention*: To express this goal, $\gamma$ contains exactly those runs in which the coerced voter indeed successfully voted for her favorite candidate if the coercer instructed her to vote at all.

(c) If ballots are sent over an unreliable channel, the goals just described are typically too strong. In this case, we can use a *weaker* voter goal $\gamma$ that contains all those runs in which the coerced voter successfully voted for her favorite candidate if her ballot was delivered (and the coercer instructed her to vote).

(d) Another more complex example is the following: As already explained, in elections where the probability for one candidate, say $A$, to get a vote is very low, the level of coercion resistance can be quite low, i.e., $\delta$ can be quite large, because the coercer can instruct the coerced voter to vote for $A$. Even in an ideal voting protocol, the coerced voter has not much choice in such a situation than to vote for $A$. However, if there are two other candidates, $B$ and $C$, say, with reasonably high probabilities, and the goal of the coerced voter is to vote for $C$, then $\gamma$ could be defined as follows: If the coercer asks the coerced voter to vote for $B$ (and the coerced voter can tell that this is the case), then the coerced voter votes for $C$. For such a (weakened) goal, $\delta$ would be smaller, saying that the level of coercion resistance, as far as $\delta$ is concerned, is high if the coercer wants the coerced voter to vote for a candidate with high probability and the favorite candidate of the coerced voter has reasonably high probability as well.

**Beyond trace properties.** With $\gamma$ all goals can be expressed that are properties of single traces (*trace properties*). However, some goals that cannot be expressed as trace properties may be relevant in some applications. For example, one might require that by running the counter-strategy the coerced voter not only votes for her favorite candidate but also does not reveal (any partial information about) her private key to the coercer. This voter goal, which is stronger than the two standard goals introduced above, can be expressed in the usual way as an indistinguishability game, but not as a trace property. In future work, we plan to analyze Civitas [9], a voting system that involves a public key infrastructure, with respect to this kind of voter goal.

**Family of voter goals.** Definition 1 is formulated w.r.t. a single set of goal achieving strategies $\tilde{V}$. This can easily be generalized to a family of sets of goal achieving strategies: a protocol is coercion-resistant for such a family if it is coercion-resistant for all members of this family. For example, if $c$ denotes a candidate or the choice of abstaining from voting and $\tilde{V}_c$ describe the goal that by running $\tilde{v} \in \tilde{V}_c$ the coerced voter is guaranteed to successfully vote according to $c$, then the family of goals could contain $\tilde{V}_c$ for all choices $c$.

**Coercion strategies.** In Definition 1, we assume that the coercer wants the coerced voter to run the dummy strategy $\mathsf{dum}$. Alternatively, one could assume that the coercer wants the coerced voter to run some arbitrary coercion strategy $v \in V_S$. Then, one would demand that for every coercion strategy $v \in V_S$, there exists a counter-strategy $v' \in \tilde{V}$ such that (1) is satisfied (with $\mathsf{dum}$ replaced by $v$ and $\tilde{v}$ replaced by $v'$). However, given some closure property of $\tilde{V}$ (detailed below), it is easy to see that this formulation of coercion resistance is not stronger than Definition 1. Intuitively, the reason is that the coercer can run $v$ himself. For concreteness, let us assume that $\tilde{V} = \tilde{V}(\gamma)$ for some property $\gamma$ as in (2). This guarantees a sufficient closure property of $\tilde{V}$, but the following also holds in more general cases. Now, if there exists a counter-strategy $\tilde{v} \in \tilde{V}$ for $\mathsf{dum}$, then it is easy to define a counter-strategy $v'$ for a coercion strategy $v$, namely $v' = (v \parallel \tilde{v})$. It is easy to see that $(v \parallel \tilde{v}) \in \tilde{V}$, since for every $c \in C_S$, the system $(c \parallel (v \parallel \tilde{v}) \parallel \mathsf{e}_S)$ behaves exactly as the system $((c \parallel v) \parallel \tilde{v} \parallel \mathsf{e}_S)$ and $(c \parallel v)$ can be seen as a coercer $c' \in C_S$. By definition of $\tilde{v}$, we know that $\mathsf{Pr}[(c' \parallel \tilde{v} \parallel \mathsf{e}_S)^{(\ell)} \mapsto \gamma]$ is overwhelming, as a function of $\ell$. (More generally, the key closure property for $\tilde{V}$ we need is that $(v \parallel \tilde{v}) \in \tilde{V}$.) Condition (1) is satisfied following a similar reasoning: the system $S_1 = (c \parallel v \parallel \mathsf{e}_S)$ behaves exactly as $S_1' = ((c \parallel v) \parallel \mathsf{dum} \parallel \mathsf{e}_S)$, since $\mathsf{dum}$ merely forwards messages. Moreover, the system $S_2 = (c \parallel (v \parallel \tilde{v}) \parallel \mathsf{e}_S)$ is equivalent to $S_2' = ((c \parallel v) \parallel \tilde{v} \parallel \mathsf{e}_S)$. Now, as above, $(c \parallel v)$ can be considered to be a coercer $c' \in C_S$ and by definition of $\tilde{v}$, we know that $\mathsf{Pr}[S_1' \mapsto 1] - \mathsf{Pr}[S_2' \mapsto 1]$ is $\delta$-bounded, as a function of the security parameter $k$, and hence, this is true for $\mathsf{Pr}[S_1 \mapsto 1] - \mathsf{Pr}[S_2 \mapsto 1]$.

We use Definition 1 since it simplifies proofs. Also, $\tilde{v}$ is the strongest counter-strategy in that it can be used to construct counter-strategies for all coercion strategies (as shown above). Therefore, when designing a voting protocol, we believe that the counter-strategy $\tilde{v}$ should in fact be part of the protocol specification.

**Multi-voter coercion.** So far, we have focused on the case where only one voter is coerced. In reality a coercer can coerce many voters. From the point of view of a single coerced voter – say, Alice – the behavior of other coerced voters may deviate in arbitrary ways from the prescribed protocol. Alice should be able to resist coercion, independently of the other coerced voters, whom Alice might not know anyway, and independently of their behavior. However, this is already captured by Definition 1 since other coerced voters can simply be considered to be dishonest voters, and hence, they are subsumed by the coercer. This makes the coercer only more powerful, since now he even fully dictates the behavior of other coerced voters in the coercion of Alice.

Conversely, coerced voters might want to team up, e.g., to have better chances to sell their votes. This can also be modeled using Definition 1 since dum and $\tilde{v}$ can represent a set of coerced voters. So, dum would be a parallel composition of single dummy strategies, one for every coerced voter, and $\tilde{v}$ could be either a joint counter-strategy or a parallel composition of independent counter-strategies.

## 3 Comparison with Other Definitions

One of the first rigorous definitions of coercion resistance was presented by Juels et al. [16]. They defined coercion resistance relative to an ideal system. However, their definition is tailored towards voting in a public-key setting, with protocols having a specific structure. In particular, neither the Bingo voting system nor ThreeBallot fall into the class of protocols considered by Juels et al. Conversely, the voting protocol proposed by Juels et al., and also the Civitas system [9] which generalizes the protocol by Juels et al., falls in our framework.

A rather general definition of coercion resistance within the simulation-based approach was presented by Moran and Naor [22], based on a definition of coercion resistance for multi-party computation by Canetti and Gennaro [5]. In this approach, a protocol is considered to be coercion-resistant if it realizes an ideal voting functionality. The advantage of such definitions, compared to game-based definitions considered here, is that they provide composability by construction. However, this comes with a price: many reasonable voting protocols cannot be proven secure due to the so-called commitment problem. This is, for example, the case for the Bingo voting system (see Section 5 for details). Other protocols are equipped with more advanced cryptographic primitives only in order for the security proofs in the simulation-based setting to go through (see, e.g., the split-ballot protocol [23]). Even if the commitment problem does not occur, the simulation-based definition is often too strong: it gives a yes/no answer—the difference between the ideal and real system is negligible or not—instead of measuring the level of coercion resistance (as we do in our definition). Indeed for many protocols, such as paper-based protocols, the difference between a real and ideal system is non-negligible, but still reasonably small: For example, in some paper-based protocols there is a certain probability that a single fake ballot can be produced without being noticed (since, for example, only partial auditing is done). If a coerced voter gets such a fake ballot, her vote might be revealed. The probability that a fixed coerced voter gets the fake ballot might be small (but non-negligible), e.g., approximately $\frac{1}{n}$, where $n$ is the number of voters. Hence, the coercion level is increased by $\frac{1}{n}$, i.e., in our definition, $\delta$ is increased by $\frac{1}{n}$. This might be considered small for practical purposes, but is not captured by a yes/no answer as given in the simulation-based definition. In the simulation-based definition, one could replace negligibility by $\delta$-boundedness. Unfortunately, even in situations like the above, in simulation-based definitions, $\delta$ needs to be quite big for reasons unrelated to the actual degradation of the level of coercion resistance: the environment knows how honest voters vote, and hence, it can tell with high probability whether it deals with the real or ideal system. So, replacing "negligible" by "$\delta$-boundedness" in the simulation-based definition often does not yield satisfactory results. In [23], Moran and Naor proposed and analyzed the paper-based voting protocol split-ballot which, in fact, is not perfect due to fake ballots. In this work, they indeed do not opt for $\delta$-boundedness or the like, but change the ideal functionality. This approach can be problematic since it might not be clear whether the resulting functionality can be considered ideal. In particular, in their "ideal" functionality, they allow the adversary to *retroactively* change the votes of corrupted voters as *a function of the tally*, where the difference to the original tally is bounded only by the security parameter. For other "imperfect" protocols, such as ThreeBallot (see Section 6), finding a reasonable ideal functionality which is not too close to the protocol itself can be very challenging.

In [28], Unruh and Müller-Quade generalize the simulation-based framework of [22] and [5] for coercion resistance. Independently of our work, this paper presents a game-based definition of coercion resistance which is similar to our definition; both game-based definitions are conceptually close to the symbolic definition proposed in [21]. However, the definition in [28] is not further applied, except as a means to illustrate the simulation-based framework. In fact, Unruh and Müller-Quade cannot apply their framework to most published voting protocols: As for their simulation-based definition the reason is as described for the definition by Moran and Naor. Their game-based definition misses two important parameters, which are crucial in the analysis of practical voting protocols: (i) While we have a parameter $\tilde{V}$ for the goal of a coerced voter, they fix a specific goal, requiring that the coerced voter has to vote for a specific candidate. As explained in Section 2.3, such a goal is too strong for most practical protocols (e.g., in presence of network delays or observable misbehavior). (ii) While we have a parameter $\delta$ for specifying the level of coercion resistance, they fix $\delta$ to be the level of coercion resistance an ideal protocol guarantees, plus a negligible function. As argued before, many reasonable protocols, such as some paper-based protocols, do not achieve this level of coercion resistance.

Teague et al. [27] proposed a definition of coercion resistance which takes a quantitative approach. However, this definition has the following limitations: (i) It is intended to be used for ideal protocols, combined, as the authors suggest, with a simulation-based definition. (ii) The coercer may only use a specific strategy to decide whether to punish the coerced voter or not. Also, the class of counter-strategies available to the coerced voter is limited. (iii) Only the probability that a cheating voter gets punished is considered, ignoring the possibility that a voter might try to sell her vote by following the instructions of the coercer.

A recent definition of coercion resistance by Gardner et al. [12] is specifically tailored to the protocol considered by the authors. It also considers only a very restricted part of an election process, denying, for example, the coercer access to information in the tallying phase. In particular, the Bingo voting system and ThreeBallot are not in the scope of this definition.

The present paper is a full and extended version of the paper [18]. We note that in [18] our definition of coercion resistance has goal achieving strategies corresponding to a property $\gamma$ (see (2)) explicitly built into the definition. In contrast, here we chose a more general approach by allowing for arbitrary sets of goal achieving strategies, which, as explained, allows us to express goals beyond trace properties.

As already mentioned in the introduction, several definitions of coercion resistance were proposed in symbolic models (see, for example, [11, 1, 21]). Our game-based definition is inspired by the definition in [21].

## 4 Analyzing the Ideal Protocol

In this section, we analyze an ideal voting protocol and precisely establish the level of coercion resistance this protocol provides: we determine the optimal (i.e., minimal) constant $\delta_{min}$ for which the ideal protocol is coercion-resistant. In particular, no real protocol can be $\delta$-coercion-resistant for any $\delta < \delta_{min}$. As already explained in the introduction, the results of this section are motivated by the fact that the analysis of real voting protocols can often be reduced to the ideal case (see Section 5 for an example).

We consider here the most common tallying function. It returns the number of votes each candidate gets. This kind of tallying function is used in the protocols that we analyze in the following sections. We note that the level of coercion resistance depends on the tallying functions used in an election.

**The ideal protocol.** In the ideal protocol, denoted by $\mathsf{P}_{\mathsf{ideal}}$, a voter sends her choice directly to the fully trusted election process. The election process properly counts the votes and outputs the result, without revealing any additional information. Here we consider a result to be a $(k+1)$-tuple indicating the number of abstaining voters and the number of votes each of the $k$ candidates got.

More precisely, let $S = \mathsf{P}_{\mathsf{ideal}}(k,m,n,\vec{p})$ denote the election system defined as follows. The system $S$ contains exactly one voting authority. The program of an honest voter randomly picks a choice according to the distribution $\vec{p}$ and either abstains from voting or, otherwise, sends the chosen candidate on some untappable channel to the voting authority. In particular, unless a voter is dishonest, only the voter and the voting authority know whether the voter abstained from voting and, if the voter did not abstain, the chosen candidate. The program of the voting authority simply collects the votes received on the untappable channels from the voters (one vote for each voter) and then outputs the result of the election.

The coercer completely controls the dishonest voters and can also send messages to the coerced voter. In fact, by definition of the ideal protocol, the only reasonable message the coercer can send to the coerced voter and on the untappable channels of the dishonest voters are the desired candidates; everything else would be ignored by the voting authority. Since the protocol does not output messages to voters, the coercer does not expect to receive messages. Hence, the view of the coercer merely consists of his own random coins and the result of the election.

A coerced voter, running the dummy strategy or emulating it by running the counter-strategy, can receive a message from the coercer and send her choice on the untappable channel to the voting authority.

**Goals of the coerced voter.** The set of goal achieving strategies is defined according to (2). We will consider goals $\gamma_i$ of the coerced voter, for $i \in \{1,\ldots,k\}$, defined as follows: $\gamma_i$ is satisfied in a run if whenever the coerced voter has sent her candidate to the voting authority, she has successfully voted for the $i$-th candidate. This implies that if the coerced voter is not instructed by the coercer to vote, i.e., the coercer does not send his candidate to the coerced voter, and hence, effectively wants the coerced voter to abstain from voting, the coerced voter does not have to vote in order to fulfill $\gamma_i$. In other words, $\gamma_i$ corresponds to the goal *favorite candidate up to forced abstention* defined in Section 2.3, and hence, by $\gamma_i$ forced-abstention attacks are not prevented. We define $\tilde{V}_i = \tilde{V}(\gamma_i)$ (recall (2)).

Alternatively, we could consider a stronger and simpler goal $\gamma'_i$ which requires the coerced voter to vote for $i$, even if the coercer wants the coerced voter to abstain; this goal corresponds to what we called *favorite candidate goal* in Section 2.3. In fact, for this goal we obtain very similar results. However, $\gamma'_i$ is too strong for most practical protocols, including the ones we consider in this paper. For reasons of uniformity, we therefore restrict ourselves to the goal $\gamma_i$.

We also note that, for the ideal protocol, we could consider abstention to be a goal of the coerced voter. But again, this goal cannot be achieved in most practical protocols in which a voter is given a receipt, as such receipts can be used by the coercer to verify that the voter has actually voted.

**The optimal constant $\delta_{min}$.** We now establish the optimal constant $\delta_{min}$ mentioned above. As this constant depends on the number of candidates, on the number of honest voters, and the probability distribution $\vec{p}$, we will will denote it by $\delta_{min}(k,n,\vec{p})$.

For this purpose, we consider the following counter-strategy $\tilde{v}$ of the coerced voter: if the coerced voter receives a candidate from the coercer, then the coerced voter indicates to the voting authority that she wants to vote for the $i$-th candidate; otherwise, she abstains from voting. Clearly, this counter-strategy guarantees that $\gamma_i$ is met, that is, $\tilde{v} \in \tilde{V}_i$.

To determine $\delta_{min}$, we need some terminology and notation. Since the coercer knows the votes of dishonest voters, he can simply subtract these votes from the final result and obtain what we will call the *pure result* of the election. The pure result only depends on the votes of the $n$ honest voters and the coerced voter. Hence, a pure result is a tuple $\vec{r} = (r_0,\ldots,r_k)$ of non-negative integers such that $r_0 + \cdots + r_k = n+1$, where $r_i$, for $i \in \{1,\ldots,k\}$, is the number of votes from honest voters and the coerced voter for the $i$-th candidate and $r_0$ denotes the number of honest voters, including the coerced voter, who abstained from voting. As already mentioned above, the coercer has to base his decision—accept or reject—solely on such a pure result $\vec{r}$. We denote the set of pure results by *Res*.

In the definition of $\delta_{min}(k,n,\vec{p})$, we will use the probability $A^i_{\vec{r}}$ that the choices made by the honest voters and the coerced voter yield the pure result $\vec{r} = (r_0,\ldots,r_k)$, given that the coerced voter votes for the $i$-th candidate. Since $\vec{r}$ follows a multinomial distribution with parameters $n$ and $\vec{p}$, we have:

$$\begin{aligned} A^i_{\vec{r}} &= \binom{n}{r_0,\ldots,r_{i-1},r_i-1,r_{i+1},\ldots,r_k} \cdot p_0^{r_0} \ldots p_{i-1}^{r_{i-1}} p_i^{r_i-1} p_{i+1}^{r_{i+1}} \ldots p_k^{r_k} \\ &= \frac{n!}{r_0! \cdots r_k!} \cdot p_0^{r_0} \ldots p_k^{r_k} \cdot \frac{r_i}{p_i}, \end{aligned}$$

where $\binom{n}{m_0,\ldots,m_k} = \frac{n!}{m_0! \cdot \ldots \cdot m_k!}$.

The intuition behind the definition of $\delta_{min}(k,n,\vec{p})$ is the following: if the coercer wants the coerced voter to vote for $j$ and the coerced voter wants to vote for $i$, for some $i, j \in \{1,\ldots,k\}$, then, as we will show, the best strategy of the coercer to distinguish whether the coerced voter has voted for $j$ or $i$ is to accept a run if the pure result $\vec{r}$ of the election in this run is such that $A^i_{\vec{r}} \leq A^j_{\vec{r}}$. Let $M^*_{i,j} = \{\vec{r} \in Res : A^i_{\vec{r}} \leq A^j_{\vec{r}}\}$ be the set of those results, for which—according to his best strategy—the coercer should accept the run.

The following lemma yields a convenient and intuitive characterization of the set $M^*_{i,j}$. It says that a result should be accepted by the coercer iff the actual ratio $\frac{r_j}{r_i}$ of the number of votes for $j$ to the number of votes for $i$ is bigger than the expected ratio $\frac{p_j}{p_i}$.

