Verifiability, Privacy, and Coercion-Resistance: New Insights from a Case Study

Ralf Küsters, Tomasz Truderung, and Andreas Vogt
University of Trier, Germany
{kuesters,truderun,vogt}@uni-trier.de

***Abstract*—In this paper, we present new insights into central properties of voting systems, namely verifiability, privacy, and coercion-resistance. We demonstrate that the combination of the two forms of verifiability considered in the literature—individual and universal verifiability—are, unlike commonly believed, insufficient to guarantee overall verifiability. We also demonstrate that the relationship between coercion-resistance and privacy is more subtle than suggested in the literature.**

Our findings are partly based on a case study of prominent voting systems, ThreeBallot and VAV, for which, among others, we show that, unlike commonly believed, they do not provide any reasonable level of verifiability, even though they satisfy individual and universal verifiability. Also, we show that the original variants of ThreeBallot and VAV provide a better level of coercion-resistance than of privacy.

***Keywords*-voting; verifiability; coercion-resistance; privacy; protocol analysis**

I. Introduction

Verifiability, privacy, and coercion-resistance are central security requirements for modern voting systems. *Privacy* is the most basic security requirement, which says that the way a particular voter voted is not revealed to anybody, see, e.g., [2], [4]. Intuitively, *verifiability* means that voters have a way of checking that their votes were actually counted and that the published result of the election is correct. In the literature, traditionally two forms of verifiability are considered: individual and universal verifiability [14], [1], [24], [12]. As stated, e.g., in [14], *individual verifiability* means that a voter can check that her own ballot appears on the bulletin board. *Universal verifiability* requires that anyone can check that the election outcome corresponds to the ballots published on the bulletin board. In addition, it is explicitly or implicitly required that each vote in the election outcome was cast by an eligible voter and that there is at most one vote per voter, a fact that in some voting protocols can be verified as well. To achieve (individual) verifiability, a voter typically obtains some kind of receipt which, together with additional data published in the election, she can use to check that her vote was counted. This, however, potentially opens up the possibility for vote buying and voter coercion. Besides verifiability, voting systems should therefore also provide so-called *coercion-resistance* [2], [13], [23].

In this paper, we present new insights into these central security properties. Our findings are partly based on a case study for prominent voting protocols, ThreeBallot and VAV [25]. More precisely, the contribution of this paper is as follows.

Contribution of this Paper. We demonstrate, using ThreeBallot and VAV as examples, that the combination of the two mentioned forms of verifiability considered in the literature—individual and universal verifiability—are, unlike commonly believed, insufficient to guarantee overall verifiability. More precisely, based on a definition of verifiability proposed in [19], we precisely measure the level of verifiability ThreeBallot and VAV provide. It turns out that, while ThreeBallot and VAV satisfy individual and universal verifiability, there is an attack on the verifiability of these protocols, which results in an insufficient level of verifiability. Our attack allows a dishonest bulletin board (or the scanner), collaborating with m dishonest voters, to turn m votes of honest voters for A into m votes for B. This goes unnoticed even if all honest voters check whether their receipts appear on the bulletin board and even if they check that the published result corresponds to the ballots shown on the bulletin board.

As for privacy and coercion-resistance, we first provide a game-based definition of privacy, along the lines of a game-based definition of coercion-resistance proposed in [18], since, as to the best of our knowledge, such a definition does not exist. There are, however, definitions of coercion-resistance in simulation-based settings (see, e.g., [21]) and in an abstract, Dolev-Yao style model [9]. We note that both our definition of privacy and the definition of coercion-resistance from [18] allow to measure the level of privacy/coercion-resistance a protocol provides: for privacy, it is measured how well external observers can distinguish whether an honest voter voted for candidate j or j'; for coercion-resistance, the ability of coercers to distinguish whether coerced voters followed the coercer's instructions or not is measured. This is important in order to make meaningful statements about protocols, as many voting protocols, in particular many paper-based protocols (e.g., ThreeBallot and VAV [25], Prêt à Voter [6], and Split-Ballot [22]), do not provide the ideal level of privacy/coercion-resistance. As discussed in [18], simulation-based security definitions, e.g., the one by Moran and Naor [21] provide a yes/no-answer, rather than measuring the level of privacy/coercion-resistance. Also, they are more demanding than game-based definitions and deem many reasonable protocols insecure.

One would expect that privacy and coercion-resistance are closely related: If the level of privacy is low, i.e., there is a good chance of correctly determining how a voter voted, then this should give the coercer leverage to coerce a voter. Some works in the literature indeed suggest a close connection. For example, the definition of coercion-resistance by Moran and Naor [21], being simulation-based, has privacy "built in".

However, our case study, in which we precisely measured the level of privacy and coercion-resistance of different variants of ThreeBallot and VAV proposed in the literature,[1] demonstrates that the relationship between privacy and coercion-resistance is more subtle than what can be gathered from existing work.