**Lemma 1.** *$A^i_{\vec{r}} \leq A^j_{\vec{r}}$ iff $\frac{r_j}{r_i} \geq \frac{p_j}{p_i}$, and therefore $M^*_{ij} = \{\vec{r} \in Res : \frac{r_j}{r_i} \geq \frac{p_j}{p_i}\}$.*

*Proof.* We have the following equation:

$$A^j_{\vec{r}} - A^i_{\vec{r}} = \frac{n!}{r_0! \cdots r_k!} \cdot p_0^{r_0} \cdots p_k^{r_k} \cdot \left( \frac{r_j}{p_j} - \frac{r_i}{p_i} \right).$$

This term is $\geq 0$ if and only if $\frac{r_j}{r_i} \geq \frac{p_j}{p_i}$. □

Now, we are ready to define the constant $\delta^i_{min}$, which we will show to be optimal:

$$\delta^i_{min}(n,k,\vec{p}) = \max_{j \in \{1,\ldots,k\}} \sum_{\vec{r} \in M^*_{i,j}} (A^j_{\vec{r}} - A^i_{\vec{r}}).$$

In the definition of this constant, we take into account all possible candidates $1,\ldots,k$ that the coercer can wish the coerced voter to vote for, excluding abstention, as in this case the counter-strategy coincides with the dummy strategy. (Recall that our definition of $\gamma$ does not prevent forced-abstention attacks.) We take the worst possible case, i.e., the index $j$ for which the sum in the expression above is maximal.

The following theorem shows that $\delta^i_{min}$ is indeed optimal. The proof of this theorem is presented in Appendix A.

**Theorem 1.** *Let $S = \mathsf{P}_{\mathsf{ideal}}(k,m,n,\vec{p})$. Then $S$ is $\delta$-coercion-resistant with respect to $\tilde{V}_i$, where $\delta = \delta^i_{min}(n,k,\vec{p})$. Moreover, $S$ is not $\delta'$-coercion-resistant for any $\delta' < \delta$.*

In Figure 1, we depict values of $\delta = \delta^i_{min}$ for some selected cases. These values illustrate that the level of coercion resistance heavily depends on the number of honest voters, the number of candidates, and the probability distribution on the choices. Note

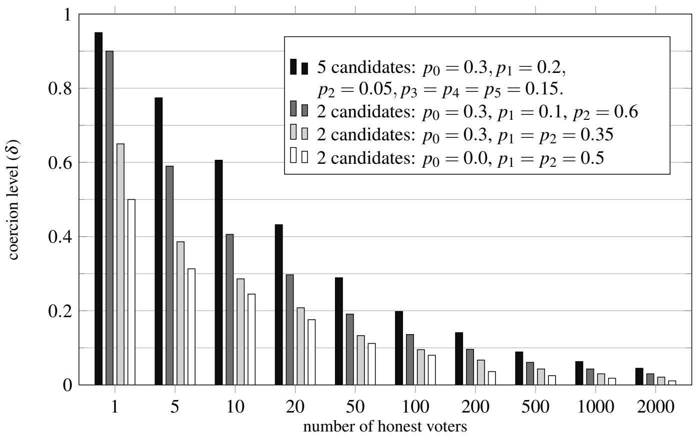


Figure 1: Level of coercion resistance ($\delta$) for the ideal protocol. The goal of the coerced voter is, in each case, to vote for candidate 1.

that even for national elections, it is realistic to assume that the number of voters is small, since results are often published per polling station and the number of voters who voted in one polling station is often not more than a few hundred.

The following example illustrates differences in the level of coercion resistance depending on the parameters.

**Example 1.** Consider two elections that use the ideal protocol. In both cases, we assume that the goal of the coerced voter is $\gamma_1$ (to vote for 1) and that the coercer is willing to pay \$50 to a coerced voter if (using some decision procedure) he decides that the voter followed his instructions.

In the first election, we assume 2000 honest voters, two candidates, and probabilities $p_0 = 0.3$, $p_1 = 0.35$, $p_2 = 0.35$ that an honest voter abstains from voting, chooses candidate 1, or chooses candidate 2, respectively. By Theorem 1 we know that this system is $(0.021)$-coercion-resistant w.r.t. $\tilde{V}_1 = \tilde{V}(\gamma_1)$. This means that if the coerced voter runs her counter-strategy to vote for her own candidate, then she will be paid — even if the coercer tries very hard in telling whether the coerced voter is following his instructions or not — with probability only 2.1% smaller, and thus will earn, on average, \$1.05 less, compared to the case when she follows the instructions of the coercer. Hence, in this case, the coerced voter has little incentive to follow the instructions of the voter. Conversely, by running the counter-strategy, the chance of being accused of not following the instructions of the coercer is not much bigger than in the case where the coerced voter follows the coercer's instructions.

In the second election, we take 100 honest voters, five candidates, and probabilities $p_0 = 0.3$, $p_1 = 0.2$, $p_2 = 0.05$, and $p_3 = p_4 = p_5 = 0.15$. In this case the system is only $(0.198)$-coercion resistant w.r.t. $\tilde{V}_1$, which means that the coerced voter can earn on average up to \$9.9 less when she runs her counter-strategy, which might give sufficient incentive to obey the coercer. Also, the chance of being accused of not following the coercer's instructions is now much higher when running the counter-strategy than when following the coercer's instructions.

## 5 Analyzing Bingo Voting

In this section, we prove that the Bingo voting system [4] enjoys the same level of coercion resistance as the ideal protocol, except for forced-abstention attacks, which the Bingo voting system does not prevent.

### 5.1 Description of the System

We describe the Bingo Voting system, which we denote by $\mathsf{P}_{\mathsf{Bingo}}$. In addition to the voters, the participants in this system are the following: (i) A *voting machine*, which is the main component in the voting process. (ii) A *trusted random number generator (RNG)*, which is an independent source of randomness, with its own display, and which is connected to the voting machine. (iii) A *bulletin board*. (iv) Some number of *auditors* who will contribute randomness in a distributed way used for randomized partial checking (RPC) in zero-knowledge proofs provided by the voting machine. While in our analysis we concentrate on the case of one voting machine, the analysis easily carries over to the case of several voting machines, as they are independent.

The election consists of three phases described below: initialization, voting, and tallying.

**Initialization phase.** In this phase, the voting machine, for every candidate $j$, generates $n$ random numbers $x_1^j, \dots, x_n^j$ (where $n$ is the number of voters), along with an unconditionally hiding commitment $c_i^j = \mathsf{comm}(j, x_l^j)$ for each pair $(j, x_l^j)$; more precisely, Pedersen commitments are used. All commitments are then shuffled and published on the bulletin board. Moreover, zero-knowledge proofs are published to guarantee that the same number $n$ of commitments is created for every candidate (see below for details).

**Voting phase.** In this phase, a voter enters the voting booth to indicate the candidate of her choice, say $j$, to the voting machine, by pressing a button corresponding to $j$. A voter can of course also abstain from voting. Then, the RNG creates a fresh random number which is displayed to the voter and transfered to the voting machine. The machine then prints a receipt consisting of the candidate names along with the following numbers next to them: The number next to the chosen candidate is the fresh random number, where the voter is expected to check that this number is the same as the one displayed by the RNG. Next to every other candidate $j'$, the machine prints a so far unused number $x_l^{j'}$, for some $l$.

**Tallying phase.** In this phase, the voting machine publishes the result of the election as well as all the receipts given to voters (in a lexicographical order). The machine also opens the commitments to all pairs $(j, x_l^j)$ where the number $x_l^j$ is unused, i.e., $x_l^j$ has not been printed on any receipt.

Moreover, the machine provides zero-knowledge proofs to show that the commitments that it has not opened yet can be correctly assigned to the receipts, i.e., for every receipt, $k-1$ commitments (belonging to $k-1$ different candidates and different for every receipt) can be assigned to $k-1$ different candidates so that the number next to a candidate coincides with the number in the corresponding commitment. These zero-knowledge proofs are described below.

The zero-knowledge proofs are checked as described below. If they are valid, every observer can verify the correctness of the result: the number of votes for candidate $j$ should be the number of opened commitments of the form $\mathsf{comm}(j, x_l^j)$, for some $x_l^j$, minus the number of abstaining voters.

**Zero-knowledge proofs.** Now, we describe the zero-knowledge proofs used both in the tallying phase and the initialization phase.

**Zero-knowledge proofs in the tallying phase.** First, the voting machine generates a new commitment on the pair $(j, r)$, where $j$ is the chosen candidate and $r$ is the number generated by the RNG and printed next to $j$. Then, all the commitments for the receipt are published in a random order: one of them is the commitment just described, the other $(k-1)$ commitments are unopened commitments published on the bulletin board in the initialization phase, where for different receipts, different commitments are taken from the bulletin board. An observer can verify that this is the case. Next, these commitments are re-randomized and shuffled twice; both the intermediate and the final set of commitments are published. The final commitments are opened. Now an observer can check that, for each candidate, there is exactly one commitment corresponding to the value printed next to this candidate. Finally, the auditors choose a random bit in some distributed way. Depending on the value of this bit, the voting machine publishes the random factors for the first or for the second re-randomization step.

If the voting machine tried to cheat, this would be detected with a probability of 50%; this probability can be increased to $1-(\frac{1}{2})^s$ by repeating the procedure $s$ times.

**Zero-knowledge proofs in the initialization phase.** This proof was not precisely defined in [4], but it can be implemented by randomized partial checking similarly to the zero-knowledge proof in the tallying phase. To this end, we assume that a commitment $\mathsf{comm}(j, x_l^j)$ on a pair $(j, x_l^j)$ is implemented as the pair $(C_l^j, D_l^j) = (\mathsf{comm}(j), \mathsf{comm}(x_l^j))$, where the commitments on the single components are Pederson commitments. Now, to show that among the published commitments there are exactly $n$ of the form $\mathsf{comm}(j, x_l^j)$ for every candidate $j$, the zero-knowledge proof proceeds similarly as in the tallying phase, except that it only uses the first component $C_l^j$ of a commitment.

### 5.2 Modeling and Security Assumptions

The modeling of the Bingo voting system as an election system $S = \mathsf{P}_{\mathsf{Bingo}}(k, m, n, \vec{p})$ is straightforward. We present a detailed model in Appendix B. Here, we highlight some modeling issues, and most importantly, our security assumptions.

**Voting authorities.** We assume that the voting machine and the random number generator are honest; the bulletin board may be dishonest. This assumption is necessary for the Bingo voting system to be coercion-resistant. (Note that for accountability and verifiability one does not have to assume that the voting machine is honest; see [19] for a formal definition of these security properties and an analysis of the Bingo voting w.r.t. these properties.) Note that we do not assume that auditors are honest.

**Honest voters.** As usual, an honest voter first makes a choice according to the probability distribution $\vec{p}$. If the choice is to abstain from voting, she abstains, otherwise follows the procedure described for the voting phase. After the voting phase is finished, an honest voter reveals her (paper) receipt, e.g., mails it to an organization to ask it to verify the correctness of the voting process w.r.t. her receipt or to publish it on some bulletin board. In particular, the coercer will get to see all receipts of honest voters, and hence, knows whether a voter voted or not. The assumption that the paper receipts are revealed after the voting phase is finished is reasonable. Also, the (presumably small) fraction of honest voters for which the coercer manages to get hold of the receipt earlier, could be considered to be dishonest. In any case, the assumption helps in the proof and we believe that our results also hold without that assumption.

**The coerced voter.** A coerced voter, running the dummy strategy or emulating it by running a counter-strategy, can communicate with the coercer and send her candidate on an untappable channel to the voting authority.

**The coercer.** The coercer can freely communicate with dishonest participants (voters and authorities) as well as with the coerced voter; in fact, dishonest participants are considered to be part of the coercer program. In a run of the system the coercer can see the following: (v1) his random coins, (v2) all messages published by the voting machine, both in the initialization phase and the tallying phase, (v3) receipts of all honest voters, as already explained above, and (v4) the messages received from the coerced voter and dishonest parties, including their receipts. However, the coercer cannot directly see the information the coerced voter obtains in the voting booth. In particular, the coerced voter can lie about what she sees and does in the voting booth, such as the random number shown by the RNG or the candidate she picked. So, while talking to the coercer on the phone would be allowed, taking pictures or videos should be prohibited (unless they can be manipulated on-the-fly, which, however, is unrealistic).

### 5.3 Coercion Resistance of the System

We now show that the Bingo voting system enjoys the same level of coercion resistance as the ideal protocol. However, since we assume that the coercer can observe whether or not a coerced voter enters the voting booth, the coerced voter can be forced to abstain from voting. (Another reason forced-abstention attacks are possible is that we allow the coercer to see the receipts of all voters and whether or not a voter has a receipt, and hence, voted or not.) The coerced voter can therefore only achieve the *favorite candidate up to forced abstention goal* $\gamma_i$, but not the *favorite candidate goal* $\gamma_i'$ (see Sections 2.3 and 4). For $\mathsf{P}_{\mathsf{Bingo}}(k,m,n,\vec{p})$, the goal $\gamma_i$, $i \in \{1,\dots,k\}$, is satisfied in a run if, whenever the coerced voter has indicated her candidate to the voting machine, she has successfully voted for the $i$-th candidate; in particular, if not instructed to vote by the coercer, the coerced voter does not have to vote in order to achieve $\gamma_i$. As usual, let $\tilde{V}_i = \tilde{V}(\gamma_i)$ be the set of programs of the coerced voter which guarantee $\gamma_i$, regardless of the actions of the coercer.

According to the terminology introduced in Section 2.3, we prove that the Bingo voting system achieves coercion resistance up to forced abstention with respect to the ideal $\delta$.

**Theorem 2.** *Let $S = \mathsf{P}_{\mathsf{Bingo}}(k,m,n,\vec{p})$. Then $S$ is $\delta$-coercion-resistant with respect to $\tilde{V}_i$, where $\delta = \delta^i_{min}(n,k,\vec{p})$.*

As already mentioned in Section 3, other approaches are unsuitable for the analysis of the Bingo voting system. We note that the simulation-based definitions [22, 28] cannot be applied due to the commitment problem. However, they might be applicable if we weakened the security assumptions, assuming that *all* auditors are honest; of course this is unrealistic — in fact, the reason to have multiple auditors is to allow some of them to be dishonest. Provided all auditors are honest, a simulator can simulate these auditors, which allows it to fake the zero-knowledge proofs in the tallying phase, as it “knows” the challenges. Another alternative could be to consider more advanced commitments, as, for example, in [23].

The remainder of this section is devoted to the proof of Theorem 2. First, we define the counter-strategy $\tilde{v}$ of the coerced voter: $\tilde{v}$ coincides with the dummy strategy $\mathsf{dum}$, with the following exceptions:

1. $\tilde{v}$ votes for candidate $i$, i.e., the coerced voter presses the button for candidate $i$, if the coercer instructs the coerced voter to vote for some candidate $j$.
2. If $\mathsf{dum}$ would forward the number that is shown on the display of the random number generator to the coercer, $\tilde{v}$ forwards the number next to the candidate $j$, as shown on her receipt.

It is easy to see that $\tilde{v} \in \tilde{V}_i$. Note that if the coercer does not instruct the coerced voter to vote for some candidate $j$ (forced-abstention attack), then following the counter-strategy the coerced voter abstains from voting, which is in accordance with the goal $\gamma_i$.

It remains to prove Condition (1) of Definition 1. For this purpose, let us fix a program $c$ of the coercer. We need to prove that $\mathsf{Pr}[T \mapsto 1] - \mathsf{Pr}[\tilde{T} \mapsto 1] \leq \delta$, where $T = (c \parallel \mathsf{dum} \parallel \mathsf{e}_S)$ and $\tilde{T} = (c \parallel \tilde{v} \parallel \mathsf{e}_S)$. The rest of the proof consists of the two parts already mentioned in the introduction: a cryptographic and a combinatorial part. The cryptographic part is Lemma 2. Using Lemma 2, the combinatorial part is a reduction to the ideal case, as studied in the previous section.

As introduced in Section 2.2, by $\omega_1 \in \Omega_1$ we denote a vector of choices made by the honest voters and by $\omega_2 \in \Omega_2$ we denote all the remaining random coins of a system. We denote by $\rho$ a view of the coercer, as described in Section 5.2, (v1)–(v4). We use the notion of a *pure result* $\vec{r} = (r_0, \ldots, r_k)$ as introduced in Section 4. In particular, it holds that $r_0 + \cdots + r_k = n+1$ and the coercer can compute this result from his view, by subtracting the votes of dishonest voters from the result of the election. We will denote the pure result determined by a view $\rho$ of the coercer by $\text{res}(\rho)$. A pure result determined by $\omega_1$ and the choice $j$ of the coerced voter will be denoted by $\text{res}(\omega_1, j)$.

For a coercer view $\rho$ in a run of the system, we denote by $f(\rho)$ the candidate the coercer wants the coerced voter to vote for; if the coercer does not instruct the coerced voter to vote, then $f(\rho)$ is undefined. Note that the coercer has to provide the coerced voter with $f(\rho)$ before the end of the election. Consequently, all messages the coercer has seen up to this point only depend on $\omega_2$ and are independent of the choices made by honest voters, which are determined by $\omega_1$. Therefore, we sometimes write $f(\omega_2)$ for the candidate the coercer wants the coerced voter to vote for in runs that use the random coins $\omega_2$.

The coercer can derive from his view which voters abstained from voting as he sees the receipts of the voters that successfully voted. Given a view $\rho$ of the coercer, we denote by $\text{abst}(\rho)$ the set of voters who abstained from voting, among the honest voters and the coerced voter; the number of such voters is referred to by $r_0(\rho) = |\text{abst}(\rho)|$. Below we will consider only views $\rho$ such that $f(\rho)$ is defined. In this case the set $\text{abst}(\rho)$ and the number $r_0(\rho)$ depend only on $\omega_1$. We will therefore also write $\text{abst}(\omega_1)/r_0(\omega_1)$.

For a coercer view $\rho$ in a run of $T$, let $\varphi_\rho$ be a predicate over $\Omega_1$ such that $\varphi_\rho(\omega_1)$ holds iff $\text{res}(\omega_1, f(\rho)) = \text{res}(\rho)$ and $\text{abst}(\omega_1) = \text{abst}(\rho)$, i.e., the choices $\omega_1$ of the honest voters are consistent with the view of the coercer, as far as the result of the election and the set of abstaining voters is concerned, in case the coerced voter runs the dummy strategy. Analogously, for $\tilde{T}$ we define that $\tilde{\varphi}_\rho(\omega_1)$ holds iff $\text{res}(\omega_1, i) = \text{res}(\rho)$ and $\text{abst}(\omega_1) = \text{abst}(\rho)$.

For a coercer view $\rho$, by $T(\omega_1, \omega_2) \mapsto \rho$ we denote the fact that the system $T$, when run with $\omega_1, \omega_2$, produces the view $\rho$ for the coercer. For a set $M$ of views, we write $T(\omega_1, \omega_2) \mapsto M$ if $T(\omega_1, \omega_2) \mapsto \rho$ for some $\rho \in M$. We write $\Pr[T \mapsto \rho]$ for the probability that a run of $T$ produces the view $\rho$ for the coercer, that is, $\Pr[T \mapsto \rho] = \Pr_{\omega_1,\omega_2}[T(\omega_1, \omega_2) \mapsto \rho]$; similarly for $\tilde{T}$.

The following lemma is the key fact used in the proof of Theorem 2 (see Appendix C for the proof). It constitutes the cryptographic part of the proof of Theorem 2. Intuitively, the lemma says that the view of the coercer is information-theoretically independent of the choices of honest voters and the coerced voter as long as these choices are consistent with the result of the election given in this view.

**Lemma 2.** *Let $\rho$ be a coercer view such that $f(\rho)$ is defined. Let $\omega_1^\rho$ and $\tilde{\omega}_1^\rho$ be some fixed elements of $\Omega_1$ such that $\varphi_\rho(\omega_1^\rho)$ and $\tilde{\varphi}_\rho(\tilde{\omega}_1^\rho)$, respectively. Then, the following equations hold true:*

$$\Pr[T \mapsto \rho] = \Pr_{\omega_1}[\varphi_\rho(\omega_1)] \cdot \Pr_{\omega_2}[T(\omega_1^\rho, \omega_2) \mapsto \rho] \tag{3}$$

$$\Pr[\tilde{T} \mapsto \rho] = \Pr_{\omega_1}[\tilde{\varphi}_\rho(\omega_1)] \cdot \Pr_{\omega_2}[\tilde{T}(\tilde{\omega}_1^\rho, \omega_2) \mapsto \rho] \tag{4}$$

$$\Pr_{\omega_2}[T(\omega_1^\rho, \omega_2) \mapsto \rho] = \Pr_{\omega_2}[\tilde{T}(\tilde{\omega}_1^\rho, \omega_2) \mapsto \rho] \ . \tag{5}$$

□

Now, using this lemma, we reduce the analysis of the Bingo voting system to the ideal case, linking the level of coercion resistance the Bingo voting system provides to the optimal bound $\delta_{min}$, established in Section 4.

Clearly, if $f(\rho)$ is defined, we have:

$$\begin{aligned}\Pr_{\omega_1}[\varphi_\rho(\omega_1)] = \Pr_{\omega_1}[\mathrm{res}(\omega_1, f(\rho)) = \mathrm{res}(\rho)]\cdot \\ \cdot \Pr_{\omega_1}[\mathrm{abst}(\omega_1) = \mathrm{abst}(\rho) \mid \mathrm{res}(\omega_1, f(\rho)) = \mathrm{res}(\rho)] \\ = A^{f(\rho)}_{\mathrm{res}(\rho)} \cdot \Pr_{\omega_1}[\mathrm{abst}(\omega_1) = \mathrm{abst}(\rho) \mid \mathrm{res}(\omega_1, f(\rho)) = \mathrm{res}(\rho)]\end{aligned}$$

and similarly

$$\Pr_{\omega_1}[\tilde{\varphi}_\rho(\omega_1)] = A^{i}_{\mathrm{res}(\rho)} \cdot \Pr_{\omega_1}[\mathrm{abst}(\omega_1) = \mathrm{abst}(\rho) \mid \mathrm{res}(\omega_1, i) = \mathrm{res}(\rho)].$$

Furthermore, we have

$$\begin{aligned}&\Pr_{\omega_1}[\mathrm{abst}(\omega_1) = \mathrm{abst}(\rho) \mid \mathrm{res}(\omega_1, f(\rho)) = \mathrm{res}(\rho)] = \\ &= \Pr_{\omega_1}[\mathrm{abst}(\omega_1) = \mathrm{abst}(\rho) | r_0(\omega_1) = r_0(\rho)] \\ &= \Pr_{\omega_1}[\mathrm{abst}(\omega_1) = \mathrm{abst}(\rho) \mid \mathrm{res}(\omega_1, i) = \mathrm{res}(\rho)],\end{aligned}$$

as the set of abstaining voters does not depend on the entire pure result of the election, but merely on the number of abstaining voters.

Together with Lemma 2, we immediately obtain for all $\omega_1^\rho$ with $\varphi_\rho(\omega_1^\rho)$:

$$\begin{aligned}&\Pr[T \mapsto \rho] - \Pr[\tilde{T} \mapsto \rho] \ = \\ &\left(A^{f(\rho)}_{\mathrm{res}(\rho)} - A^{i}_{\mathrm{res}(\rho)}\right) \cdot \Pr_{\omega_2}[T(\omega_1^\rho, \omega_2) \mapsto \rho] \cdot \Pr_{\omega_1}[\mathrm{abst}(\omega_1) = \mathrm{abst}(\rho) | r_0(\omega_1) = r_0(\rho)].\end{aligned}$$

Note that if there does not exist $\tilde{\omega}_1^\rho$ such that $\tilde{\varphi}_\rho(\tilde{\omega}_1^\rho)$, then $A^{i}_{\mathrm{res}(\rho)} = 0$ and $\Pr[\tilde{T} \mapsto \rho] = 0$.

Let $M$ be the set of views that are accepted by the program $c$ of the coercer, i.e., for which the coercer outputs 1. In what follows, let $j$ range over the set of candidate names $\{1, \ldots, k\}$, $\vec{r} = (r_0, \ldots, r_k)$ over all the pure results and $S$ over all subsets of honest voters, of which there are $n$, and the coerced voter. Let $M_j^{\vec{r},S} = \{\rho \in M : f(\rho) = j,\ \mathrm{abst}(\rho) = S \text{ and } \mathrm{res}(\rho) = \vec{r}\}$. Further, for $j, \vec{r}, S$ with $M_j^{\vec{r},S} \neq \emptyset$, let $\omega_1^{j,\vec{r},S}$ be an arbitrary element, such that $\mathrm{res}(\omega_1^{j,\vec{r},S}, j) = \vec{r}$ and $\mathrm{abst}(\omega_1^{j,\vec{r},S}) = S$. Then we have

$\varphi_\rho(\omega_1^{j,\vec{r},S})$ for all $\rho \in M_j^{\vec{r},S}$. Now, we obtain the following equations:

$$\begin{aligned}
\Phi &= \Pr[T \mapsto 1] - \Pr[\tilde{T} \mapsto 1] \\
&= \Pr[T \mapsto M] - \Pr[\tilde{T} \mapsto M] \\
&= \sum_j \sum_{\vec{r}} \sum_S \sum_{\rho \in M_j^{\vec{r},S}} \big(\Pr[T \mapsto \rho] - \Pr[\tilde{T} \mapsto \rho]\big) \\
&= \sum_j \sum_{\vec{r}} \sum_S \sum_{\rho \in M_j^{\vec{r},S}} (A_{\vec{r}}^j - A_{\vec{r}}^i) \cdot \Pr_{\omega_2}[T(\omega_1^{j,\vec{r},S}, \omega_2) \mapsto \rho] \cdot \Pr_{\omega_1}[\text{abst}(\omega_1) = S | r_0(\omega_1) = r_0] \\
&= \sum_j \sum_{\vec{r}} (A_{\vec{r}}^j - A_{\vec{r}}^i) \sum_S \sum_{\rho \in M_j^{\vec{r},S}} \Pr_{\omega_2}[T(\omega_1^{j,\vec{r},S}, \omega_2) \mapsto \rho] \cdot \Pr_{\omega_1}[\text{abst}(\omega_1) = S | r_0(\omega_1) = r_0].
\end{aligned}$$

For the third equation, we use that $\Pr[T \mapsto \rho] - \Pr[\tilde{T} \mapsto \rho] = 0$ if $f(\rho)$ is not defined. (Recall that if $f(\rho)$ is not defined, the coercer wants the coerced voter to abstain from voting, and that in this case the counter-strategy is to abstain. In other words, in runs in which the coercer wants the coerced voter to abstain from voting, there is no difference between the systems $T$ and $\tilde{T}$.) Let $M_{i,j}^* = \{\vec{r} : A_{\vec{r}}^j \geq A_{\vec{r}}^i\}$. Then, we obtain:

$$\Phi \leq \sum_j \sum_{\vec{r} \in M_{i,j}^*} (A_{\vec{r}}^j - A_{\vec{r}}^i) \sum_S \sum_{\rho \in M_j^{\vec{r},S}} \Pr_{\omega_2}[T(\omega_1^{j,\vec{r},S}, \omega_2) \mapsto \rho] \cdot \Pr_{\omega_1}[\text{abst}(\omega_1) = S \mid r_0(\omega_1) = r_0].$$

Next, we use that, by the definition of $M_j^{\vec{r},S}$, for $\rho \in M_j^{\vec{r},S}$ we have $f(\rho) = j$ and, because $f(\rho)$ depends only on $\omega_2$, $T(\omega_1^{j,\vec{r},S}, \omega_2) \mapsto \rho$ implies $f(\omega_2) = j$. Therefore, we have $\sum_{\rho \in M_j^{\vec{r},S}} \Pr_{\omega_2}[T(\omega_1^{j,\vec{r},S}, \omega_2) \mapsto \rho] \leq \Pr_{\omega_2}[f(\omega_2) = j]$. Now, we can conclude:

$$\begin{aligned}
\Phi &\leq \sum_j \sum_{\vec{r} \in M_{i,j}^*} (A_{\vec{r}}^j - A_{\vec{r}}^i) \sum_S \Pr_{\omega_1}[\text{abst}(\omega_1) = S \mid r_0(\omega_1) = r_0] \cdot \Pr_{\omega_2}[f(\omega_2) = j] \\
&= \sum_j \Pr_{\omega_2}[f(\omega_2) = j] \sum_{r \in M_{i,j}^*} (A_r^j - A_r^i) \sum_S \Pr_{\omega_1}[\text{abst}(\omega_1) = S \mid r_0(\omega_1) = r_0] \\
&\leq \sum_j \Pr_{\omega_2}[f(\omega_2) = j] \sum_{\vec{r} \in M_{i,j}^*} (A_{\vec{r}}^j - A_{\vec{r}}^i) \\
&\leq \sum_j \Pr_{\omega_2}[f(\omega_2) = j] \cdot \delta_{min}^i(n,k,\vec{p}) \ \leq \delta_{min}^i(n,k,\vec{p}) = \delta.
\end{aligned}$$

This completes the proof of Theorem 2.

## 6 Analyzing ThreeBallot

The ThreeBallot voting system [26, 25] is designed to provide (some degree of) coercion resistance and verifiability. We measure how the level of coercion resistance of ThreeBallot degrades as the number of candidates increases, i.e., we study the level of coercion resistance of ThreeBallot in case the so-called short ballot assumption is not

(a) | A: o B: x | A: x B: o | A: o B: x |        (b) | A: x B: x | A: o B: o | A: o B: x |

Figure 2: Two ways of voting for the second candidate (candidate B) in the ThreeBallot protocol, where x represents a marked position and o represents an unmarked position. All other ways of voting for B can be obtained as permutations of these two.

met. We also show that the level of coercion resistance ThreeBallot provides is significantly lower than that of an ideal system, even in case of short ballots. We first recall the ThreeBallot voting system and state our security assumptions.

### 6.1 Description of the System

In ThreeBallot, a voter is given a multi-ballot consisting of three simple ballots, where the candidates are written in a fixed order. In the secrecy of a voting booth, the voter is supposed to fill out all three simple ballots in the following way: she marks the candidate of her choice on exactly *two* simple ballots and every other candidate on exactly *one* simple ballot. Figure 2 shows two ways of voting for candidate B in an election with two candidates. After this, she feeds all three simple ballots to a machine (some kind of scanner) and indicates the simple ballot she wants to get as a receipt. The machine checks the well-formedness of the multi-ballot, prints secretly random numbers on each simple ballot, where numbers on different simple ballots are chosen independently, and gives the voter a copy of the chosen simple ballot, with the random number printed on it. Note that the voter does not get to see the random numbers of the remaining two simple ballots. The scanner keeps all ballots.

In the tallying phase, all cast simple ballots are shuffled by a voting authority and published on a bulletin board. From the simple ballots displayed on the bulletin board, everyone can determine the result of the election as follows: the number of votes for the $i$-th candidate is the number of simple ballots with the $i$-th position marked minus the total number of votes, which is the total number of simple ballots on the bulletin board divided by three.

Intuitively, the system is coercion-resistant (at least to some extent), as every receipt — a simple ballot — can be part of a multi-ballot for any candidate. The fact that every voter can check whether her receipt appears on the bulletin board is supposed to provide some ensures that the ballots were not tampered with (verifiability).

In our analyzes we consider the variant of ThreeBallot as proposed in [10]. In this variant a specific way of filling out the ballots is proposed: A voter first, for each candidate, marks the position corresponding to this candidate on a randomly chosen simple ballot. Then, she randomly chooses one simple ballot to be taken as a receipt. Finally, she marks the position corresponding to the candidate of her choice on some simple ballot, but not the one she chose as a receipt; if there is more than one possibility, one of the two possible simple ballots is chosen randomly. The advantage of this procedure is that the receipt an honest voter gets is stochastically independent from the candidate the voter votes for, which gives better privacy. We note that in [10], ThreeBallot was analyzed in a simulation-based setting, focussing on privacy. The analysis was based on the (only informally stated) assumption that the adversary is not able to reconstruct the multi-ballot corresponding to a receipt. However, this assumption is unjustified: runs for which an adversary can reconstruct the multi-ballots occur with non-negligible probability (see Section 6.3).

### 6.2 Modeling and Security Assumptions

We now highlight some modeling issues and our security assumptions for ThreeBallot. From our description it is easy to derive a formal model of ThreeBallot as an election system, along the lines of our model for the Bingo voting system presented in Appendix B.

**Voting authorities.** We assume that the scanner and the authorities in charge of shuffling the ballots are honest; the bulletin board may be dishonest. We note that without this assumption, ThreeBallot would not be coercion-resistant; it would provide only 1-coercion resistance.

**Honest voters.** As usual, an honest voter first makes a choice according to the probability distribution $\vec{p}$. If the choice is to abstain from voting, she abstains, and otherwise, follows the procedure described in Section 6.1 for the voting phase. After the voting phase is finished, an honest voter may reveal her (paper) receipt. However, to measure how much information a coercer gains from the receipts of honest voters, we will also consider the case in which the coercer does not see the receipts of honest voters.

**The coerced voter.** A coerced voter, running the dummy strategy or emulating it by running a counter-strategy, can communicate with the coercer. Just as an honest voter, she can also fill out a multi-ballot, feed it to the scanner and pick a receipt. If the coerced voter follows the dummy strategy, she will carry out these steps following the instructions of the coercer. Of course, if she follows the counter-strategy she can deviate from these instructions.

**The coercer.** As usual, the coercer subsumes dishonest voters and can freely communicate with the coerced voter. In a run of the system, the coercer can see the following: (v1) his random coins, (v2) the bulletin board consisting of the shuffled simple ballots (with serial numbers) cast by the voters (v3) depending on the case under consideration, the receipts of the honest voters, after the voting-phase is finished, (v4) the messages received from the coerced voter, including the receipt of the coerced voter. As in case of Bingo voting, the coercer cannot directly see the information the coerced voter obtains or the actions she performs in the voting booth.

By $\mathsf{P}^{-}_{\mathsf{ThreeBallot}}(k,m,n,\vec{p})$ we will denote the election system which models the ThreeBallot protocol along the line of the above assumptions, where the coercer cannot see the receipts of the honest voters. Similarly, by $\mathsf{P}^{+}_{\mathsf{ThreeBallot}}(k,m,n,\vec{p})$ we denote the version of the system where the coercer can see the receipts of the honest voters after the voting phase is finished.

### 6.3 ThreeBallot with Two Candidates

Based on our definition, we now precisely measure the level of coercion resistance ThreeBallot provides and show that it is significantly lower than that of an ideal system, even in case of short ballots, and hence, under the so-called short ballot assumption (see, e.g., [25]). More precisely, we analyze the case of two candidates. The case for multiple candidates will be studied in Section 6.4.

As a warm-up, we note that the bulletin board and the receipts potentially reveal more information to the coercer than just the result of the election: It may happen, for instance, that the multi-ballots of all voters are of the form $(\underline{\substack{x\\o}},\substack{x\\x},\substack{o\\o})$ or $(\underline{\substack{o\\x}},\substack{x\\x},\substack{o\\o})$, where the underlined ballots ($\substack{x\\o}$ and $\substack{o\\x}$, respectively) are picked as receipts. In this case, a receipt directly indicates the choice of the voter, which allows for successful coercion.

In what follows, we often use the above notation for multi-ballots and the receipt picked, and refer to this object as a *pattern*. A pattern does not fix the order of simple ballots, e.g., $(\underline{\substack{o\\x}},\substack{x\\x},\substack{o\\o})$ is considered to be the same pattern as, say, $(\substack{x\\x},\underline{\substack{o\\x}},\substack{o\\o})$.

As in our previous case studies, our analysis is with respect to the goal $\gamma_i$ (favorite candidate up to forced abstention), for $i \in \{1,2\}$, which requires the voter to vote for candidate $i$, if she is instructed by the coercer to vote for some, possibly different, candidate. As usual, we define $\tilde{V}_i = \tilde{V}(\gamma_i)$. Note that, just as in the case of the Bingo voting system, the stronger goal $\gamma_i'$ (favorite candidate) cannot be achieved, and hence, ThreeBallot does not achieve strong coercion resistance, but merely coercion resistance up to forced abstention (recall the terminology from Section 2.3).

We proceed as follows: First, we define a counter-strategy, which is optimal for the coerced voter. Second, we define the constant $\delta$, which describes the optimal level of coercion resistance ThreeBallot achieves. For this, we introduce what we call an essential view of the coercer which abstracts away from some details of the actual view of the coercer. Finally, we state the main result of this section, namely $\delta$-coercion resistance of ThreeBallot and the optimality of $\delta$. This is done both for the case where the coercer gets to see all receipts of voters and for the case where receipts of honest voters are hidden from the coercer, resulting in two constants $\delta_{TB^+}$ and $\delta_{TB^-}$.

**Counter-strategy.** We define the *counter-strategy* of the coerced voter to coincide with the dummy strategy with one exception: if the coerced voter is requested to fill out her ballot and cast it according to a certain pattern $Z$, then the coerced voter will, instead, fill out the ballot according to $C(Z,i)$, as defined next. (Recall that the goal of the coerced voter is to vote for $i$.)

We define $C(Z,i)$ in such a way that it yields the same receipt as $Z$ does and adjusts the two remaining ballots in such a way that the resulting multi-ballot is a vote for candidate $i$. By this requirement, $C(Z,i)$ is uniquely determined, except for two cases: $C((\underline{\substack{x\\o}},\substack{o\\x},\substack{o\\x}),1)$ and $C((\underline{\substack{o\\x}},\substack{x\\o},\substack{x\\o}),2)$. In the former case, for instance, one can take $(\underline{\substack{x\\o}},\substack{x\\o},\substack{o\\x})$, $(\underline{\substack{x\\o}},\substack{x\\x},\substack{o\\o})$, or randomly pick one of the two, possibly based on further information. For these cases, we define $C((\underline{\substack{x\\o}},\substack{o\\x},\substack{o\\x}),1) = (\underline{\substack{x\\o}},\substack{x\\o},\substack{o\\x})$ and $C((\underline{\substack{o\\x}},\substack{x\\o},\substack{x\\o}),2) = (\underline{\substack{o\\x}},\substack{o\\x},\substack{x\\o})$. We use this strategy in the proof of Theorem 3. As we will see, from the proof of this theorem it follows that this counter-strategy achieves the maximal level of coercion resistance and, in this sense, is optimal for the coerced voter.

**Essential views.** In the essential view of the coercer, we abstract away from the following information: the serial numbers on the simple ballots, the order of the simple ballots on the bulletin board, the order of the receipts of the honest voters (if considered), the random coins of the coercer (i.e., randomness does not help the coercer), the receipt of the coerced voter (as both in the dummy strategy and the counter-strategy as defined above, she returns what the coercer expects her to return) and the simple ballots of the dishonest voters (which are as expected by the coercer).

More precisely, an *essential view* of the coercer consists only of (i) three integers $n_{\substack{\times\\\times}}, n_{\substack{\times\\\circ}}, n_{\substack{\circ\\\times}}$, indicating the number of the respective simple ballots on the bulletin board and (ii) if the coercer can see the receipts of honest voters, three integers $rec = (r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}})$, indicating the number of the respective receipts taken by the those voters. Note that from these numbers the number of $\binom{\circ}{\circ}$-ballots on the bulletin board and the number of $\binom{\circ}{\circ}$-receipts can be derived by the coercer: For the $\binom{\circ}{\circ}$-ballots, observe that the number of $\binom{\circ}{\circ}$-ballots coincides with the number of $\binom{\times}{\times}$-ballots. From this, we immediately get the number of non-abstaining voters $N$, as it is the sum of all the ballots divided by three. Now, we get the number of $\binom{\circ}{\circ}$-receipts by subtracting the number of the other receipts from $N$.