Among others, it turns out that improving the level of privacy of a protocol in a natural way (e.g., by changing the way honest voters fill out ballots) can lead to a *lower* level of coercion-resistance. This is the case when going from the original variant of ThreeBallot to a "privacy enhanced" variant proposed by de Marneffe et al. [8]. Clearly, in general, one does not expect privacy to imply coercion-resistance. Still the effect is quite surprising.

A maybe even more important and unexpected finding that comes out of our case study is that the level of privacy of a protocol can be much *lower* than its level of coercion-resistance; this is the case for the original variant of ThreeBallot [25] and a natural variant of VAV. The reason behind this phenomenon is basically that it may happen that the counter-strategy a coerced voter may run to defend against coercion hides the behavior of the coerced voter, including her vote, better than the honest voting program.

To complete the picture, we also study the case in which the counter-strategy does not "outperform" the honest voting program in the above sense. For this case, we are able to prove a general theorem that states that if a voting system provides a certain level of coercion-resistance, it provides at least the same level of privacy. As discussed in Section VI-E, this theorem is applicable to many voting protocols.

Structure of this Paper. We first introduce some basic terminology and introduce the notion of a voting protocol. In Sections III and IV we recall (the variants of) ThreeBallot and VAV, respectively. Our findings on verifiability are presented in Section V and those for privacy and coercion-resistance in Section VI. We conclude in Section VII. Some more details are provided in the appendix. Further details and proofs can be find in the full version of this paper [20].

II. Preliminaries and Protocols

In this section, we introduce some basic terminology and the notion of a voting protocol.

Preliminaries. As usual, a function f from the natural numbers to the real numbers is *negligible*, if for every $c > 0$ there exists ℓ_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 *δ-bounded* if f is bounded by δ plus a negligible function, i.e., for every $c > 0$ there exists ℓ_0 such that $f(\ell) \leq \delta + \frac{1}{\ell^c}$ for all $\ell > \ell_0$.

We use systems of probabilistic polynomial-time interactive Turing machines (ITMs) as our computational model (see, e.g., [15]). In a system of ITMs, also called a *process*, ITMs can communicate with other ITMs via input/output tapes, also called *(external) input/output* channels. If π and π' are processes (each with a set of external input/output tapes), then we write $\pi \parallel \pi'$ for the concurrent composition of π and π'. A process defines a family of probability distributions over runs, indexed by the security parameter.

Voting Protocols. A voting protocol P specifies the programs of the honest voters and authorities in a voting process. More precisely, let k be the number of candidates and q be the number of voters. Then, P specifies:

- A set $\{\mathsf{a}_1, \ldots, \mathsf{a}_l\}$ of voting authorities and a program $\hat{\pi}_{\mathsf{a}}$, for every voting authority a. The specification of $\hat{\pi}_{\mathsf{a}}$ includes the specification of the interface of a to the rest of the voting process, i.e., the channels via which a is connected to other parties.
- A program (formally a process) $\hat{\pi}_{\mathsf{v}}$, for every voter $\mathsf{v} \in \{\mathsf{v}_1, \ldots, \mathsf{v}_q\}$. The specification of $\hat{\pi}_{\mathsf{v}}$ includes the specification of the interface of v to the rest of the voting process. The program $\hat{\pi}_{\mathsf{v}}$ takes a choice $j \in \{0, \ldots, k\}$, where $j = 0$ stands for abstention from voting, as parameter, indicating which candidate v votes for (if any).

In the following, we will consider a probability distribution $\vec{p} = p_0, \ldots, p_k$ on the possible choices honest voters have, i.e., $p_0, \ldots, p_k \in [0,1]$ and $\sum_{i=0}^{k} p_i = 1$, where p_0 is the probability that a voter abstains from voting and p_i, $i \in \{1, \ldots, k\}$, is the probability that a voter votes for candidate i. We define $\hat{\pi}_{\mathsf{v}}(\vec{p})$ to be the process which first chooses $j \in \{0, \ldots, k\}$ according to $\vec{p}$ and then runs $\hat{\pi}_{\mathsf{v}}(j)$. We sometimes simply write $\hat{\pi}_{\mathsf{v}}$ instead of $\hat{\pi}_{\mathsf{v}}(\vec{p})$, if the distribution $\vec{p}$ is clear from the context.

Because, as we will see, the level of privacy, coercion-resistance, and verifiability of a protocol P depends on several parameters, we consider protocol instantiations P^* of P, for which these parameters are fixed. The parameters are the following:

(i) the set $A_H \subseteq \{\mathsf{a}_1, \ldots, \mathsf{a}_l\}$ of honest voting authorities A_H,
(ii) the total number q of voters and the set $V_H \subseteq \{\mathsf{v}_1, \ldots, \mathsf{v}_q\}$ of honest voters (static corruption),
(iii) the number k of candidates, and
(iv) the probability distribution $\vec{p}$, as described above.

Such a *protocol instantiation* will be denoted by $P^* = P(A_H, q, V_H, k, \vec{p})$. We note that in our theorems, only the number of honest (and dishonest) voters will matter, not the specific set V_H of honest voters. Therefore, we often simply write $P(A_H, q, n, k, \vec{p})$ with $n = |V_H|$.

III. The ThreeBallot voting scheme

In ThreeBallot [25], a voter is given a multi-ballot consisting of three simple ballots. On every simple ballot the candidates are printed in the same fixed order. In the secrecy of a voting booth, the voter is supposed to fill out all three simple ballots

[1] We note that coercion-resistance of one of the variants of ThreeBallot considered in this paper has already been studied in [18].

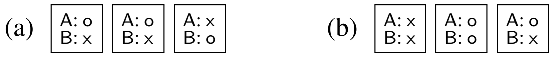

Fig. 1. Two ways of voting for the second candidate (candidate B) in the ThreeBallot protocol, where x represents a marked position and o represents an unmarked position. All the other possibilities of voting for B can be obtained as permutations of these two.

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 1 shows two ways of voting for candidate B. After this, she feeds all three simple ballots to a voting 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 length of these numbers is the length of the security parameter and 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 simple ballots (now separated) in a ballot box. We assume that clerks guarantee that only registered voters can vote and that every voter votes at most once.

In the tallying phase, the voting machine posts on the bulletin board (electronic copies of) all the cast simple ballots in a random order. From the ballots shown on the bulletin board the result can easily be computed: 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 simple ballot that a voter can take as a receipt can be part of a multi-ballot that forms a valid vote for any candidate. Also, ThreeBallot was meant to provide (some level of) verifiability. For this, a crucial assumption, already made in the original paper [25], is that neither the scanner, the voting authority, nor the bulletin board knows which simple ballots have been taken as receipts by honest voters before all ballots were published. Now, as each voter may check whether the simple ballot she has taken as a receipt appears on the bulletin board, it should be risky for any party to remove or alter simple ballots in order to manipulate the result since the probability that the modification of k simple ballots goes undetected is merely $(\frac{2}{3})^k$. Unfortunately, as we will see in Section V-B, this argument, found in the literature, is flawed.

As mentioned in the introduction, there are two variants of ThreeBallot which differ in the way an honest voter fills out the ballot: the original variant by Rivest [25] and a variant by de Marneffe et al. [8].

The original variant. In this variant of the protocol a voter first, for each candidate, randomly chooses a simple ballot on which she then marks the position corresponding to this candidate. Then, she randomly picks a simple ballot for which the position corresponding to the candidate of her choice is not yet marked, and she marks this position. Finally, she randomly chooses one ballot as a receipt.

The variant of de Marneffe et al. In this variant of the protocol 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 a randomly chosen simple ballot on which this position has not yet been marked and which is not the ballot chosen as a receipt; we remark that in some cases there will be only one such simple ballot.

The advantage of this procedure is that the receipt an honest voter gets is stochastically independent of the candidate the voter votes for, which in turn should give better privacy. We note that in [8], ThreeBallot was analyzed in a simulation-based setting, focusing on privacy. The analysis was based on the (only informally stated) assumption that the adversary, given a receipt, is not able to reconstruct the exact way the corresponding multi-ballot was filled out. However, this assumption is unjustified: Runs for which an adversary can reconstruct the multi-ballots occur with non-negligible probability, as illustrated by the following example:

It may happen (with *non-negligible* probability, depending only on the probability distribution $\vec{p}$ and the number of voters) that *each* voter marks both positions on the first simple ballot, no position on the second one, exactly one position on the third ballot, and then take the last ballot as her receipt, as shown in Figure 1, (b) for the case that the voter votes for B. In this case, a receipt directly indicates the choice of the voter, which completely breaks privacy.

In what follows, we denote by $\big(\substack{\mathsf{x}\\ \underline{\mathsf{o}}},\substack{\mathsf{x}\\ \mathsf{x}},\substack{\mathsf{o}\\ \mathsf{o}}\big)$, $\big(\substack{\mathsf{o}\\ \underline{\mathsf{x}}},\substack{\mathsf{x}\\ \mathsf{x}},\substack{\mathsf{x}\\ \mathsf{o}}\big)$ etc. multi-ballots filled out by voters, where the underlined simple ballots ($\substack{\mathsf{x}\\ \mathsf{o}}$ and $\substack{\mathsf{o}\\ \mathsf{x}}$, respectively) represent those simple ballots picked as receipts by the voters; we refer to these objects as *patterns*. A pattern does not fix the order of simple ballots, e.g., $\big(\substack{\mathsf{o}\\ \underline{\mathsf{x}}},\substack{\mathsf{x}\\ \mathsf{x}},\substack{\mathsf{o}\\ \mathsf{o}}\big)$ is considered to be the same pattern as $\big(\substack{\mathsf{x}\\ \mathsf{x}},\substack{\mathsf{o}\\ \underline{\mathsf{x}}},\substack{\mathsf{o}\\ \mathsf{o}}\big)$.