By $V^+$ and $V^-$ we denote the set of all essential views of the coercer, for the cases in which he can or cannot see the receipts of the honest voters, respectively.

**The constants $\delta^i_{TB^-}$ and $\delta^i_{TB^+}$.** To define these constants we use the probability $A^Z_\rho$ that the choices made by the honest voters and the coerced voter result in an essential view $\rho$, given that the coerced voter chooses the pattern $Z$.

The intuition behind the result given below is similar to the one for the ideal protocol (Section 4): If the coercer wants the coerced voter to choose the pattern $Z$ and the coerced voter wants to vote for candidate $i$, then the best strategy of the coercer to distinguish whether the coerced voter has chosen $Z$ or $C(Z,i)$ is to accept a run if the essential view $\rho$ in this run is such that $A^{C(Z,i)}_\rho \leq A^Z_\rho$. Let $M^-_{Z,i} = \{\rho \in V^- : A^{C(Z,i)}_\rho \leq A^Z_\rho\}$ and $M^+_{Z,i} = \{\rho \in V^+ : A^{C(Z,i)}_\rho \leq A^Z_\rho\}$ be the sets of those essential views for which—according to his best strategy—the coercer should accept the run.

Now, we are ready to define the constants expressing the (optimal) level of coercion resistance of ThreeBallot, for the case that the coercer cannot see the receipts of the honest voters:

$$\delta^i_{TB^-}(n,\vec{p}) = \max_Z \sum_{\rho \in M^-_{Z,i}} (A^Z_\rho - A^{C(Z,i)}_\rho) \ , \tag{6}$$

and for the case that the coercer can see these receipts:

$$\delta^i_{TB^+}(n,\vec{p}) = \max_Z \sum_{\rho \in M^+_{Z,i}} (A^Z_\rho - A^{C(Z,i)}_\rho). \tag{7}$$

The following theorem shows that the two constants (more precisely, functions) just defined in fact capture the optimal level of coercion resistance provided by ThreeBallot in case of an election with two candidates. Also, as mentioned before, we consider coercion resistance up to forced abstention.

**Theorem 3.** *1. The system* $\mathsf{P}^{-}_{\mathsf{ThreeBallot}}(2,m,n,\vec{p})$ *is* $\delta$*-coercion resistant with respect to* $\tilde{V}_i$ *for* $\delta = \delta^{i}_{TB^-}(n,\vec{p})$.

*2. The system* $\mathsf{P}^{+}_{\mathsf{ThreeBallot}}(2,m,n,\vec{p})$ *is* $\delta$*-coercion resistant with respect to* $\tilde{V}_i$ *for* $\delta = \delta^{i}_{TB^+}(n,\vec{p})$.

*Moreover, in both cases, the constants are optimal, i.e., the systems are not* $\delta'$*-coercion-resistant for any* $\delta' < \delta$.

The proof of this theorem is given in Appendix D. The main part of this proof is to show that the additional information given in a full view of the coercer, and omitted in an essential view, can safely be discarded. This is similar in spirit to the proof of Theorem 2, where we reduced the analysis of the Bingo voting system to the ideal protocol, although the technical details differ and are simpler for ThreeBallot.

For ThreeBallot the bigger challenge is to come up with explicit formulas for the probabilities $A^Z_\rho$ in order to be able to compute the level of coercion resistance for concrete parameters; this is particularly true for the case where the coercer can see the receipts of honest voters. The formulas are stated in the following two lemmas.

**Lemma 3.** *Consider the case where the coercer cannot see the receipts of the honest voters. Let* $\rho$ *be an essential view of the form* $(n_{\substack{\times\\\times}}, n_{\substack{\times\\\circ}}, n_{\substack{\circ\\\times}})$ *. Then, we have* $A^Z_\rho = A_{\rho - Z}$*, where* $\rho - Z$ *denotes the view we get by removing the ballots of* $Z$ *from* $\rho$ *and where, for* $\rho = (n_{\substack{\times\\\times}}, n_{\substack{\times\\\circ}}, n_{\substack{\circ\\\times}})$,

$$A_\rho = \binom{n}{N}\binom{N}{R} \cdot p_0^{n-N} \cdot p_1^{R} \cdot p_2^{N-R} \cdot \binom{N}{n_{\substack{\times\\\times}}}\left(\frac{2}{3}\right)^{n_{\substack{\times\\\times}}}\left(\frac{1}{3}\right)^{N-n_{\substack{\times\\\times}}},$$

*with* $N = (2n_{\substack{\times\\\times}} + n_{\substack{\times\\\circ}} + n_{\substack{\circ\\\times}})/3$ *denoting the total number of non-abstaining voters and* $R = (n_{\substack{\times\\\times}} + n_{\substack{\times\\\circ}}) - N$ *denoting the number of votes for candidate 1.*

*Proof.* An elementary calculation yields that the probability that an honest voter submits a multi-ballot where one simple-ballot is of the form $\substack{\times\\\times}$ is $\frac{2}{3}$. Moreover, this is independent of the chosen candidate. Hence, to compute the probability that a given view $(n_{\substack{\times\\\times}}, n_{\substack{\times\\\circ}}, n_{\substack{\circ\\\times}})$ occurs, we first choose $N$ non-abstaining voters, and among these non-abstaining voters, we (independently) distribute $R$ voters for candidate 1 and $n_{\substack{\times\\\times}}$ voters that submit a simple ballot of the form $\substack{\times\\\times}$. From a simple combinatorial argument we then obtain the above formula. □

The following formula for the case where the coercer gets to see the receipts of all voters is harder to obtain (see Appendix D for the proof of the following lemma).

**Lemma 4.** *Consider the case where the coercer can see the receipts of the honest voters. Let* $\rho$ *be an essential view of the form* $(n_{\substack{\times\\\times}}, n_{\substack{\times\\\circ}}, n_{\substack{\circ\\\times}}, r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}})$*. Then, we have*

*$A^Z_\rho = A_{\rho - Z}$, where, for $\rho = (n_{\substack{\times\\\times}}, n_{\substack{\times\\\circ}}, n_{\substack{\circ\\\times}}, r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}})$,*

$$
\begin{aligned}
A_\rho = \binom{n}{N} p_0^{n-N} p_1^R p_2^{N-R} \cdot \binom{N}{r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}}} \cdot \left(\frac{1}{9}\right)^{r_{\substack{\times\\\times}}} \left(\frac{2}{9}\right)^{r_{\substack{\times\\\circ}}+r3} \left(\frac{4}{9}\right)^{N-r_{\substack{\times\\\times}}-r_{\substack{\times\\\circ}}-r_{\substack{\circ\\\times}}} \cdot \\
\cdot \sum_{\tau_1,\tau_2} \binom{r_{\substack{\times\\\circ}} + r_{\substack{\circ\\\times}} - \tau_1 - \tau_2}{n_{\substack{\times\\\times}} - N + r_{\substack{\times\\\circ}} + r_{\substack{\circ\\\times}}} \left(\frac{1}{2}\right)^{r_{\substack{\times\\\circ}}+r_{\substack{\circ\\\times}}-\tau_1-\tau_2} \cdot \\
\cdot \binom{N - r_{\substack{\times\\\circ}} - r_{\substack{\circ\\\times}}}{R - (r_{\substack{\times\\\circ}} - \tau_1) - \tau_2} \binom{r_{\substack{\times\\\circ}}}{\tau_1} \binom{r_{\substack{\circ\\\times}}}{\tau_2}.
\end{aligned}
$$

*with $N$ as in Lemma 3, and $\tau_1$ and $\tau_2$ ranging over the set $\{0,\ldots,n\}$. We use the convention that $\binom{m}{l} = 0$ for $m < 0$.*

Using these formulas, we have computed the level of coercion resistance for several concrete values (see Figure 3). Figure 3 also shows the level of coercion resistance of the ideal protocol in order to be able to compare the level of coercion resistance ThreeBallot provides with the one for the ideal protocol.

As can be seen from Diagram (a) in Figure 3, when the coercer can see the receipts of honest voters, and there are at least five honest voters, the level of coercion resistance of the ideal protocol is about double that of ThreeBallot; that is, the value for $\delta$ for the ideal protocol is half of that for ThreeBallot. This difference is quite significant. It means that for ThreeBallot, the expected gain when trying to sell one's vote (by following the instructions of the coercer instead of running the counter-strategy) is twice as high as in the ideal protocol. Conversely, by running the counter-strategy rather than obeying the coercer, the expected growth in the risk of being caught is twice as big as in the ideal protocol.

The difference between ThreeBallot and the ideal protocol is smaller if the coercer cannot see the receipts of honest voters. We found that it also decreases if the probability distribution for the candidates is less uniform (see Diagram (b) in Figure 3).

As already pointed out in Section 3, other cryptographic definitions for coercion resistance proposed in the literature are unsuitable for analyzing the coercion resistance of ThreeBallot.

### 6.4 ThreeBallot with multiple candidates

We now analyze the degradation of coercion resistance of ThreeBallot as the number of candidates grows, i.e., the case where the so-called small ballot assumption is not met. The degradation itself is not surprising since certain patterns become very unlikely to occur. This has been noted for variants of ThreeBallot, for example, in [8, 14]. However, our definition allows us to measure the degradation rigorously in the context of coercion resistance, showing that the level of coercion resistance ThreeBallot provides is completely insufficient already with five to seven candidates and a few hundred voters.

More precisely, in this section we state negative results for ThreeBallot with multiple candidates by providing *lower bounds* for the level of coercion of ThreeBallot, that is, we show that the studied systems are *not* $\delta$-coercion-resistant for any $\delta$ smaller than

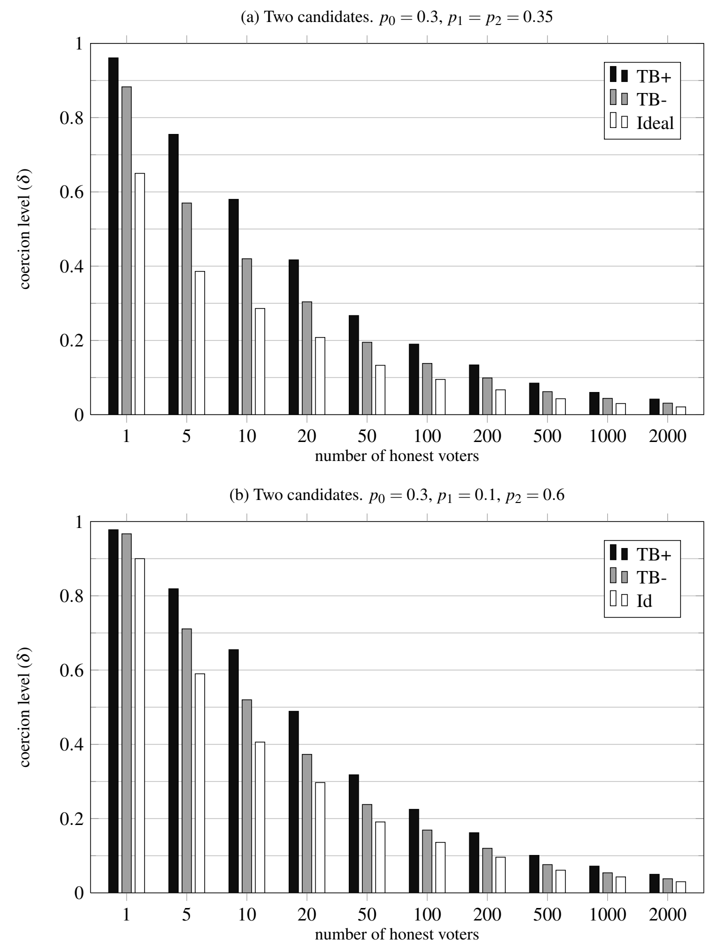


Figure 3: Level of coercion resistance ($\delta$) for the ideal protocol (Id), ThreeBallot without revealing receipts of the honest voters (TB-) and with revealed receipts of the honest voters (TB+).

the lower bound. (Unlike the two candidate case, we do not show that the systems *are* coercion-resistant for the given lower bound.) These results apply both to the case with and without receipts.

To obtain the lower bounds, we consider a restricted class $\mathscr{C} \subseteq C_S$ of programs of the coercer and define a counter-strategy for the coerced voter, which we show to be optimal w.r.t. $\mathscr{C}$. We then calculate the optimal $\delta$ w.r.t. $\mathscr{C}$ and the considered counter-strategy.

The class $\mathscr{C}$ is defined as follows. In every program $c \in \mathscr{C}$, the coercer instructs the coerced voter to vote for some candidate $j$ by marking all positions on one single ballot (we will call such a single ballot *fully marked*), the $j$-th position on the second ballot, and no position on the third ballot. The coercer then asks the coerced voter to bring the second ballot (the one with one position marked) as a receipt. The program $c$, which decides whether to accept a run, only uses the following parts of its view: (v1) the receipt given by the coerced voter, (v2) the pure result $\vec{r} = (r_0, \ldots, r_k)$, as introduced in Section 4, and (v3) the number $u$ of all fully marked ballots on the bulletin board cast by honest voters and the coerced voter. We call a tuple of the form $\rho = (\vec{r}, u)$ a *restricted view* of the coercer. Similarly to the two-candidate case, the receipt of the coerced voter is not part of the restricted view as the counter-strategy will always return the expected receipt.

Now, let $\gamma_i$ denote the goal as specified in the two candidate case. We define the counter-strategy $v^*$ as follows: The coerced voter, if instructed to vote in a way specified by the coercer, fills out the multi-ballot in such a way that (a) she votes for $i$ and (b) one of the single ballots is the required receipt. This can be done in possibly many ways; $v^*$ just fixes one of them.

This counter-strategy is optimal for $\mathscr{C}$ because any two strategies satisfying (a) and (b) produce exactly the same restricted views (since they do not use fully marked ballots), and it is clear that any successful counter-strategy has to satisfy (a) and (b).

Now, the technique for obtaining the lower bound is very similar to the one used for the case with two candidates without receipts.

Let $n$, $k$, and $\vec{p}$ be as usual. Let $\rho = (\vec{r}, u)$ be a restricted view. We will denote by $A_\rho^{i,o}$ ($A_\rho^{i,c}$) the probability that the choices of the honest voters and the coerced voter result in the restricted view $\rho$, given that the coerced voter votes for candidate $i$ with (without) one fully marked ballot. Note that if the coerced voter obeys the instructions of the coercer, her multi-ballot contains a fully marked ballot; otherwise, it does not. It is easy to see that

$$A_\rho^{i,o} = A_{\vec{r}}^{i} \cdot \binom{n - r_0}{u - 1} q^{u-1} (1-q)^{n-r_0-u+1}$$

and

$$A_\rho^{i,c} = A_{\vec{r}}^{i} \cdot \binom{n - r_0}{u} q^{u} (1-q)^{n-r_0-u},$$

respectively, where $A_{\vec{r}}^{i}$ is defined as in Section 4 and $q = \frac{2}{3^{k-1}}$ is the probability that an honest, non-abstaining voter produces a fully marked ballot (this probability can be obtained by some elementary computations).

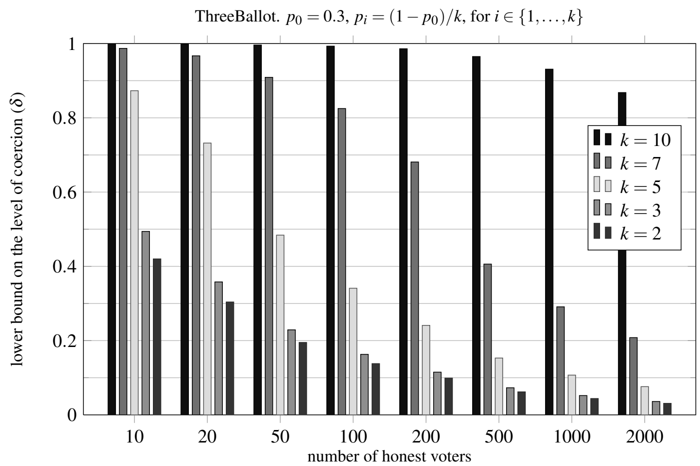


Figure 4: Lower bounds for coercion resistance ($\delta$) of the ThreeBallot system, with the probability distribution $\vec{p}$ over candidates defined as follows: $p_0 = 0.3$ (probability for abstention) and $p_i = (1-p_0)/k$, for every candidate $i \in \{1,\dots,k\}$.

Let $M_{i,j}$ be the set of those restricted views $\rho$ for which $A_\rho^{i,c} < A_\rho^{j,o}$ and let

$$\delta_i(n,k,\vec{p}) = \max_{j\in\{1,\dots,k\}} \sum_{\rho\in M_{i,j}} (A_\rho^{j,o} - A_\rho^{i,c}) \ . \tag{8}$$

Then, we obtain the following result, with the proof presented in Appendix E:

**Theorem 4.** *Let $S = \mathsf{P}_{\mathsf{ThreeBallot}}(k,m,n,\vec{p})$. Then $S$ is not $\delta$-coercion resistant w.r.t. $\tilde{V}_i$ for any $\delta < \delta_i(n,k,\vec{p})$.*

This result allows us to compute lower bounds for the level of coercion of ThreeBallot for concrete values, with some examples depicted in Figure 4. As already mentioned, the figure shows that the level of coercion resistance ThreeBallot provides is completely insufficient already with five to seven candidates and a few hundred voters. Note that the values of $\delta$ can even be higher than depicted in this figure.

## 7 Conclusion

In this paper, we presented a simple and intuitive definition of coercion resistance, in the style of game-based cryptographic definitions. Our definition allows us to precisely measure the level of coercion resistance a protocol has, in terms of the parameters $\delta$ (which measures the chances of a coercer to successfully tell whether or not a coerced voter followed the coercer's instructions) and $\tilde{V}$ (which specifies the goal a coerced voter should be able to achieve under coercion, such as voting for a certain candidate, with or without tolerating forced-abstention attacks). As demonstrated by our case studies, this flexibility is important in order to be able to make reasonable statements about voting protocols; simple yes/no-answers are often too coarse. Our case studies also illustrate the wide applicability of our definition. The results we obtain for the Bingo voting and ThreeBallot systems are out of the scope of existing approaches. Our proofs exhibit a similar overall structure, and by this, suggest a useful proof technique. Also, the results for the ideal protocol are interesting in this respect as in some cases—demonstrated for Bingo voting—they allow the analysis to be reduced to the ideal case.

We note that in subsequent work [20] we also analyzed Scantegrity II [6], for which we show that, similarly to Bingo voting, it provides an ideal level of coercion resistance (up to forced-abstention attacks). Again this result could not have been obtained with other definitions of coercion resistance.

**Acknowledgements.** We would like to thank the anonymous reviewers for their helpful comments. This work was carried out within the Priority Programme 1496 *Reliably Secure Software Systems – RS*$^3$ supported by *Deutsche Forschungsgemeinschaft* (DFG) under Grant KU 1434/6-1.

## References

[1] M. Backes, C. Hritcu, and M. Maffei. Automated Verification of Remote Electronic Voting Protocols in the Applied Pi-Calculus. In *Proceedings of the 21st IEEE Computer Security Foundations Symposium (CSF 2008)*, pages 195–209. IEEE Computer Society, 2008.

[2] M. Bär, C. Henrich, J. Müller-Quade, S. Röhrich, and C. Stüber. Real World Experiences with Bingo Voting and a Comparison of Usability. In *IAVoSS Workshop On Trustworthy Elections (WOTE 2008)*, 2008.