IV. The VAV voting scheme

In this section, we describe the VAV voting scheme [25]. In VAV, a voter is given a multi-ballot consisting of three simple ballots. On every simple ballot the candidates are printed in the same fixed order. On the top of one of those simple ballots the letter A is printed; on the top of the remaining two simple ballots the letter V is printed. In the secrecy of a voting booth, the voter is supposed to fill out her multi-ballot in the following way: (S1) She marks the position next to the candidate of her choice on one of the V-ballots and then (S2) she marks the position next to some randomly chosen candidate on the two remaining simple ballots (one V- and one A-ballot). Figure 2 shows all three ways of filling out the multi-ballot for candidate 1 in an election with three candidates. After this, she feeds all three simple ballots to a voting machine (some kind of scanner) and indicates the simple ballot she wants to get as a receipt. The machine checks

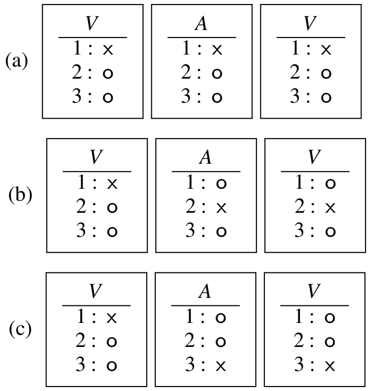

Fig. 2. Three ways of voting for the candidate 1 in the VAV protocol, where x represents a marked position and o represents an unmarked position.

the well-formedness of the multi-ballot, prints secretly random numbers on each simple ballot, where the length of these numbers is the length of the security parameter and 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 simple ballots (now separated) in a ballot box.

In the tallying phase, the voting machine posts on the bulletin board (electronic copies of) all the cast simple ballots in a random order. From the ballots shown on the bulletin board the result can easily be computed: All A-ballots and the corresponding V-ballots (i.e. V-ballots marked at the same position) are removed. From the remaining V-ballots the number of votes for each candidate can directly be read off.

Intuitively, as in the case of ThreeBallot, the system is coercion-resistant (at least to some extent), as every simple ballot that a voter can take as a receipt can be part of a multi-ballot that forms a valid vote for any candidate. Similarly to ThreeBallot, VAV is supposed to provide (some level of) verifiability as it should be risky for any party to remove or alter simple ballots in order to manipulate the result.

In the description of VAV in [25] it is not specified how exactly a voter chooses the receipt. This, however, can heavily affect the properties we study in this paper. We therefore investigate two variants of this protocol.

Simple variant. In the simplest case, a voter could choose one of her three simple ballots with uniform probability as her receipt. We will call this the *simple variant* of VAV. For this variant, however, the receipt is not independent of the candidate of her choice, which, as we will see, has a significant negative effect on privacy and coercion-resistance. This is why we consider also another variant, where privacy is enhanced.

Privacy enhanced variant. In this variant, a voter chooses one of the two simple ballots *not used* in step (S1), that is, one of those ballots where the random candidate has been marked. By this, the receipt is stochastically independent of the candidate the voter votes for.

While the privacy enhanced variant significantly improves the level of privacy and coercion-resistance, it decreases the level of verifiability: The voting machine is often able to determine one simple ballot in a multi-ballot that was certainly *not* taken as a receipt and consequently it can change this simple ballot without being detected. For instance, if an honest voter submits a multi-ballot as shown in Figure 2, (b) or (c), the machine knows that the left-most ballot cannot be chosen as a receipt. However, as we will see, the VAV voting scheme suffers from the same kind of attack the ThreeBallot protocol does, independently of how the receipts are chosen. Therefore, the level of verifiability is, in any case, very low.

In what follows, we will use, analogously to the case of ThreeBallot, the notion of a *pattern* which specifies how a multi-ballot is filled out and which simple ballot is taken as a receipt.

V. Verifiability

In this section, we first recall the definition of verifiability from [19], where, however, we use a slightly simplified definition which is sufficient for our setting. Next, we present our analysis of verifiability for ThreeBallot and VAV, including the mentioned attacks. We then conclude with remarks on the inadequacy of the notions of individual and universal verifiability demonstrated by our attacks.

A. Definition of Verifiability

The definition of verifiability in [19] assumes a *verifier*, also called a *judge*, who can be an honest regular protocol participant or an honest external observer. Now, informally speaking, verifiability says that if in a run of the voting protocol an important goal is not achieved — typically, the published result of the election is not correct, i.e., does not correspond to the votes actually cast by eligible voters —, then the verifier does not accept the run/the election. Conversely, if in a run certain parties which are supposed to make sure that the goal is achieved, such as (a subset of) the voting authorities, behave honestly, then the verifier accepts the run.

More formally, let $P^* = P(A_H, q, V_H, k, \vec{p})$ be a protocol instantiation. Given P^*, for each protocol participant a in P^*, we consider the set $\Pi(a)$ of all programs a may run. This set is defined as follows: If a is assumed to be honest (i.e. $a \in A_H \cup V_H$), then $\Pi(a) = \{\hat{\pi}_a\}$, i.e., $\Pi(a)$ consists only of the honest program of a as specified by the protocol. Otherwise, if a is not assumed to be honest, then $\Pi(a)$ consists of *all* processes limited only by a's network interface, which is the network interface that $\hat{\pi}_a$ has. Note that in any case $\hat{\pi}_a \in \Pi(a)$.

Let $\Sigma = \{b_1, \ldots, b_t\}$ be the set of all protocol participants of P^*. Then, a *process induced by* P^*, also called an *instance*, is a process of the form $\pi = (\pi_{b_1} \parallel \ldots \parallel \pi_{b_t})$, where $\pi_{b_i} \in \Pi(b_i)$. Such a process is called an *instance with honest* $B \subseteq \Sigma$ if $\pi_{b_i} = \hat{\pi}_{b_i}$ for all $b_i \in B$. A *run of* P^* is a run of some instance of P^*. Such a run is called a *run with honest* B if it is a run of an instance of P^* with honest B.

The definition of verifiability is parameterized by a goal γ, which, formally, is a set of runs of instances of P^*. In the context of voting, γ will typically contain all those runs in which the published result of the election is correct, i.e., corresponds to the votes actually cast by eligible voters.

We say that a party a, playing the role of a verifier, *accepts a run*, if in this run a sends the message accept on some designated channel $\mathsf{decision}_a$. Intuitively, a accepts a run if she believes that the goal γ has been achieved in this run.

For an instance π of P^*, by $\mathsf{Pr}[\pi(1^\ell) \mapsto (a : \mathsf{accept})]$ we denote the probability that π, running with security parameter 1^ℓ, produces a run which is accepted by a. Similarly, by $\mathsf{Pr}[\pi(1^\ell) \mapsto \neg\gamma, (a : \mathsf{accept})]$ we denote the probability that π, running with security parameter 1^ℓ, produces a run in which the goal has not been achieved, i.e., a run that does not belong to γ, but which nevertheless was accepted by a.

Definition 1 ([19], simplified[2])**.** *Let $P^* = P(A_H, q, V_H, k, \vec{p})$ be a protocol instantiation and let Σ be the set of protocol participants in P^*. Let $\delta \in [0,1]$, $B \subseteq \Sigma$, $a \in A_H \cup V_H$ (playing the role of the verifier), and γ be a goal of P^*. Then,* the goal γ is guaranteed in P^* by B and δ-verifiable by a *if, for every instance π of P^*, the following conditions are satisfied:*

- *(i) If π is an instance with honest B, then $\mathsf{Pr}[\pi(1^\ell) \mapsto (a : \mathsf{accept})]$ is overwhelming as a function of the security parameter.*
- *(ii) $\mathsf{Pr}[\pi(1^\ell) \mapsto \neg\gamma, (a : \mathsf{accept})]$ is δ-bounded as a function of the security parameter.*

Condition (ii) guarantees that the probability that a accepts a run even though the goal has *not* been achieved (e.g., the published result of the election is incorrect) is “small”, i.e., bounded by δ. Condition (i) says that the protocol is sound w.r.t. a set B of agents in the following sense: If the agents in B are honest, then a accepts runs with overwhelming probability, which by Condition (ii) implies that in those runs the goal has indeed been achieved. Typically, the set B includes (a subset of) voting authorities/machines, i.e., those agents that suffice to guarantee that the goal is achieved. Note that without Condition (i) every protocol in which no runs are accepted by the verifier would be verifiable. Also note that requiring the probability in (ii) to be negligible, i.e., requiring $\delta = 0$, while highly desirable, would be too strong for many reasonable protocols. This is due to the fact that checks (by authorities and voters) are often imperfect and partial, as illustrated in subsequent sections. The value of δ determines the *level of verifiability* a protocol provides.

B. Verifiability of ThreeBallot and VAV

In this section, we study verifiability of ThreeBallot and VAV. We precisely measure the level of verifiability of these systems and show that, unlike commonly believed, these systems do not provide any reasonable level of verifiability. We start with the analysis of ThreeBallot.

ThreeBallot. We first describe the attack on the verifiability of ThreeBallot and then precisely state the level of verifiability of this system.

The Attack on the Verifiability of ThreeBallot. As mentioned in Section III, in the literature the reasoning for the verifiability of ThreeBallot has so far been that, if voters check whether their receipt (a simple ballot) appears on the bulletin board, it should be risky for any party to remove or alter simple ballots since the probability that the modification of k simple ballots goes undetected is merely $(\frac{2}{3})^k$. However, the following attack shows that this reasoning is flawed.

Our attack assumes that there are dishonest voters and that one of the voting authorities, the voting machine or the bulletin board, is dishonest and collaborates with the dishonest voters. It is clearly realistic to assume dishonest voters and a dishonest voting authorities; defending against malicious authorities is the main point of verifiability. In what follows, we first consider the case of an election with two candidates and assume that the bulletin board is dishonest.

As already mentioned in the introduction, the effect of our attack is that m dishonest voters, collaborating with the dishonest bulletin board, can effectively vote for candidate B and, additionally, turn m votes of honest voters voting for A into votes for B. For instance, with 10 dishonest voters out of 101 voters, candidate B can win the election, even if 60 honest voters vote for A and only 31 honest voters vote for B. This goes unnoticed, provided no post-election audit based on paper ballots is performed, even if all honest voters check whether their receipts appear on the bulletin board and even if they check that the published result corresponds to the ballots shown on the bulletin board. Note that if no voter complains, then no post-election audit may be carried out. Moreover, for the post-election audit to be effective, additional trust assumptions are required.