[3] J. C. Benaloh and D. Tuinstra. Receipt-free secret-ballot elections (extended abstract). In *Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing (STOC 1994)*, pages 544–553. ACM Press, 1994.

[4] J.-M. Bohli, J. Müller-Quade, and S. Röhrich. Bingo Voting: Secure and Coercion-Free Voting Using a Trusted Random Number Generator. In A. Alkassar and M. Volkamer, editors, *E-Voting and Identity (VOTE-ID 2007)*, volume 4896 of *Lecture Notes in Computer Science*, pages 111–124. Springer, 2007.

[5] R. Canetti and R. Gennaro. Incoercible Multiparty Computation (extended abstract). In *37th Annual Symposium on Foundations of Computer Science (FOCS '96)*, pages 504–513. IEEE Computer Society, 1996.

[6] D. Chaum, R. Carback, J. Clark, A. Essex, S. Popoveniuc, R. L. Rivest, P. Y. A. Ryan, E. Shen, and A. T. Sherman. Scantegrity II: End-to-End Verifiability for Optical Scan Election Systems using Invisible Ink Confirmation Codes. In *USENIX/ACCURATE Electronic Voting Technology (EVT 2008)*. USENIX Association, 2008. See also `http://www.scantegrity.org/elections.php`.

[7] B. Chevallier-Mames, P.-A. Fouque, D. Pointcheval, J. Stern, and J. Traoré. On Some Incompatible Properties of Voting Schemes. In *IAVoSS Workshop On Trustworthy Elections (WOTE 2006)*, 2006.

[8] Jacek Cichon, Miroslaw Kutylowski, and Bogdan Weglorz. Short Ballot Assumption and Threeballot Voting Protocol. In Viliam Geffert, Juhani Karhumäki, Alberto Bertoni, Bart Preneel, Pavol Návrat, and Mária Bieliková, editors, *Theory and Practice of Computer*

*Science, 34th Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM 2008)*, volume 4910 of *Lecture Notes in Computer Science*, pages 585–598. Springer, 2008.

[9] M. R. Clarkson, S. Chong, and A. C. Myers. Civitas: Toward a Secure Voting System. In *2008 IEEE Symposium on Security and Privacy (S&P 2008)*, pages 354–368. IEEE Computer Society, 2008.

[10] O. de Marneffe, O. Pereira, and J.-J. Quisquater. Simulation-Based Analysis of E2E Voting Systems. In A. Alkassar and M. Volkamer, editors, *E-Voting and Identity (VOTE-ID 2007)*, volume 4896 of *Lecture Notes in Computer Science*, pages 137–149. Springer, 2007.

[11] S. Delaune, S. Kremer, and M.D. Ryan. Coercion-Resistance and Receipt-Freeness in Electronic Voting. In *Proceedings of the 19th IEEE Computer Security Foundations Workshop (CSFW'06)*, pages 28–39. IEEE Computer Society Press, 2006.

[12] R. Gardner, S. Garera, and A. D. Rubin. Coercion Resistant End-to-end Voting. In Roger Dingledine and Philippe Golle, editors, *13th International Conference on Financial Cryptography (FC 2009)*, volume 5628 of *Lecture Notes in Computer Science*, pages 344–361. Springer, 2009.

[13] O. Goldreich. *Foundations of Cryptography – Basic Tools*, volume 1. Cambridge Press, 2001.

[14] Kevin Henry, Douglas R. Stinson, and Jiayuan Sui. The Effectiveness of Receipt-based Attacks on ThreeBallot. *IEEE Transactions on Information Forensics and Security*, 4(4):699–707, 2009.

[15] H.L. Jonker and W. Pieters. Receipt-Freeness as a special case of Anonymity in Epistemic Logic. In *IAVoSS Workshop On Trustworthy Elections (WOTE 2006)*, 2006.

[16] A. Juels, D. Catalano, and M. Jakobsson. Coercion-resistant Electronic Elections. In *Proceedings of Workshop on Privacy in the Eletronic Society (WPES 2005)*, pages 61–70. ACM Press, 2005.

[17] R. Küsters. Simulation-Based Security with Inexhaustible Interactive Turing Machines. In *Proceedings of the 19th IEEE Computer Security Foundations Workshop (CSFW-19 2006)*, pages 309–320. IEEE Computer Society, 2006.

[18] Ralf Küsters, Tomasz Truderung, and Andreas Vogt. A Game-based Definition of Coercion-Resistance and its Applications. In *23th IEEE Computer Security Foundations Symposium, CSF 2010*, pages 122–136. IEEE Computer Society, 2010.

[19] Ralf Küsters, Tomasz Truderung, and Andreas Vogt. Accountability: Definition and Relationship to Verifiability. In *Proceedings of the 17th ACM Conference on Computer and Communications Security (CCS 2010)*, pages 526–535. ACM, 2010.

[20] Ralf Küsters, Tomasz Truderung, and Andreas Vogt. Proving Coercion-Resistance of Scantegrity II. In Miguel Soriano, Sihan Qing, and Javier López, editors, *Proceedings of the 12th International Conference on Information and Communications Security (ICICS 2010)*, volume 6476 of *Lecture Notes in Computer Science*, pages 281–295. Springer, 2010.

[21] Ralf Küsters and Tromasz Truderung. An Epistemic Approach to Coercion-Resistance for Electronic Voting Protocols. In *2009 IEEE Symposium on Security and Privacy (S&P 2009)*, pages 251–266. IEEE Computer Society, 2009.

[22] T. Moran and M. Naor. Receipt-Free Universally-Verifiable Voting With Everlasting Privacy. In C. Dwork, editor, *Advances in Cryptology - CRYPTO 2006, 26th Annual International Cryptology Conference, Proceedings*, volume 4117 of *Lecture Notes in Computer Science*, pages 373–392. Springer, 2006.

[23] T. Moran and M. Naor. Split-ballot voting: everlasting privacy with distributed trust. In P. Ning, S. De Capitani di Vimercati, and P. F. Syverson, editors, *Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007*, pages 246–255. ACM, 2007.

[24] T. Okamoto. Receipt-Free Electronic Voting Schemes for Large Scale Elections. In B. Christianson, B. Crispo, T. M. A. Lomas, and M. Roe, editors, *Proceedings of the 5th International Workshop on Security Protocols*, volume 1361 of *Lecture Notes in Computer Science*, pages 25–35. Springer, 1997.

[25] R. L. Rivest and W. D. Smith. Three Voting Protocols: ThreeBallot, VAV and Twin. In *USENIX/ACCURATE Electronic Voting Technology (EVT 2007)*, 2007.

[26] Ron Rivest. The ThreeBallot Voting System, October 1, 2006. `http://people.csail.mit.edu/rivest/Rivest-TheThreeBallotVotingSystem.pdf`.

[27] V. Teague, K. Ramchen, and L. Naish. Coercion-Resistant tallying for STV voting. In *IAVoSS Workshop On Trustworthy Elections (WOTE 2008)*, 2008.

[28] Dominique Unruh and Jörn Müller-Quade. Universally Composable Incoercibility. In *Advances in Cryptology (Crypto 2010)*, volume 6223 of *Lecture Notes in Computer Science*, pages 411–428. Springer, 2010.


# A Proof of Theorem 1

Our goal is to prove that $S = \mathsf{P}_{\mathsf{ideal}}(k,m,n,\vec{p})$ is $\delta$-coercion-resistant w.r.t. $\tilde{V}_i$, where $i \in \{1,\ldots,k\}$ and $\delta = \delta^i_{min}(n,k,\vec{p})$. To show $\delta$-coercion resistance, we take the counter-strategy $\tilde{v}$ which, when the coerced voter is instructed to vote (for some candidate), votes for the $i$-th candidate. Clearly, when running $\tilde{v}$, the coerced voter either abstains from voting or votes for the $i$-th candidate. Therefore, $\tilde{v} \in \tilde{V}_i$. Hence, it only remains to be shown that Condition (1) of Definition 1 is satisfied. We begin with some auxiliary definitions and facts. Let

$$\Delta_{ij} = \sum_{\vec{r} \in M^*_{i,j}} (A^j_{\vec{r}} - A^i_{\vec{r}}). \tag{9}$$

So, we have $\delta^i_{min}(n,k,\vec{p}) = \max_{j\in\{1,\ldots,k\}} \Delta_{ij}$.

By $\mathrm{res}(\omega_1, i)$, where $\omega_1 \in \Omega_1$ (recall that $\Omega_1$ is the set of candidate choices made by honest voters) and $i \in \{1,\ldots,k\}$, we denote the pure result of the election (i.e., an element of *Res*) obtained when the honest voters vote according to $\omega_1$ and the coerced voter $\mathsf{v}_0$ votes for $i$. Therefore, we have

$$A^i_{\vec{r}} = \Pr_{\omega_1}[res(\omega_1, i) = \vec{r}]. \tag{10}$$

By definition of $M^*_{i,j}$, it is easy to see that for every $i, j \in \{1,\ldots,k\}$ and every set $M \subseteq Res$ of pure results, the following inequality holds:

$$\sum_{\vec{r}\in M} \left(A^j_{\vec{r}} - A^i_{\vec{r}}\right) \leq \sum_{\vec{r}\in M^*_{i,j}} \left(A^j_{\vec{r}} - A^i_{\vec{r}}\right) = \Delta_{ij}. \tag{11}$$

Now, to prove Condition (1) of Definition 1, let $c \in C_S$. Recall that the view of the coercer in a run of the system consists only of his random coins $\omega_2 \in \Omega_2$ and the result of the election.

The only action of the coerced voter, besides receiving messages from the coercer, is to indicate the candidate of choice to the voting machine. Therefore, the dummy strategy of the coerced voter also only needs to forward one message, namely the candidate name; no other message is going to be accepted by the voting machine. Moreover, this message has to be sent before the result of the election is published, in order for the voting machine to accept the message. Therefore, if the coercer demands that the coerced voter votes for candidate $j$, he has to do this before the result is published. In particular, the coercer has to determine $j$—the candidate he wants the coerced voter to vote for—based solely on his random coins $\omega_2$, independently of the result of the election. Hence $j$ is a function of $\omega_2$, which we denote by $f(\omega_2)$; we take $f(\omega_2) = 0$ if the coercer does not want the coerced voter to vote. Note that $f(\omega_2) \in \{0, \ldots, k\}$. Hence, if the coerced voter runs the dummy strategy $\mathsf{dum}$, the view of the coercer is the random variable $\mathit{view}(\mathsf{dum}, c)$ defined by the equation

$$\mathit{view}(\mathsf{dum}, c)(\omega_1, \omega_2) = (\omega_2, \mathrm{res}(\omega_1, f(\omega_2)))$$

for every $\omega = (\omega_1, \omega_2) \in \Omega$.

We define $\Omega_2' = \{\omega_2 \in \Omega_2 \mid f(\omega_2) \neq 0\}$. Let $g(\omega_2) = i$ if $\omega_2 \in \Omega_2'$ and $g(\omega_2) = 0$ if $\omega_2 \notin \Omega_2'$. Note that $g(\omega_2)$ is the choice made by the coerced voter, according to her counter-strategy $\tilde{v}$ in a run with $\omega_2$. Therefore, if the coerced voter runs the counter-strategy $\tilde{v}$, the view of the coercer is:

$$\mathit{view}(\tilde{v}, c)(\omega_1, \omega_2) = (\omega_2, \mathit{res}(\omega_1, g(\omega_2)))\,.$$

Note that for $\omega_2 \notin \Omega_2'$, the counter-strategy behaves exactly like the dummy strategy, namely, abstains from voting, and therefore we have $\mathit{view}(\mathsf{dum}, c)(\omega_1, \omega_2) = \mathit{view}(\tilde{v}, c)(\omega_1, \omega_2)$ for all $\omega_1 \in \Omega_1$.

Now, let $M_c$ be the set of views accepted by the machine $c$. Each element of $M_c$ is of the form $(\omega_2, \vec{r})$, where $\omega_2 \in \Omega_2$ and $\vec{r}$ is a pure result. For $\omega_2 \in \Omega_2$, we define $M_c^{\omega_2}$

to be $\{\vec{r} \in Res : (\omega_2, r) \in M_c\}$. With this, we obtain:

$$\begin{aligned}
&\mathsf{Pr}[(c \parallel \mathsf{dum} \parallel \mathsf{e}_S) \mapsto 1] - \mathsf{Pr}[(c \parallel \tilde{v} \parallel \mathsf{e}_S) \mapsto 1] = \\
&\quad = \mathsf{Pr}[view(\mathsf{dum}, c) \in M_c] - \mathsf{Pr}[view(\tilde{v}, c) \in M_c] \\
&\quad = \sum_{\omega_2 \in \Omega_2'} \sum_{\vec{r} \in M_c^{\omega_2}} \Big( \mathsf{Pr}[view(\mathsf{dum}, c) = (\omega_2, \vec{r})] - \\
&\qquad\qquad - \mathsf{Pr}[view(\tilde{v}, c) = (\omega_2, \vec{r})] \Big) \\
&\quad = \sum_{\omega_2 \in \Omega_2'} \sum_{\vec{r} \in M_c^{\omega_2}} \Big( \mu_2(\omega_2) \cdot \mathsf{Pr}_{\omega_1}[\mathrm{res}(\omega_1, f(\omega_2)) = \vec{r}] \\
&\qquad\qquad - \mu_2(\omega_2) \cdot \mathsf{Pr}_{\omega_1}[\mathrm{res}(\omega_1, i) = \vec{r}] \Big) \\
&\quad = \sum_{\omega_2 \in \Omega_2'} \mu_2(\omega_2) \cdot \sum_{\vec{r} \in M_c^{\omega_2}} \Big( A_{\vec{r}}^{f(\omega_2)} - A_{\vec{r}}^{i} \Big) \\
&\quad \leq \sum_{\omega_2 \in \Omega_2'} \mu_2(\omega_2) \cdot \sum_{\vec{r} \in M^*_{i, f(\omega_2)}} \Big( A_{\vec{r}}^{f(\omega_2)} - A_{\vec{r}}^{i} \Big) && \text{(by (11))} \\
&\quad = \sum_{\omega_2 \in \Omega_2'} \mu_2(\omega_2) \cdot \Delta_{i, f(\omega_2)} && \text{(by (9))} \\
&\quad \leq \sum_{\omega_2 \in \Omega_2'} \mu_2(\omega_2) \cdot \max_{j \in \{1, \ldots, k\}} \Delta_{i,j} \\
&\quad = \max_{j \in \{1, \ldots, k\}} \Delta_{i,j} = \delta^i_{min}(n, k, \vec{p}).
\end{aligned}$$

This implies that $\mathsf{Pr}[(c \parallel \mathsf{dum} \parallel \mathsf{e}_S) \mapsto 1] - \mathsf{Pr}[(c \parallel \tilde{v} \parallel \mathsf{e}_S) \mapsto 1]$ is $\delta$-bounded, for $\delta = \delta^i_{min}(n, k, \vec{p})$. So, Condition (1) of Definition 1 follows.

It remains to show that $\delta$ is optimal. First, we observe that in the above inequalities we obtain equality for a coercer program $c = c_0$ which (i) always instructs the coerced voter to vote for candidate $j_0$, where, for the fixed $i$, $\Delta_{i,j}$ takes its maximum for $j = j_0$, and hence, $f(\omega_2) = j_0$ for all $\omega_2 \in \Omega_2$, and (ii) accepts a run only if the pure result $\vec{r}$ in his view belongs to $M^*_{i,j_0}$. Hence

$$\mathsf{Pr}[(c_0 \parallel \mathsf{dum} \parallel \mathsf{e}_S) \mapsto 1] - \mathsf{Pr}[(c_0 \parallel \tilde{v} \parallel \mathsf{e}_S) \mapsto 1] = \delta \tag{12}$$

and, therefore, $\delta$ is optimal for the counter-strategy $\tilde{v}$ we have considered. To complete the proof we only need to show that for no other counter-strategy $\tilde{v}' \in \tilde{V}_i$ one can obtain a smaller delta.

Let us observe that, if the coercer wants the coerced voter to vote, any counter-strategy $\tilde{v}'$ in $\tilde{V}_i$ has to vote for $i$ with overwhelming probability, by definition of $\tilde{V}_i$. Therefore, the systems $(c_0 \parallel \tilde{v} \parallel \mathsf{e}_S)$ and $(c_0 \parallel \tilde{v}' \parallel \mathsf{e}_S)$, where $c_0$ is defined as above, coincide with overwhelming probability. Hence, having (12), we conclude that the expression $\mathsf{Pr}[(c_0 \parallel \mathsf{dum} \parallel \mathsf{e}_S) \mapsto 1] - \mathsf{Pr}[(c_0 \parallel \tilde{v}' \parallel \mathsf{e}_S) \mapsto 1]$ may be at most negligibly smaller than $\delta$. This means that there is no constant $\delta'$ such that $\delta' < \delta$ and $S$ is $\delta'$-coercion-resistant with respect to $\tilde{V}_i$.

## B Modeling of the Bingo Voting Protocol

Our modeling of the Bingo Voting system is, as already mentioned in Section 2.1, based on the IITM model [17]. In the IITM model *inexhaustible interactive Turing machines (IITMs)* communicate via tapes. In a system of IITMs only one IITM is active at a time. Such a machine may perform computations polynomially bounded in the security parameter (and the length of the input received so far). It may send a message to another IITM (to which it is connected via a tape). This machine is then triggered. We assume that, in every system, there is one special IITM, called the master, which is triggered first. This machine is also triggered if, at some point, no message is sent by other machines. The systems of IITMs we consider are such that the length of every run is polynomially bounded in the length of the security parameter. We refer the reader to [17] for details of the IITM model. However, these details are not essential to understand the modeling of the Bingo Voting system.

In $\mathsf{P}_{\mathsf{Bingo}}(k,m,n,\vec{p})$ the set $\Sigma$ of protocol participants consists of the coerced voter $v_0$, the honest voters $v_1,\dots,v_n$, the voting machine $M$, the random number generator $RNG$, the coercer $c$ who subsumes all parties that are not assumed to be honest: dishonest voters, auditors, and the bulletin board. We assume also an additional participant, *the scheduler*, which is the master IITM in the system. The role of the scheduler is to make sure that every party gets a chance to perform some actions in every protocol phase.

**Channels.** The set of tapes, also called channels, we consider here includes the tapes/channels $\mathsf{ch}_b^a$, for every $a,b \in \Sigma \setminus \{RNG\}$. The channel $\mathsf{ch}_b^a$ is an output channel of $a$ and an input channel of $b$. Therefore, $a$ and $b$ can communicate using $\mathsf{ch}_b^a$ and $\mathsf{ch}_a^b$. Furthermore, the $RNG$ is connected to the machine $M$ by channels $\mathsf{ch}_{RNG}^M$ and $\mathsf{ch}_M^{RNG}$; the RNG is, however, not connected to other parties.

For simplicity, because we assume that $M$ is honest, we do not (have to) model direct communication between the RNG and a voter who is present in the voting booth at a given time. Instead, the number generated by the RNG is sent to $M$ and forwarded by $M$ to the voter, as described below.

**The Scheduler.** As already mentioned, in every instance of the protocol (which formally is a system of IITMs), the scheduler is the master IITM. The role of the scheduler is to trigger every party so that it is given a chance to perform the actions required by the protocol. The program of $\mathscr{S}$ is as follows:

- $\mathscr{S}$ first triggers the coercer. When the coercer stops, $\mathscr{S}$ triggers the voting machine $M$ which posts some setup parameters (parameters for the cryptographic primitives, i.e., for the commitment scheme) and the commitments, as specified by the protocol. Because the bulletin board is subsumed by the coercer, these messages are sent directly to him. The coercer, who subsumes also the auditors, is now supposed to reply with challenges for the zero-knowledge proof in the initialization phase. If the coercer does not provide $M$ with the challenges, the program of $M$ halts; otherwise, $M$ posts the required zero-knowledge proof, as specified by the protocol, by sending them back to the coercer.