The attack works as follows. Let us assume that there exists an honest voter who votes for candidate A and that the bulletin board, collaborating with some dishonest voter, wants to swap such a vote. To do so, the dishonest voter casts $\left(\genfrac{}{}{0pt}{}{\times}{\underline{\circ}},\genfrac{}{}{0pt}{}{\circ}{\times},\genfrac{}{}{0pt}{}{\circ}{\times}\right)$ and sends the serial number on her receipt to the bulletin board. Then, the bulletin board replaces the simple ballot with this serial number by $\genfrac{}{}{0pt}{}{\circ}{\times}$. The result of this manipulation is as if the dishonest voter had cast $\left(\genfrac{}{}{0pt}{}{\circ}{\times},\genfrac{}{}{0pt}{}{\circ}{\times},\genfrac{}{}{0pt}{}{\circ}{\times}\right)$. The bulletin board remains consistent as these three simple ballots together with the multi-ballot submitted by the honest voter voting for A (which must be either $\left(\genfrac{}{}{0pt}{}{\times}{\circ},\genfrac{}{}{0pt}{}{\times}{\times},\genfrac{}{}{0pt}{}{\circ}{\circ}\right)$ or $\left(\genfrac{}{}{0pt}{}{\times}{\circ},\genfrac{}{}{0pt}{}{\times}{\circ},\genfrac{}{}{0pt}{}{\circ}{\times}\right)$) result in two valid votes for candidate B. Note that the multi-ballot of the honest voter remains unchanged, and hence, no voter or external observer will suspect any fraud.

By this attack, the bulletin board can safely change m votes of honest voters for one candidate to another candidate, where m is the number of dishonest voters.

A similar attack works for the case of multiple candidates. Here the simplest case is that the voting machine is dishonest. First observe that, given any multi-ballot of an honest voter voting for candidate i, it is easy to construct three simple

[2] Definition 1 is a specific instance of the general definition presented in [19]. In [19], instead of the set B, we use a more general formalism to specify sets of protocol participants, namely positive boolean formulas.

ballots (which potentially do not form a valid multi-ballot) such that these simple ballots together with the multi-ballot of the honest voter form two valid multi-ballots for candidate j. Hence, for every dishonest voter, a voting machine can change the simple ballots of this voter in such a way that they, together with a multi-ballot of an honest voter, result in two valid votes for the candidate of the machine's choice. Note that several side channels are conceivable over which a voter could reveal himself as dishonest to the voting machine, e.g., voting at a specific time, pressing buttons in a specific unusual order, or in case of many candidates, using a pre-agreed pattern to fill out the ballot. Note that this attack works even if the voting machine does not know which simple ballots are taken as receipts.

The Precise Level of Verifiability of ThreeBallot. We now study the precise level of verifiability of both the original variant of ThreeBallot and the variant by de Marneffe et al., showing that only changing votes beyond the number of dishonest voters increases the risk of being detected.

In both cases, we assume that there is a protocol participant *ver* (a regular protocol participant or an external observer), the verifier, who *does not* accept a run iff some voter complains rightly (i.e. she has a receipt that does not appear correctly on the bulletin board) or the bulletin board is inconsistent (e.g., the number of simple ballots is not divisible by three, two serial numbers occur twice, a candidate got less marks than the number of voters, etc.). We assume that an honest voter checks that her receipt occurs on the bulletin board with probability p_{check}—it is realistic to assume that not all voters check their receipt. Clearly, this probability will affect the level of verifiability. We also make the following assumptions:

1. Only eligible voters will be allowed to vote, and only once. Also, the number of voters who actually voted is properly counted. This is typically guaranteed by clerks. A polling station should at least have one honest clerk who oversees the actions of other clerks. This assumption prevents that the voting machine or the bulletin board can place extra ballots on the bulletin board.
2. Nobody involved in publishing the result, in particular, the voting machine and the bulletin board, should get to know which receipts honest voters chose before all ballots have been published. This assumption is clearly necessary in order to achieve any reasonable level of verifiability, as otherwise the voting machine and the bulletin board could safely change the ballots that were not taken as receipts, and hence, fabricate arbitrary outcomes.
3. The verifier *ver* behaves as described above.

We note that we neither assume the voting machine nor the bulletin board to be honest.

Let $\mathsf{P}^o_{\mathsf{TB}}$ and $\mathsf{P}^p_{\mathsf{TB}}$ denote the ThreeBallot protocol in the original variant and the variant by de Marneffe et al., respectively. Based on the assumptions made above, it is straightforward to formally define the protocol instantiations $\mathsf{S}^o_{\mathsf{TB}} = \mathsf{P}^o_{\mathsf{TB}}(\{ver\}, q, V_H, k, \vec{p})$ of $\mathsf{P}^o_{\mathsf{TB}}$ and $\mathsf{S}^p_{\mathsf{TB}} = \mathsf{P}^o_{\mathsf{TB}}(\{ver\}, q, V_H, k, \vec{p})$ of $\mathsf{P}^p_{\mathsf{TB}}$, along with the sets $\Pi(a)$ for every protocol participant a as introduced in Section V-A. Note that $A_H = \{ver\}$ does not include the voting machine or the bulletin board as they are not assumed to be honest. The verifier *ver* could also belong to V_H. Clerks are not modeled explicitly. The interface the voters have to the rest of the system guarantees assumption 1. above. We define $n = |V_H|$.

We consider the goal γ_ℓ which, intuitively, states that at most ℓ votes of honest voters are changed, i.e., the published result is correct (1) up to votes of dishonest voters and (2) up to ℓ votes of honest voters. Note that for dishonest voters not much can be guaranteed as they might, for example, ignore the fact that their receipts are not shown or were modified on the bulletin board. More precisely, γ_ℓ is defined as follows: γ_ℓ contains all runs for which there exist choices of the dishonest voters (where a choice is either to abstain or to vote for one of the candidates) such that the result obtained together with the choices made by the honest voters in this run differs only by ℓ votes from the published result (i.e. the result that can be computed from the simple ballots on the bulletin board).

The following theorem states the levels of verifiability of the two variants of ThreeBallot (see Appendix A for the proof), where q_ℓ denotes the probability of the event that in a run of the protocol there exists a candidate c such that the sum of all votes of honest voters for all candidates except c is at least ℓ. Note that if such an event has not occurred, then it is impossible to violate the goal $\gamma_{\ell-1}$, because, by assumption 1., dishonest parties cannot add new ballots to the bulletin board (as opposed to changing/replacing ballots). However, q_ℓ will typically be quite close to 1.

Theorem 1. *Let B be the set containing the voting machine and the bulletin board and $x \in \{o, p\}$. Then, the goal γ_ℓ is guaranteed in $\mathsf{S}^x_{\mathsf{TB}}$ by B and δ^x_{Ver}-verifiable by ver, where*

$$\delta^p_{Ver} = q_{\ell+1} \cdot \left(1 - \frac{1}{6} \cdot p_{check}\right)^{\ell+1-min(\ell+1,m)}$$

and

$$\delta^o_{Ver} = q_{\ell+1} \cdot \left(1 - \frac{1}{3} \cdot p_{check}\right)^{\ell+1-min(\ell+1,m)} ,$$

with m being the number of dishonest voters. Moreover, δ^x_{Ver} is optimial, i.e., there is no $\delta' < \delta^x_{Ver}$ for which the goal γ_ℓ is guaranteed in $\mathsf{S}^x_{\mathsf{TB}}$ by B and δ'-verifiable by a.

Now, note that if $\ell + 1 \leq m$, i.e., the number of votes to be changed is at most the number of dishonest voters, then $\delta = \delta^p_{Ver} = \delta^o_{Ver} = q_{\ell+1}$. By the definition of δ-verifiability and the fact that δ is optimal, this means that there exists an attack—in fact, the one discussed—such that the probability that in a run more than ℓ votes of honest voters were changed and the verifier still accepted the run is δ. Note also that for $\ell > m$ the original variant of the protocol provides a better level of verifiability than the variant by de Marneffe et al., since in this case $\delta^o_{Ver} < \delta^p_{Ver}$.

VAV. An attack similar to the one for ThreeBallot also works for VAV: Let us assume that there exists an honest voter who votes for candidate j and that the bulletin board, collaborating with some dishonest voter, wants to switch such a vote to a vote for candidate i. To do so, the dishonest voter casts a multi-ballot where i is marked on all three ballots. Moreover, the dishonest voter takes the A-ballot as receipt and sends the serial number on her receipt to the bulletin board. Then, the bulletin board replaces the simple ballot with this serial number by an A-ballot where candidate j is marked. The bulletin board remains consistent, as the three simple ballots by the dishonest voter, with the A-ballot modified, together with the multi-ballot submitted by the honest voter voting for j constitute two valid votes for candidate i.

Hence, VAV too does not provide a reasonable level of verifiable. The exact (low) level of verifiability VAV achieves can be found in Appendix B.

C. Inadequacy of Individual and Universal Verifiability

In our analysis of verifiability of ThreeBallot and VAV above, we used Definition 1 as proposed by Küsters et al.[19]. This definition, applied to voting protocols, captures what is called *global verifiability* by Küsters et al. — in short, if the published result of the election is not correct, the verifier (a regular protocol participant or an external observer) should not accept the run, or only with small probability.

However, in the literature (see, e.g., [14], [1], [24], [12]), verifiability of voting protocols has traditionally been expressed by two forms of verifiability, as already mentioned in the introduction: *individual verifiability* (a voter can check that her own ballot appears on the bulletin board) and *universal verifiability* (anyone can check that the election outcome corresponds to the ballots published on the bulletin board). Note that, unlike global verifiability, these forms of verifiability assume some particular structure of the voting protocol. Also note that these forms of verifiability can be captured by Definition 1 using appropriate goals γ.