- Then, $\mathscr{S}$ starts the voting phase. For this, $\mathscr{S}$ triggers the coercer and waits for the coercer to tell it which voter (honest, dishonest or coerced) to trigger. If asked by the coercer to trigger a specific voter, $\mathscr{S}$ triggers that voter by sending a message on the corresponding channel, if this voter was not triggered already. (Note that in case of a dishonest voter, the coercer is triggered by $\mathscr{S}$ since the coercer is connected to the corresponding channel.) The scheduler $\mathscr{S}$ will not move to the next stage until all voters have been triggered once.

- Once all voters have been triggered, $\mathscr{S}$ starts the tallying phase by triggering the voting machine $M$, which (1) publishes all the receipts of voters together with the identities of the voters and (2) opens the unused commitments, by sending the respective information to the coercer (representing the bulleting board). Analogously to the first zero-knowledge proof, the coercer then should provide $M$ with challenges for the zero-knowledge proof, in which case $M$ posts the required proofs.

  Note that by sending to the coercer the receipts with identities of the voters, we model not only that the coercer can get hold of those receipts, but also that the voters, including the coerced voter, cannot fake receipts.

**Programs of honest participants and the coerced voter.** The IITMs of honest participants (except for the IITM of the scheduler, which is given above) are defined according to the description of the Bingo voting system in Section 5. In particular, if the IITM of an honest voter is triggered by the scheduler, then it makes a choice according to $\vec{p}$. If the choice is to abstain from voting, the IITM stops and otherwise sends the chosen candidate to $M$. Then, $M$ triggers the RNG, which replies with a fresh random number. The machine $M$ then sends the resulting receipt back to the voter. The machine $M$ behaves in the same way when triggered by dishonest voters or the coerced voter. The coerced voter either implements $\mathsf{dum}$ or the counter-strategy.

**View of the Coercer.** We now provide a detailed description of the view of the coercer in a run of the system $\mathsf{P}_{\mathsf{Bingo}}(k,m,n,\vec{p})$. This is needed in our proofs, in particular for the proof of Lemma 6. For simplicity of presentation, we omit the description of the zero-knowledge proofs in the initialization phase. However, our proofs can easily be extended to deal with these zero-knowledge proofs in a similar manner as with the zero-knowledge proofs for the tallying phase, which we handle.

We start with introducing notation for the *cryptographic components* used in the protocol. In the following, by $\mathsf{comm}(a)^r$ we denote the commitment on $a$ with randomness $r$. We first describe in detail the structure of the sequence $\omega_2 \in \Omega_2$ of random coins.

(a) $\alpha$ — the random coins of the coercer.

(b) $\beta$ — the randomness used by the machine to determine the setup parameters.

(c) $x_i^j$ and $r_i^j$, for $i \in \{0,\dots,m\}$ and $j \in \{1,\dots,k\}$ — the random numbers and the randomness used in the commitments $c_i^j = \mathsf{comm}(j,x_i^j)^{r_i^j}$ in the initialization phase.

(d) $\pi$ — the permutation used by the machine to shuffle the commitments $c_i^j$.

(e) $x_i$, for $i \in \{0,\dots,m\}$ — the random number generated by the RNG for the $i$-th voter.

(f) $\pi_j$, for every candidate $j \in \{1,\dots,k\}$ — a permutation of $\{0,\dots,m\}$, such that $x^j_{\pi_j(i)}$ is the number (taken from the pool of random numbers generated for the $j$-th candidate) assigned by the machine in the voting booth to the $i$-th voter (if necessary, that is, if the $i$-th voter does not abstain and does not vote for $j$).

(g) $r_i$, for every candidate $i \in \{0,\dots,m\}$ who does not abstain — a random number used by the voting machine to create a commitment $c_i = \mathsf{comm}(v_i, x_i)^{r_i}$ in the zero-knowledge proof in the tallying phase.

(h) $\sigma^0_i$, for every candidate $i \in \{0,\dots,m\}$ who does not abstain — a permutation of $\{1,\dots,k\}$ used by the machine to shuffle the commitments associated with the receipt $R_i$ of the $i$-th voter (see (B4) below).

(i) $\tau^1_{i,j}$ and $\sigma^1_i$, for every $j \in \{1,\dots,k\}$ and every candidate $i \in \{0,\dots,m\}$ who does not abstain — random numbers and permutations used for masking and shuffling commitments in $C^i_{left}$ (see (B6) below).

(j) $\tau^2_{i,j}$ and $\sigma^2_i$, for every $j \in \{1,\dots,k\}$ and every candidate $i \in \{0,\dots,m\}$ who does not abstain — random numbers and permutations used for masking and shuffling commitments in $C^i_{middle}$ (see (B7) below).

The *view* $\rho$ *of the coercer*, depending on $\omega_2$ and the choices $v_0,\dots,v_n$ taken by the voters, consists of the following components (or some subset of these components, if the protocol cannot be finished for some reason, for instance, if the coercer does not provides the challenges required for the zero-knowledge proofs):

(B1) $\alpha$ — random coins of the coercer.

(B2) $Par(\beta)$ — the setup parameters.

(B3) The commitments $c^j_i$ shuffled with $\pi$.

(B4) $R_i$ — the receipt of the $i$-th voter, for every non-abstaining voter $i$. Such a receipt is of the form $s_1,\dots,s_k$, where $s_j = (j, x^j_{\pi_j(i)})$, for $j \neq v_i$, and $s_{v_i} = (v_i, x_i)$.

(B5) The values $x^j_{\pi_j(i)}$ and $r^j_{\pi_j(i)}$ for opening the unused commitments $c^j_{\pi_j(i)}$, for all $j \in \{0,\dots,k\}$ and $i \in \{0,\dots,m\}$ such that $v_i = 0$ or $v_i = j$.

In the following items, $i$ ranges over all the non-abstaining voters $i \in \{0,\dots,m\}$. Also, by $\hat{\tau}^1_{i,j}$ and $\hat{\tau}^2_{i,j}$ we denote the random numbers used to masking commitments related to the $i$-th voter and the $j$-th candidate, that is:

$$\hat{\tau}^1_{i,j} = \tau^1_{i,(\sigma^0_i)^{-1}(j)} \qquad \text{and} \qquad \hat{\tau}^2_{i,j} = \tau^2_{i,(\sigma^1_i)^{-1}((\sigma^0_i)^{-1}(j))}.$$

(B6) The list of commitments $C^i_{left} = d^1_i,\dots,d^k_i$ shuffled with $\sigma^0_i$, where $d^j_i = c_i$, if $j = v_i$, and $d^j_i = c^j_{\pi_j(i)}$, otherwise,

(B7) The list of commitments $C^i_{middle} = \bar{d}^1_i,\dots,\bar{d}^k_i$ shuffled with $(\sigma^1_i \circ \sigma^0_i)$, where $\bar{d}^j_i = \mathsf{comm}(v_i,x_i)^{r_i+\hat{\tau}^1_{i,j}}$, if $j = v_i$, and $\bar{d}^j_i = \mathsf{comm}(j, x^j_{\pi_j(i)})^{r^j_{\pi_j(i)}+\hat{\tau}^1_{i,j}}$, otherwise.

(B8) The list of commitments $C^i_{right} = \hat{d}^1_i, \ldots, \hat{d}^k_i$ shuffled with $(\sigma^2_i \circ \sigma^1_i \circ \sigma^0_i)$, where $\hat{d}^j_i = \mathsf{comm}(v_i, x_i)^{r_i + \hat{\tau}^1_{i,j} + \hat{\tau}^2_{i,j}}$, if $j = v_i$, and $\hat{d}^j_i = \mathsf{comm}(j, x^j_{\pi_j(i)})^{r^j_{\pi_j(i)} + \hat{\tau}^1_{i,j} + \hat{\tau}^2_{i,j}}$, otherwise.

(B9) The values $r_i + \hat{\tau}^1_{i,j} + \hat{\tau}^2_{i,j}$ and $r^j_{\pi_j(i)} + \hat{\tau}^1_{i,j} + \hat{\tau}^2_{i,j}$, for $j \in \{1, \ldots, k\}$, $j \neq v_i$, for opening the commitment in $C^i_{right}$.

(B10) The masking factors $\tau^s_{i,j}$ and permutations $\sigma^s_i$, where $s$ denotes the challenge the coercer creates for the zero-knowledge proof, depending on (B1)–(B9).

Note that the coercer gets to see the receipt of the coerced voter (if any) — in fact, the receipts of all voters — and that the number shown by the RNG to the coerced voter is printed on her receipt. We therefore do not need to include this number explicitly in the view of the coercer.

## C Proof of Lemma 2

The core of Lemma 2 is stated in the following fact.

**Lemma 5.** *Let $\rho$ be an arbitrary coercer view such that $f(\rho)$ is defined. Let $\omega_1, \omega_1', \omega_1'', \omega_1'''$ be some elements of $\Omega_1$ with $\varphi_\rho(\omega_1), \varphi_\rho(\omega_1')$, $\tilde{\varphi}_\rho(\omega_1'')$, and $\tilde{\varphi}_\rho(\omega_1''')$. Then the sets*

$$A = \{\omega_2 : T(\omega_1, \omega_2) \mapsto \rho\}, \qquad C = \{\omega_2 : \tilde{T}(\omega_1'', \omega_2) \mapsto \rho\},$$
$$B = \{\omega_2 : T(\omega_1', \omega_2) \mapsto \rho\}, \qquad D = \{\omega_2 : \tilde{T}(\omega_1''', \omega_2) \mapsto \rho\}.$$

*have the same cardinality, and hence, $\mu_2(A) = \mu_2(B) = \mu_2(C) = \mu_2(D)$.*

Before we prove this lemma, we show that it implies Lemma 2. Let $\rho$, $\omega_1^\rho$ and $\tilde{\omega}_1^\rho$ be as in Lemma 2. Let us observe that, for all $\omega_1$ and $\omega_2$, if $T(\omega_1, \omega_2) \mapsto \rho$, then $\varphi_\rho(\omega_1)$. Using this observation and Lemma 5 we obtain

$$\begin{aligned}
\mathsf{Pr}[T \mapsto \rho] &= \mathsf{Pr}_{\omega_1, \omega_2}[\varphi_\rho(\omega_1), T(\omega_1, \omega_2) \mapsto \rho] \\
&= \sum_{\omega_1' : \varphi_\rho(\omega_1')} \mathsf{Pr}_{\omega_1, \omega_2}[\omega_1 = \omega_1', T(\omega_1', \omega_2) \mapsto \rho] \\
&= \sum_{\omega_1' : \varphi_\rho(\omega_1)} \mu_1(\omega_1') \cdot \mathsf{Pr}_{\omega_2}[T(\omega_1', \omega_2) \mapsto \rho] \\
&= \sum_{\omega_1' : \varphi_\rho(\omega_1)} \mu_1(\omega_1') \cdot \mathsf{Pr}_{\omega_2}[T(\omega_1^\rho, \omega_2) \mapsto \rho] \\
&= \mathsf{Pr}_{\omega_1}\left[\varphi_\rho(\omega_1)\right] \cdot \mathsf{Pr}_{\omega_2}\left[T(\omega_1^\rho, \omega_2) \mapsto \rho\right].
\end{aligned}$$

This proves (3). The proof for (4) is analogous. Statement (5) follows immediately from Lemma 5.

**Proof of Lemma 5.** In the proof of Lemma 5 we will use the following result. By $\tilde{T}_j$ we denote the system $(c \parallel \tilde{v}_j \parallel \mathsf{e}_S)$, where $\tilde{v}_j$ is defined just like $\tilde{v}$ but votes for $j$ instead of $i$. So we have $\tilde{v} = \tilde{v}_i$ and $\tilde{T} = \tilde{T}_i$. Moreover, for each view $\rho$ of the coercer, for which $f(\rho)$ is defined, we clearly have: $T(\omega_1, \omega_2) \mapsto \rho$ iff $\tilde{T}_{f(\rho)}(\omega_1, \omega_2) \mapsto \rho$.

A *permutation* $\sigma$ *on a tuple* $\vec{v} = (v_0, \ldots, v_n) \in \{0, 1, \ldots, k\}^{n+1}$ is a permutation on the set of indices $\{0, \ldots, n\}$. We write $\sigma(\vec{v})$ for the tuple $(v_{\sigma(0)}, \ldots, v_{\sigma(n)})$. We will also write $\sigma(\vec{v})[i]$ for $v_{\sigma(i)}$ and $\sigma(\vec{v})[i..j]$ for the tuple $(v_{\sigma(i)}, \ldots, v_{\sigma(j)})$. We say that $\sigma$ does not change the abstaining votes of $\vec{v}$ if $\sigma(j) = j$ for every $j \in \{0, \ldots, k\}$ with $v_j = 0$. For $j \in \{1, \ldots, k\}$ and $\omega_1 \in \Omega_1 (= \{0, 1, \ldots, k\}^n)$, we consider $(j, \omega_1)$ to be a $(n+1)$-tuple over $\{0, 1, \ldots, k\}$.

**Lemma 6.** *For every* $j \in \{1, \ldots, k\}$*,* $\omega_1 \in \Omega_1$ *and every permutation* $\sigma$ *on* $\vec{v} = (j, \omega_1)$ *that does not change the abstaining votes, there is a bijective function* $h = h^{j, \omega_1, \sigma}$ *from* $\Omega_2$ *to* $\Omega_2$ *such that for all* $\omega_2$ *we have that* $\tilde{T}_j(\omega_1, \omega_2)$ *yields the same view as* $\tilde{T}_{\sigma(\vec{v})[0]}(\sigma(\vec{v})[1..n], h(\omega_2))$*.*

We postpone the proof of this lemma to the end of this section. Now, we show that Lemma 5 follows directly from Lemma 6: Given the assumptions of Lemma 5, there are permutations $\sigma_1$, $\sigma_2$, and $\sigma_3$ that do not change the abstaining votes such that $(f(\rho), \omega_1) = \sigma_1(f(\rho), \omega_1') = \sigma_2(i, \omega_1'') = \sigma_3(i, \omega_1''')$. Moreover, $T(\omega_1, \omega_2) \mapsto \rho$ iff $\tilde{T}_{f(\rho)}(\omega_1, \omega_2) \mapsto \rho$ and $\tilde{T}(\omega_1, \omega_2) \mapsto \rho$ iff $\tilde{T}_i(\omega_1, \omega_2) \mapsto \rho$. From this and Lemma 6 we obtain that the functions $h^{f(\rho), \omega_1, (\sigma_1)^{-1}}$, $h^{f(\rho), \omega_1, (\sigma_2)^{-1}}$, and $h^{f(\rho), \omega_1, (\sigma_3)^{-1}}$ are bijections between $A$ and $B$, $A$ and $C$, and $A$ and $D$, respectively.

**Proof of Lemma 6.** Because every permutation is the finite composition of permutations that switch only two successive positions, it suffices to consider the case where $\sigma^0$ flips the positions $l$ and $l+1$; the rest follows from composing permutations and bijections. Let $(v_0, \ldots, v_m) = \vec{v} = (j, \omega_1)$. Let $\tilde{v}_0, \ldots, \tilde{v}_n$ be such that

$$\sigma^0(v_0, \ldots, v_n) = (\tilde{v}_0, \ldots, \tilde{v}_n) = (v_0, \ldots, v_{l+1}, v_l, \ldots, v_n).$$

Further, we assume that $v_l = y \neq z = v_{l+1}$, as the case that $\sigma^0(v_0, \ldots, v_n) = (v_0, \ldots, v_n)$ is trivial. Recall that, by assumption, we have that $y, z \neq 0$.

Let $\omega_2$ be any element of $\Omega_2$ and let $\alpha$, $\beta$, $x_i^j$, $r_i^j$, $\pi$, $x_i$, $\pi_j$, $r_i$, $\sigma_i^0$, $\tau_{i,j}^1$, $\sigma_i^1$, $\tau_{i,j}^2$ and $\sigma_i^2$ be the components of $\omega_2$ defined as above. Here, $i$ ranges over $0, \ldots, m$ and $j$ over $1, \ldots, k$. We will denote the corresponding components of $h(\omega_2)$ by $\tilde{\alpha}$, $\tilde{\beta}$, $\tilde{x}_i^j$, and so on. We define $h(\omega_2)$ as follows:

- $\tilde{\alpha} = \alpha$ and $\tilde{\beta} = \beta$. As one can see, (B1) and (B2) remain unchanged.
- $\tilde{\pi}_j = \pi_j$, for all $j$.
- $\tilde{x}_i^j$ are defined like $x_i^j$, except for:

$$\tilde{x}_l = x^z_{\pi_z(l)}, \qquad \tilde{x}^y_{\pi_y(l)} = x_l, \tag{13}$$

$$\tilde{x}_{l+1} = x^y_{\pi_y(l+1)}, \qquad \tilde{x}^z_{\pi_z(l+1)} = x_{l+1}, \tag{14}$$

$$\tilde{x}^z_{\pi_z(l)} = x^z_{\pi_z(l+1)}, \qquad \tilde{x}^y_{\pi_y(l+1)} = x^y_{\pi_y(l)}. \tag{15}$$

One can check that, by (13) and (14), the receipts (B4) remain unchanged.

- $\tilde{r}_i^j$ are defined like $r_i^j$, except for $\tilde{r}^z_{\pi_z(l)} = r^z_{\pi_z(l+1)}$ and $\tilde{r}^y_{\pi_y(l+1)} = r^y_{\pi_y(l)}$ (which, together with (15) implies that (B5) remains unchanged) and, furthermore, $\tilde{r}^y_{\pi_y(l)}$ and $\tilde{r}^z_{\pi_z(l+1)}$ are (uniquely) defined in such a way that

$$\mathsf{comm}(y,\tilde{x}^y_{\pi_y(l)})^{\tilde{r}^y_{\pi_y(l)}} = \mathsf{comm}(z,x^z_{\pi_z(l)})^{r^z_{\pi_z(l)}} \tag{16}$$

$$\mathsf{comm}(z,\tilde{x}^z_{\pi_z(l+1)})^{\tilde{r}^z_{\pi_z(l+1)}} = \mathsf{comm}(y,x^y_{\pi_y(l+1)})^{r^y_{\pi_y(l+1)}}. \tag{17}$$

  (Note that Pedersen commitments used in this protocol guarantee that for each $a$ and $b$ there exists exactly one $r$ such that $\mathsf{comm}(a)^r = b$.)
- $\tilde{\pi}$ is as $\pi$ with the straightforward adjustment such that the list of published commitments (B3) in both cases (i.e., for $\omega_2$ and $h(\omega_2)$ is exactly the same (it can be easily done, because, as one can check, the produced commitments in both cases are, up to the ordering, the same).
- $\tilde{r}_i$ are like $r_i$ with the two following exceptions: $\tilde{r}_l$ and $\tilde{r}_{l+1}$ are (uniquely) defined in such a way that

$$\mathsf{comm}(z,\tilde{x}_l)^{\tilde{r}_l} = \mathsf{comm}(y,x_l)^{r_l}$$
$$\mathsf{comm}(y,\tilde{x}_{l+1})^{\tilde{r}_{l+1}} = \mathsf{comm}(z,x_{l+1})^{r_{l+1}}.$$

- $\tilde{\sigma}_i^0$ are like $\sigma_i^0$ with the following exceptions:

$$\tilde{\sigma}_l^0(y) = \sigma_l^0(z), \qquad \tilde{\sigma}_{l+1}^0(y) = \sigma_{l+1}^0(z),$$
$$\tilde{\sigma}_l^0(z) = \sigma_l^0(y), \qquad \tilde{\sigma}_{l+1}^0(z) = \sigma_{l+1}^0(y).$$

  One can verify that (B6) remains unchanged.

In the following, we assume that $s = 1$ (where $s$ is like in (B10)); the case for $s = 2$ is very similar.

- Let $\tilde{\sigma}_i^1 = \sigma_i^1$, and $\tilde{\tau}^1_{i,j} = \tau^1_{i,j}$ for all $i, j$. One can also check that (B7) remains the same (this is because (B6) remains the same and (B7) is obtained from it using the same permutations and masking factors).
- Let $\tilde{\sigma}_i^2$ be like $\sigma_i^2$ with the following exceptions:

$$\tilde{\sigma}_l^2(\tilde{\sigma}_l^1(\tilde{\sigma}_l^0(y))) = \sigma_l^2(\sigma_l^1(\sigma_l^0(y))) \qquad \text{and} \qquad \tilde{\sigma}_l^2(\tilde{\sigma}_l^1(\tilde{\sigma}_l^0(z))) = \sigma_l^2(\sigma_l^1(\sigma_l^0(z)))$$

  and analogously for $(l+1)$.
- Let $\tilde{\tau}^2_{i,j}$ be like $\tau^2_{i,j}$, except for $\tilde{\tau}^2_{l,z'}$ and $\tilde{\tau}^2_{l,y'}$, where $z' = (\tilde{\sigma}_l^1)^{-1}((\tilde{\sigma}_l^0)^{-1}(z))$ and $y' = (\tilde{\sigma}_l^1)^{-1}((\tilde{\sigma}_l^0)^{-1}(y))$, which are defined in such a way that

$$\tilde{r}_l + \tilde{\tau}^1_{l,(\tilde{\sigma}_l^0)^{-1}(z)} + \tilde{\tau}^2_{l,z'} = r^z_{\pi_z(l)} + \tau^1_{l,(\sigma_l^0)^{-1}(z)} + \tau^2_{l,(\sigma_l^1)^{-1}((\sigma_l^0)^{-1}(z))} \tag{18}$$

$$\tilde{r}^y_{\pi_y(l)} + \tilde{\tau}^1_{l,(\tilde{\sigma}_l^0)^{-1}(y)} + \tilde{\tau}^2_{l,y'} = r_l + \tau^1_{l,(\sigma_l^0)^{-1}(y)} + \tau^2_{l,(\sigma_l^1)^{-1}((\sigma_l^0)^{-1}(y))} \tag{19}$$

  and analogously for $(l+1)$. Now, one can check that (B8) and (B9) remain the same. As (B1)–(B9) remains the same, the challenge $s$ remains the same and therefore, (B10) remains the same.

This concludes the description of $h(\omega_2)$. As we have noted, all the parts (B1)–(B10) of the views for $\omega_2$ and $h(\omega_2)$ are exactly the same. What remains to be shown is that $h$ is a bijection from $\Omega_2$ to $\Omega_2$. To do this, it is enough to prove that $\omega_2$ can be uniquely determined by $\tilde{\omega}_2 = h(\omega_2)$. We only have to deal with those parts of $\omega_2$ that are changed by $h$. We consider those changed parts of $\omega_2$ case by case:

- It is easy to see that the numbers $x_i$ and $x_i^j$ (for $j \in \{1,\dots,k\}$ and $i \in \{0,\dots,m\}$) are uniquely determined by the numbers $\tilde{x}_i$ and $\tilde{x}_i^j$.
- $r^z_{\pi_z(l)}$ can be computed from $\tilde{\omega}_2$, as it is uniquely determined by the equality (16) (recall that $x^z_{\pi_z(l)}$, as we already stated, is determined by $\tilde{\omega}_2$). Analogously for $r^z_{\pi_z(l+1)}$, $r_l$, and $r_{l+1}$. Apart from these values, $r_i^j$ and $r_i$ coincide with $\tilde{r}_i^j$ and $\tilde{r}_i$, respectively, and therefore are determined by $\tilde{\omega}_2$.
- The permutations in $\tilde{\omega}_2$ are obtained from the corresponding permutation of $\omega_2$, by switching some selected positions. It is easy to define the inverse operation.
- If $s = 1$, then $\tau^2_{l,\sigma_l^1(\sigma_l^0(z))}$ is is uniquely determined by (18), as all the other parts in the equation are determined by $\tilde{\omega}_2$. (Note that $\tau^1_{l,\sigma_l^0(z)}$ is not changed if $s = 1$ and that $s$ is determined by $h(\omega_2)$). Analogously for $l+1$ and for the case $s = 2$.

# D Proofs for ThreeBallot with Two Candidates

In this section, we prove Theorem 3 and Lemma 4, where for Theorem 3 we only prove the second statement, i.e., the more involved case in which the coercer gets to see the receipts of the honest parties; the proof of the first statement is analogous and simpler.

We first introduce some notation. We will assume that the space $\Omega_1$ of all possible combinations of choices made by honest voters determines not only the candidates the voters have chosen, but also the way they vote, that is, the exact pattern (see Section 6.3 for the definition of a pattern). We define the following random variables on $\Omega_1$: $rec(\omega_1)$ denotes the vector $(r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}})$ of numbers of receipts of honest voters of the corresponding types and $Rec(\omega_1)$ is the vector $(r_1,\dots,r_n)$, where $r_i \in \{\substack{\times\\\times}, \substack{\times\\\circ}, \substack{\circ\\\times}, \substack{\circ\\\circ}\}$ is the receipt of the $i$-th honest voter (without a serial number),

## D.1 Proof of Theorem 3

First, we can represent an element $\omega_2$ of the space of random bits $\Omega_2$ used in a run of a system, in addition to the random choices $\omega_1$, as a tuple $\omega_2 = (\alpha, \vec{r}, \pi)$, where $\alpha$ is a sequence of random coins of the coercer, $\vec{r} = (r_{ij})_{i\in\{0,\dots,m\}, j\in\{1,2,3\}}$, where $r_{ij}$ is the serial number printed by the voting machine on the $j$-th ballot cast by the $i$-th voter (where the 0-th voter is the coerced voter), and $\pi$ is a permutation applied to the set of ballots before publishing. As usually, by $\mu_2$ we denote the uniform distribution on $\Omega_2$. (Note that a serial number $r_{ij}$, for $j \in \{1,2,3\}$, is not printed, if the $i$-th voter does not vote.)

A view of the coercer consists of (1) his random coins, (2) the content of the bulletin board, which is a sequence of simple ballots with serial numbers, and (3) the sequence of receipts (where, again, a receipt is a simple ballot with a serial number) associated to the voters. We will use letter $\eta$ to range over views of the coercer. Recall that $\rho$ is used to denote an *essential* view of the coercer.

By $\rho(\eta)$ we will denote the essential view determined by $\eta$. By $\rho(Z,\omega_1)$, for a pattern $Z$ and $\omega_1 \in \Omega_1$, we denote the essential view obtained when the coerced voter casts simple ballots according to $Z$ and the honest voters casts ballots determined by $\omega_1$. By $f(\eta)$ we denote the pattern that the coercer requires the coerced voter to use in a run $\eta$, if any; otherwise $f(\eta)$ is undefined. By $Rec(\eta)$ we denote the receipts *without serial numbers* that the honest voters give to the coercer in run $\eta$.

For a coercer view $\eta$, let $\varphi_\eta$ be a predicate over $\Omega_1$ such that $\varphi_\eta(\omega_1)$ iff $\rho(f(\eta),\omega_1) = \rho(\eta)$ and $Rec(\omega_1) = Rec(\eta)$. Analogously, we define a predicate $\tilde{\varphi}_\eta(\omega_1)$ which holds iff $\rho(C(f(\eta),i),\omega_1) = \rho(\eta)$ and $Rec(\omega_1) = Rec(\eta)$.

Let $c$ be a program of the coercer. Let $T = (c \parallel \mathsf{dum} \parallel \mathsf{e}_S)$ and $\tilde{T} = (c \parallel \tilde{v} \parallel \mathsf{e}_S)$.

We now show that the view of the coercer is information-theoretically independent of the choices of honest voters and the coerced voter as long as these choices are consistent with the essential view and the order of the receipts. This is formulated in Lemma 8, with the core stated in the following lemma.

**Lemma 7.** *Let $\eta$ be a view of the coercer such that $f(\eta)$ is defined. Let $\omega_1, \omega_1', \omega_1'', \omega_1'''$ be arbitrary elements of $\Omega_1$ with $\varphi_\eta(\omega_1)$, $\varphi_\eta(\omega_1')$, $\tilde{\varphi}_\eta(\omega_1'')$ and $\tilde{\varphi}_\eta(\omega_1''')$. Then the sets*

$$A = \{\omega_2 : T(\omega_1,\omega_2) \mapsto \eta\}, \qquad B = \{\omega_2 : T(\omega_1',\omega_2) \mapsto \eta\},$$
$$C = \{\omega_2 : \tilde{T}(\omega_1'',\omega_2) \mapsto \eta\}, \qquad D = \{\omega_2 : \tilde{T}(\omega_1''',\omega_2) \mapsto \eta\}$$

*have the same cardinality, and hence, have the same probability.*

*Proof.* We will show how to construct a bijection $h : A \to B$. The proof for the remaining cases are very similar.

Let $I = \{1,\ldots,n\} \times \{1,2,3\}$. For $(i,l) \in I$, by $b_{i,l} \in \{\substack{\times\\ \times}, \substack{\circ\\ \circ}, \substack{\times\\ \circ}, \substack{\circ\\ \times}\}$ and $b'_{i,l} \in \{\substack{\times\\ \times}, \substack{\circ\\ \circ}, \substack{\times\\ \circ}, \substack{\circ\\ \times}\}$ we denote the marking on the $l$-th ballot cast by the $i$-th voter according to $\omega_1$ and $\omega_1'$, respectively. Because $\varphi_\eta(\omega_1)$ and $\varphi_\eta(\omega_1')$, we know that there exists a permutation $\sigma : I \to I$ such that $b'_{(i,l)} = b_{\sigma(i,l)}$. Moreover, we can assume that $\sigma$ preserves receipts of honest voters, that is, if the $i$-th voter picks the $l$-th ballot as her receipt according to $\omega_1$ and she picks the $l'$-th ballot as a receipt according to $\omega_1'$, then $\sigma(i,l') = (i,l)$. Note that, in this case, $b_{(i,l)} = b'_{(i,l')}$.

Let $(\alpha,\vec{r},\pi) \in \omega_2$. We define $h(\alpha,\vec{r},\pi) = (\alpha,\vec{r}',\pi')$, where $r'_{(i,j)} = r_{\sigma(i,j)}$ and $\pi'(i,l) = \pi(\sigma(i,l))$ (recall that $\pi$ determines the position $\pi(i,j)$ of the ballot $b_{(i,j)}$ on the bulletin board). It is easy to check that $h$ is a bijection from $A$ to $B$. $\square$

**Lemma 8.** *Let $\eta$ be a coercer view such that $f(\eta)$ is defined. Let $\omega_1^\eta$ and $\tilde{\omega}_1^\eta$ be some fixed elements of $\Omega_1$ such that $\varphi_\eta(\omega_1^\eta)$ and $\tilde{\varphi}_\eta(\tilde{\omega}_1^\eta)$, respectively. Then, the following equations hold true:*

$$\mathsf{Pr}[T \mapsto \eta] = \Pr_{\omega_1}[\varphi_\eta(\omega_1)] \cdot \Pr_{\omega_2}[T(\omega_1^\eta, \omega_2) \mapsto \eta] \tag{20}$$

$$\mathsf{Pr}[\tilde{T} \mapsto \eta] = \Pr_{\omega_1}[\tilde{\varphi}_\eta(\omega_1)] \cdot \Pr_{\omega_2}[\tilde{T}(\tilde{\omega}_1^\eta, \omega_2) \mapsto \eta] \tag{21}$$

$$\Pr_{\omega_2}[T(\omega_1^\eta, \omega_2) \mapsto \eta] = \Pr_{\omega_2}[\tilde{T}(\tilde{\omega}_1^\eta, \omega_2) \mapsto \eta] \ . \tag{22}$$

*Proof.* Using Lemma 7 we obtain:

$$\begin{aligned}
\mathsf{Pr}[T \mapsto \eta] &= \mathsf{Pr}[\varphi_\eta(\omega_1), T(\omega_1, \omega_2) \mapsto \eta] \\
&= \sum_{\omega_1' : \varphi_\eta(\omega_1')} \mathsf{Pr}[\omega_1 = \omega_1', T(\omega_1', \omega_2) \mapsto \eta] \\
&= \sum_{\omega_1' : \varphi_\eta(\omega_1)} \Pr_{\omega_1}[\omega_1 = \omega_1'] \cdot \Pr_{\omega_2}[T(\omega_1', \omega_2) \mapsto \eta] \\
&= \sum_{\omega_1' : \varphi_\eta(\omega_1)} \Pr_{\omega_1}[\omega_1 = \omega_1'] \cdot \Pr_{\omega_2}[T(\omega_1^\eta, \omega_2) \mapsto \eta] \\
&= \Pr_{\omega_1}[\varphi_\eta(\omega_1)] \cdot \Pr_{\omega_2}\left[T(\omega_1^\eta, \omega_2) \mapsto \eta\right].
\end{aligned}$$

This proves (20); analogously, one can prove (21). Equation (22) follows directly from Lemma 7. □

Now, using Lemma 8, we can link the level of coercion resistance ThreeBallot provides with the optimal bound $\delta^i_{TB^+}$ stated in Section 6. Clearly we have:

$$\begin{aligned}
\Pr_{\omega_1}[\varphi_\eta(\omega_1)] &= \Pr_{\omega_1}[\rho(f(\eta), \omega_1) = \rho(\eta),\ Rec(\omega_1) = Rec(\eta)] \\
&= \Pr_{\omega_1}[\rho(f(\eta), \omega_1) = \rho(\eta)] \cdot \Pr_{\omega_1}[Rec(\omega_1) = Rec(\eta) \mid \rho(f(\eta), \omega_1) = \rho(\eta)] \\
&= A^{f(\eta)}_{\rho(\eta)} \cdot \Pr_{\omega_1}[Rec(\omega_1) = Rec(\eta) \mid \rho(f(\eta), \omega_1) = \rho(\eta)]
\end{aligned}$$

and, similarly,

$$\Pr_{\omega_1}[\tilde{\varphi}_\eta(\omega_1)] = A^{C(f(\eta),i)}_{\rho(\eta)} \cdot \Pr_{\omega_1}[Rec(\omega_1) = Rec(\eta) \mid \rho(C(f(\eta), i), \omega_1) = \rho(\eta)].$$

Furthermore, it is easy to show that given two essential views with the same number of receipts of every type (and otherwise possibly different information on the bulletin board), the probability of obtaining a specific vector of receipts (which links receipts and voters) stays the same. From this it follows:

$$\begin{aligned}
&\Pr_{\omega_1}[Rec(\omega_1) = Rec(\eta) \mid \rho(f(\eta), \omega_1) = \rho(\eta)] = \\
&\qquad = \Pr_{\omega_1}[Rec(\omega_1) = Rec(\eta) \mid \rho(C(f(\eta), i), \omega_1) = \rho(\eta)] \\
&\qquad = \Pr_{\omega_1}[Rec(\omega_1) = Rec(\eta) \mid rec(\omega_1) = rec(\eta)].
\end{aligned}$$

Together with Lemma 8, we immediately obtain for all $\eta$ such that $f(\eta)$ is defined and for all $\omega_1^\eta$ with $\varphi_\eta(\omega_1^\eta)$:

$$\Pr[T \mapsto \eta] - \Pr[\tilde{T} \mapsto \eta] = \left(A_{\rho(\eta)}^{f(\eta)} - A_{\rho(\eta)}^{C(f(\eta),i)}\right) \cdot \\ \cdot \Pr_{\omega_2}[T(\omega_1^\eta, \omega_2) \mapsto \eta] \cdot \Pr_{\omega_1}[Rec(\omega_1) = Rec(\eta) \mid rec(\omega_1) = rec(\eta)].$$

Note that if there does not exist $\tilde{\omega}_1^\eta$ such that $\tilde{\varphi}_\eta(\tilde{\omega}_1^\eta)$, then $A_{\rho(\eta)}^{C(f(\eta),i} = 0$ and $\Pr[\tilde{T} \mapsto \eta] = 0$.

Now, we are ready to prove that the system $S$, as defined in Theorem 3 in the case the coercer can see the receipts of honest voters, is $\delta$-coercion resistant w.r.t. $\tilde{V}_i$ for $\delta = \delta_{TB+}^i(n, \vec{p})$.

Let $M$ be the set of views that are accepted by the program $c$ of the coercer, i.e., for which the coercer outputs 1. In what follows, let $Z$ range over the set of all possible patterns, $\rho$ over all essential views, $\eta$ over all views, and $Rec$ over all possible vectors of receipts. We abbreviate $C(Z,i)$ by $C(Z)$. Finally, let $M_Z^{\rho,Rec} = \{\eta \in M : f(\eta) = Z,\ Rec(\eta) = Rec, \text{ and } \rho(\eta) = \rho\}$. For $Z, \rho, Rec$ with $M_Z^{\rho,Rec} \neq \emptyset$ let $\omega_1^{Z,\rho,Rec}$ be an arbitrary element of $\Omega_1$ with $\rho(Z, \omega_1^{Z,\rho,Rec}) = \rho$ and $Rec(\omega_1^{Z,\rho,Rec}) = Rec$. Then we have $\varphi_\eta(\omega_1^{Z,\rho,Rec})$ for all $\eta \in M_Z^{\rho,Rec}$. We have:

$$\begin{aligned}
\Phi &= \Pr[T \mapsto 1] - \Pr[\tilde{T} \mapsto 1] \\
&= \Pr[T \mapsto M] - \Pr[\tilde{T} \mapsto M] \\
&= \sum_Z \sum_\rho \sum_{Rec} \sum_{\eta \in M_Z^{\rho,Rec}} \left(\Pr[T \mapsto \eta] - \Pr[\tilde{T} \mapsto \eta]\right) \\
&= \sum_Z \sum_\rho (A_\rho^Z - A_\rho^{C(Z)}) \sum_{Rec} \sum_{\eta \in M_Z^{\rho,Rec}} \Pr_{\omega_2}[T(\omega_1^{Z,\rho,Rec}, \omega_2) \mapsto \eta] \\
&\qquad \cdot \Pr_{\omega_1}[Rec(\omega_1) = Rec \mid rec(\omega_1) = rec(\rho)].
\end{aligned}$$

For the third equation we use the fact that $Pr[T \mapsto \eta] - Pr[\tilde{T} \mapsto \eta] = 0$, if $f(\eta)$ is not defined, as in this case $T$ and $\tilde{T}$ coincide. With $M_Z^+ = M_{Z,i}^+$ as defined in Section 6.3, we get:

$$\Phi \leq \sum_Z \sum_{\rho \in M_Z^+} (A_\rho^Z - A_\rho^{C(Z)}) \sum_{Rec} \sum_{\eta \in M_Z^{\rho,Rec}} \Pr_{\omega_2}[T(\omega_1^{Z,\rho,Rec}, \omega_2) \mapsto \eta] \\ \cdot \Pr_{\omega_1}[Rec(\omega_1) = Rec \mid rec(\omega_1) = rec(\rho)]$$

Now, by the definition of $M_Z^{\rho,Rec}$, for $\eta \in M_Z^{\rho,Rec}$ we have $f(\eta) = Z$ and, because $f(\eta)$ depends only on $\omega_2$, we know that $T(\omega_1^{Z,\rho,Rec}, \omega_2) \mapsto \eta$ implies $f(\omega_2) = Z$. Therefore, we have $\sum_{\eta \in M_Z^{\rho,Rec}} \Pr_{\omega_2}[T(\omega_1^{Z,\rho,Rec}, \omega_2) \mapsto \eta] \leq \Pr_{\omega_2}[f(\omega_2) = Z]$. With this, we obtain:

$$\begin{aligned}
\Phi \;&\leq \sum_{Z} \Pr_{\omega_2}[f(\omega_2)=Z] \sum_{\rho \in M_Z^+} (A_\rho^Z - A_\rho^{C(Z)}) && (23)\\
&\qquad\qquad \sum_{Rec} \Pr_{\omega_1}[Rec(\omega_1)=Rec \mid rec(\omega_1)=rec(\rho)] \leq && (24)\\
&\leq \sum_{Z} \Pr_{\omega_2}[f(\omega_2)=Z] \sum_{\rho \in M_Z^+} (A_\rho^Z - A_\rho^{C(Z)}) && (25)\\
&\leq \sum_{Z} \Pr_{\omega_2}[f(\omega_2)=Z] \cdot \delta^i_{TB^+} && (26)\\
&\leq \delta^i_{TB^+} = \delta && (27)
\end{aligned}$$

This shows that $S$ is $\delta$-coercion resistant w.r.t. $\tilde{V}_i$. It remains to show that $\delta$ is optimal.

Let us consider the program $c$ of the coercer which requests the coerced voter to vote using $Z^*$ and accepts a view $\eta$ only if $\rho(\eta)$ is in $M^+_{Z^*,i}$, where $M^+_{Z,i}$ is as defined in Section 6.3, and $Z^*$ is a pattern with

$$\max_Z \sum_{\rho \in M^+_{Z,i}} (A_\rho^Z - A_\rho^{C(Z)}) = \sum_{\rho \in M^+_{Z^*,i}} (A_\rho^{Z^*} - A_\rho^{C(Z^*)}).$$

With this program $c$ of the coercer we have, for each essential view $\rho$:

$$\Pr[T \mapsto \rho] = A_\rho^{Z^*} \qquad \text{and} \qquad \Pr[\tilde{T} \mapsto \rho] = A_\rho^{C(Z^*)} \quad .$$

We immediately obtain:

$$\Phi = \sum_{\rho \in M^+_{Z^*,i}} \big(\Pr[T \mapsto \rho] - \Pr[\tilde{T} \mapsto \rho]\big) = \sum_{\rho \in M^+_{Z^*,i}} \big(A_\rho^{Z^*} - A_\rho^{C(Z^*)}\big) = \delta \ , \qquad (28)$$

which shows that $S$ is not $\delta'$-coercion resistant for any $\delta' < \delta$, if the counter-strategy $\tilde{v}$ is used. To complete the proof of Theorem 3, we need to show that every other counter-strategy $\tilde{v}'$ does not yield a smaller $\delta$.

First, note that every reasonable counter-strategy $\tilde{v}'$ should, up to a negligible set of runs, (a) cast ballots only when instructed by the coercer, (b) if instructed by the coercer to cast a ballot, cast a ballot for candidate $i$, and (c) take the receipt requested by the coercer. Failing to meet (b) would mean that $\tilde{v}' \notin \tilde{V}_i$. Conversely, to guarantee that $\tilde{v}' \in \tilde{V}_i$, the coerced voter only needs to vote if instructed by the coercer. Therefore, it is clear that in order to be as indistinguishable from the dummy strategy as possible, a counter-strategy should only cast a ballot if instructed to do so by the coercer, which explains (a). As for (c), it is clear that if a counter-strategy takes a receipt different from the one requested by the coercer, the coercer can easily distinguish this strategy from the dummy strategy. Therefore, $\tilde{v}'$ must be like $\tilde{v}$, up to the response if it is instructed to vote according to $Z_0 = \binom{\times}{\underline{\circ}}, \binom{\circ}{\times}, \binom{\circ}{\times}$, assuming $i = 1$; the case $i = 2$ is analogous. By (b) and (c) we know this response must be $\binom{\times}{\underline{\circ}}, \binom{\circ}{\times}, \binom{\times}{\circ}$ or $\binom{\times}{\underline{\circ}}, \binom{\times}{\times}, \binom{\circ}{\circ}$. One of these responses can be chosen randomly, according to some strategy. Recall from Section 6.3 that the response of the counter-strategy $\tilde{v}$ is $C(Z_0) = C(Z_0, 1) = \left(\substack{\times \\ \underline{\circ}}, \substack{\circ \\ \times}, \substack{\times \\ \circ}\right)$. If $C'(Z) = C'(Z, 1)$ denotes the response for a pattern $Z$ in the counter-strategy $\tilde{v}'$, we know that $C'(Z) = C(Z)$ for every $Z \neq Z_0$. For $Z = Z_0$, as just explained, $C'(Z)$ has two choices which could be chosen randomly. For simplicity of the argument, we assume that $C'(Z_0)$ always chooses $\left(\substack{\times \\ \underline{\circ}}, \substack{\times \\ \times}, \substack{\circ \\ \circ}\right)$; the case of a randomized choice can be treated similarly. (Note that whenever $C'(Z)$ chooses $\left(\substack{\times \\ \underline{\circ}}, \substack{\circ \\ \times}, \substack{\times \\ \circ}\right)$, then this would coincide with $C(Z_0)$.)

Let $c$ be the program of the coercer which requests the coerced voter to vote using $Z^\star$ and accepts a view $\eta$ only if $\rho(\eta)$ is in $\tilde{M}^+_{Z^\star,i}$, where $\tilde{M}^+_{Z,i} = \{\rho : A^Z_\rho \geq A^{C'(Z)}_\rho\}$, and $Z^\star$ is a pattern such that

$$\max_Z \sum_{\rho \in \tilde{M}^+_{Z,i}} (A^Z_\rho - A^{C'(Z)}_\rho) = \sum_{\rho \in \tilde{M}^+_{Z^\star,i}} (A^{Z^\star,i}_\rho - A^{C'(Z^\star)}_\rho) \ .$$

With this, analogously to (28), we have:

$$\tilde{\Phi} = \mathsf{Pr}[(c \parallel \mathsf{dum} \parallel \mathsf{e}_S) \mapsto 1] - \mathsf{Pr}[(c \parallel \tilde{v}' \parallel \mathsf{e}_S) \mapsto 1] = \sum_{\rho \in \tilde{M}^+_{Z^\star,i}} (A^Z_\rho - A^{C'(Z^\star)}_\rho).$$

Hence it remains to show that

$$\max_Z \sum_{\rho \in \tilde{M}^+_{Z,i}} (A^Z_\rho - A^{C'(Z)}_\rho) \geq \max_Z \sum_{\rho \in M^+_{Z,i}} (A^Z_\rho - A^{C(Z)}_\rho) = \delta.$$

Let $Z_1 = \left(\substack{\circ \\ \underline{\times}}, \substack{\times \\ \circ}, \substack{\circ \\ \times}\right)$. Then $C'(Z_1)$ is uniquely determined and equal to $C(Z_1)$. As the receipt of the coerced voter is *not* part of the essential view, we have for all $\omega_1$:

$$\rho(Z_1, \omega_1) = \rho(Z_0, \omega_1) \text{ and } \rho(C(Z_1), \omega_1) = \rho(C(Z_0), \omega_1).$$

It follows

$$\sum_{\rho \in M^+_{Z_1,i}} (A^{Z_1}_\rho - A^{C(Z_1)}_\rho) = \sum_{\rho \in M^+_{Z_0,i}} (A^{Z_0}_\rho - A^{C(Z_0)}_\rho).$$

Now, we obtain:

$$\begin{aligned} \max_Z \sum_{\rho \in \tilde{M}^+_{Z,i}} (A^Z_\rho - A^{C'(Z)}_\rho) &\geq \max_{Z \neq Z_0} \sum_{\rho \in \tilde{M}^+_{Z,i}} (A^Z_\rho - A^{C'(Z)}_\rho) \\ &= \max_{Z \neq Z_0} \sum_{\rho \in M^+_{Z,i}} (A^Z_\rho - A^{C(Z)}_\rho) \\ &= \max_Z \sum_{\rho \in M^+_{Z,i}} (A^Z_\rho - A^{C(Z)}_\rho). \end{aligned}$$

This concludes the proof of Theorem 3.

### D.2 Proof of Lemma 4

In the proof of Lemma 4, we will use the following easy to prove facts (see [10] for similar results).

**Lemma 9.** *Consider honest, non-abstaining voters.*

1. *The probability that a voter takes receipt* $\substack{\times\\\times}$ *is* $\frac{1}{9}$.
2. *The probability that a voter takes receipt* $\substack{\times\\\circ}$ *and the probability that she takes receipt* $\substack{\circ\\\times}$ *is* $\frac{2}{9}$.
3. *The probability that a voter who does not abstain votes for candidate 1 (or candidate 2) is independent of the receipt she gets and is* $\frac{p_1}{p_1+p_2}$ *(or* $\frac{p_2}{p_1+p_2}$*, respectively).*
4. *The probability that a voter produces a* $\substack{\times\\\times}$*-ballot is* $\frac{1}{2}$ *in either of the following cases: (a) if we assume that she votes for candidate* 1 *and takes* $\substack{\times\\\circ}$ *as a receipt, and (b) if she votes for candidate 2 and takes* $\substack{\circ\\\times}$ *as a receipt.*

We will use the following random variables on $\Omega_1$: $F(\omega_1)$ denotes the number of $(\substack{\times\\\times})$-ballots, $R(\omega_1)$ is the number of votes of honest voters for candidate 1, $N(\omega_1)$ is the number of non-abstaining honest voters, $\tau_1(\omega_1)$ denotes the number of voters that vote for 2 and take $\substack{\times\\\circ}$ as receipt, and $\tau_2(\omega_1)$ is the number of voters that vote for 1 and take $\substack{\circ\\\times}$ as receipt.

Let $\rho = (n_{\substack{\times\\\times}}, n_{\substack{\times\\\circ}}, n_{\substack{\circ\\\times}}, r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}})$. Let $N = (2n_{\substack{\times\\\times}} + n_{\substack{\times\\\circ}} + n_{\substack{\circ\\\times}})/3$ denote the total number of non-abstaining voters and $R = (n_{\substack{\times\\\times}} + n_{\substack{\times\\\circ}}) - N$ denote the votes for candidate 1. Then we have the following equality, where $\tau_1$ and $\tau_2$ range over $\{0,\dots,n\}$.

$$\Pr_{\omega_1}[\rho(\omega_1)=\rho] = \sum_{\tau_1,\tau_2} \Pr_{\omega_1}[\rho(\omega_1)=\rho, \tau_1(\omega_1)=\tau_1, \tau_2(\omega_1)=\tau_2].$$

Moreover, since $n_{\substack{\times\\\times}}$, $N$, and $R$ determine $n_{\substack{\times\\\circ}}$ and $n_{\substack{\circ\\\times}}$ and vice versa, we have for all $\tau_1, \tau_2$

$$\begin{aligned}
&\Pr_{\omega_1}[\rho(\omega_1)=\rho, \tau_1(\omega_1)=\tau_1, \tau_2(\omega_1)=\tau_2]\\
&=\Pr_{\omega_1}[F(\omega_1)=n_{\substack{\times\\\times}}, R(\omega_1)=R, \tau_1(\omega_1)=\tau_1, \tau_2(\omega_1)=\tau_2, rec(\omega_1)=(r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}}), N(\omega_1)=N]\\
&=\Pr_{\omega_1}\left[F(\omega_1)=n_{\substack{\times\\\times}} \mid R(\omega_1)=R, \tau_1(\omega_1)=\tau_1, \tau_2(\omega_2)=\tau_2, rec(\omega_1)=(r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}}), N(\omega_1)=N\right]\cdot \qquad (29)\\
&\quad\cdot\Pr_{\omega_1}\left[R(\omega_1)=R \mid \tau_1(\omega_1)=\tau_1, \tau_2(\omega_2)=\tau_2, rec(\omega_1)=(r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}}), N(\omega_1)=N\right]\cdot \qquad (30)\\
&\quad\cdot\Pr_{\omega_1}\left[\tau_1(\omega_1)=\tau_1, \tau_2(\omega_2)=\tau_2 \mid rec(\omega_1)=(r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}}), N(\omega_1)=N\right]\cdot \qquad (31)\\
&\quad\cdot\Pr_{\omega_1}\left[rec(\omega_1)=(r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}}) \mid N(\omega_1)=N\right]\cdot \qquad (32)\\
&\quad\cdot\Pr_{\omega_1}[N(\omega_1)=N] \quad . \qquad (33)
\end{aligned}$$

For (33), we have

$$\Pr_{\omega_1}[N(\omega_1)=N] = \binom{n}{N} p_0^{n-N}(p_1+p_2)^N.$$

For (32), we have to distribute (independently) the receipts to the $N$ non-abstaining voters. With Lemma 9 we obtain:

$$\Pr_{\omega_1}\left[rec(\omega_1) = (r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}}) \mid N(\omega_1) = N\right]$$
$$= \binom{N}{r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}}} \left(\frac{1}{9}\right)^{r_{\substack{\times\\\times}}} \left(\frac{2}{9}\right)^{r_{\substack{\times\\\circ}}+r_{\substack{\circ\\\times}}} \left(\frac{4}{9}\right)^{N-r_{\substack{\times\\\times}}-r_{\substack{\times\\\circ}}-r_{\substack{\circ\\\times}}}.$$

For (31), we have to distribute $\tau_1$ votes for candidate 2 in the set of those voters that get $\substack{\times\\\circ}$ as a receipt and, similarly, $\tau_2$ votes in the set of those voters that get $\substack{\circ\\\times}$ as a receipt.

$$\Pr\left[\tau_1(\omega_1) = \tau_1, \tau_2(\omega_1) = \tau_2 \mid rec(\omega_1) = (r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}}), N(\omega_1) = N\right]$$
$$= \binom{r_{\substack{\times\\\circ}}}{\tau_1} q^{r_{\substack{\times\\\circ}}-\tau_1}(1-q)^{\tau_1} \cdot \binom{r_{\substack{\circ\\\times}}}{\tau_2} q^{\tau_2}(1-q)^{r_{\substack{\circ\\\times}}-\tau_2}.$$

where $q = \frac{p_1}{p_1+p_2}$.

For (30), we have to distribute the rest of the votes for candidate 1 (i.e., $R-(r_{\substack{\times\\\circ}}-\tau_1)-\tau_2$) to those non-abstaining voters that do not get $\substack{\times\\\circ}$ or $\substack{\circ\\\times}$ as receipt. With Lemma 9 we have that the probability that a non-abstaining voter votes for candidate 1 is $q$, regardless of the receipt. Hence we have

$$\Pr_{\omega_1}\left[R(\omega_1) = R \mid \tau_1(\omega_1) = \tau_1, \tau_2(\omega_1) = \tau_2, rec(\omega_1) = (r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\times\\\circ}}), N(\omega_1) = N\right] =$$
$$= \binom{N - r_{\substack{\times\\\circ}} - r_{\substack{\circ\\\times}}}{R-(r_{\substack{\times\\\circ}}-\tau_1)-\tau_2} \cdot q^{R-(r_{\substack{\times\\\circ}}-\tau_1)-\tau_2} \cdot (1-q)^{N-r_{\substack{\circ\\\times}}-R-\tau_1+\tau_2}.$$

For (29), we have to spot $n_{\substack{\times\\\times}}$ voters that submit an $\substack{\times\\\times}$-ballot among all voters. Clearly, every voter that takes $\substack{\times\\\times}$ or $\substack{\circ\\\circ}$ as receipt, submits an $\substack{\times\\\times}$-ballot. Also, the voters who vote according to $\tau_1$ or $\tau_2$ do not submit a $\substack{\times\\\times}$-ballot (that was the reason for introducing $\tau_1, \tau_2$). Hence we have to distribute $n_{\substack{\times\\\times}} - r_{\substack{\times\\\times}} - (N - r_{\substack{\times\\\times}} - r_{\substack{\times\\\circ}} - r_{\substack{\circ\\\times}}) = n_{\substack{\times\\\times}} - N + r_{\substack{\times\\\circ}} + r_{\substack{\circ\\\times}}$ among $N - r_{\substack{\times\\\times}} - (N - r_{\substack{\times\\\times}} - r_{\substack{\times\\\circ}} - r_{\substack{\circ\\\times}}) - \tau_1 - \tau_2$ voters. Note that any of those voters either votes for candidate 1 with receipt $\substack{\times\\\circ}$ or for candidate 2 with receipt $\substack{\circ\\\times}$. The probability that such a voter submits a $\substack{\times\\\times}$-ballot is $\frac{1}{2}$, according to Lemma 9. Hence we have

$$\Pr\left[F(\omega_1) = n_1 \mid R(\omega_1) = R, \tau_1(\omega_1) = \tau_1, \tau_2(\omega_1) = \tau_2, rec(\omega_1) = (r_{\substack{\times\\\times}}, r_{\substack{\times\\\circ}}, r_{\substack{\circ\\\times}}), N(\omega_1) = N\right] =$$
$$= \binom{r_{\substack{\times\\\circ}} + r_{\substack{\circ\\\times}} - \tau_1 - \tau_2}{n_{\substack{\times\\\times}} - N + r_{\substack{\times\\\circ}} + r_{\substack{\circ\\\times}}} \left(\frac{1}{2}\right)^{r_{\substack{\times\\\circ}}+r_{\substack{\circ\\\times}}-\tau_1-\tau_2}.$$

By putting everything together and rewriting the formula, we obtain the formula in Lemma 4.

## E Proof of Theorem 4

Let us consider the program $c \in \mathscr{C}$ which does the following:

- It instructs the coerced voter to vote for the candidate $j$ for which the sum in equation (8) achieves its maximum. (Note that, by the definition of $\mathscr{C}$, the exact pattern the coerced voter is supposed to use is determined.)

- It accepts a run if and only if the receipt given by the voter is as required and the restricted view $\rho$ in this run is in $M_{i,j}$.

Let $v^*$ be the counter-strategy as defined in Section 6.4. As argued in this section, this strategy is optimal for $\mathscr{C}$ and therefore for $c$. Hence, to prove Theorem 4, it suffices to show that

$$\Phi = \mathsf{Pr}[T \mapsto 1] - \mathsf{Pr}[\tilde{T} \mapsto 1] \geq \delta,$$

where $T = (c \parallel \mathsf{dum} \parallel \mathsf{e}_S)$, $\tilde{T} = (c \parallel v^* \parallel \mathsf{e}_S)$, and $\delta = \delta_i(n,k,\vec{p})$. In fact, we have

$$\Phi = \sum_{\rho \in M_{i,j}} \big(\mathsf{Pr}[T \mapsto \rho] - \mathsf{Pr}[\tilde{T} \mapsto \rho]\big) = \sum_{\rho \in M_{i,j}} \big(A_\rho^{j,o} - A_\rho^{i,c}\big) = \delta,$$

where we use the equalities

$$\mathsf{Pr}[T \mapsto \rho] = A_\rho^{j,o} \quad \text{and} \quad \mathsf{Pr}[\tilde{T} \mapsto \rho] = A_\rho^{i,c} \ .$$

These equalities hold true, because the events $T \mapsto \rho$ and $\tilde{T} \mapsto \rho$ depend only on the choices made by honest voters and the coerced voter. This concludes the proof of Theorem 4.