In the literature it was widely believed that individual and universal verifiability together achieve some form of global verifiability. However, our case study on ThreeBallot and VAV shows that this is not the case. These protocols achieve both individual and universal verifiability, but as we proved, their level of (global) verifiability is completely insufficient: A voter can check whether her receipt appears on the bulletin board, which gives her relatively high assurance that all her simple ballots are unmodified and appear on the bulletin board; hence, we have individual verifiability. (More precisely, as explained in Sections III and IV, if fraud would be attempted, even only on a moderate scale, the probability that at least one voter would detect a problem with her receipt would be very high.) We also obviously have universal verifiability as the result of the election can be computed by everyone based on the information available on the bulletin board.

In general, what individual and universal verifiability ignore is that dishonest authorities/voters can break the integrity of ballots of honest voters by ill-formed ballots. Therefore, we advocate using global verifiability (see above) which directly captures the required property.

VI. Privacy and Coercion-Resistance

In this section, we first introduce our definition of privacy. We also briefly recall the definition of coercion-resistance from [18]. We then present our analysis of privacy and coercion-resistance of the variants of ThreeBallot and VAV described in Sections III and IV. We conclude the section with a discussion of the relationship between privacy and coercion-resistance.

A. Definition of Privacy

For studying privacy of a protocol P, we assume that, besides the voting authorities and the voters, there is an additional party o called an *observer*. We denote by O the set of all programs an observer can run, i.e. all probabilistic polynomial-time ITMs with the following communication interface: An observer can directly connect to the interface of dishonest voters and authorities; in fact, the observer subsumes those parties. In addition, observers can observe publicly available information, such as messages posted by voting authorities. We also assume that, in a protocol instantiation $P^* = P(A_H, q, V_H, k, \vec{p})$, among the q voters, there is a voter who is under observation.

Now, a protocol instantiation $P^* = P(A_H, q, V_H, k, \vec{p})$, along with the set O of observer processes and a program π_{v} of the voter v under observation, induces a set of processes of the form $(\pi_{\mathsf{o}} \parallel \pi_{\mathsf{v}} \parallel \mathsf{e})$, where $\pi_{\mathsf{o}} \in O$ and e denotes the concurrent composition of the processes $\hat{\pi}_{\mathsf{v}'}$, $\mathsf{v}' \in V_H$, of the honest voters and the processes $\hat{\pi}_{\mathsf{a}}$, $\mathsf{a} \in A_H$, of the honest authorities; recall that the dishonest voters and the dishonest authorities are subsumed by π_{o}. We denote by $\Pr[(\pi_{\mathsf{o}} \parallel \pi_{\mathsf{v}} \parallel \mathsf{e})^{(\ell)} \mapsto 1]$ the probability that π_{o} outputs 1 in a run of the process $(\pi_{\mathsf{o}} \parallel \pi_{\mathsf{v}} \parallel \mathsf{e})$ with security parameter 1^ℓ.

In the following definition, we formalize privacy to be the inability of the observer π_{o} to distinguish whether the voter v under observation voted for candidate j or candidate j', where v runs her *honest* voting process $\hat{\pi}_{\mathsf{v}}$ as specified by the voting protocol.

Definition 2. Let $P^* = P(A_H, q, V_H, k, \vec{p})$ be a protocol instantiation along with a set O of observer processes and a voter v under observation. Let $\delta \in [0,1]$. We say that P^* achieves *δ-privacy*, if

$$\Pr[(\pi_{\mathsf{o}} \parallel \hat{\pi}_{\mathsf{v}}(j) \parallel \mathsf{e})^{(\ell)} \mapsto 1] - \Pr[(\pi_{\mathsf{o}} \parallel \hat{\pi}_{\mathsf{v}}(j') \parallel \mathsf{e})^{(\ell)} \mapsto 1] \quad (1)$$

is δ-bounded as a function of the security parameter 1^ℓ, for all $j, j' \in \{1, \ldots, k\}$ and for all $\pi_{\mathsf{o}} \in O$.

In the above definition we merely require (1) to be δ-bounded, instead of negligible, because there is always a non-negligible probability that an observer knows how a voter voted, e.g., in case all (honest) voters and the voter under consideration voted for the same candidate. In general, and as we will see below and in Section VI-C, δ depends on the distribution $\vec{p}$, the number k of candidates, and the number $n = |V_H|$ of honest voters; the number of dishonest voters is typically not relevant. By δ, we can precisely measure the level of privacy a voting protocol offers.

We note that the above definition does not imply that an observer cannot distinguish whether or not a voter voted, i.e., abstention may be detected by an observer. If abstention should not be detectable, one can simply let j and j' range over $\{0,\dots,k\}$ instead of $\{1,\dots,k\}$ in the above definition. The above definition is motivated by the fact that for many voting protocols, including ThreeBallot and VAV, abstention can be detected by an observer, since, e.g., the observer is present at the polling station or the observer can see the receipts of all voters, and in particular, he can see whether a voter does not have a receipt.

The above definition could be generalized in the obvious way by letting the observer observe many voters at the same time and quantifying over two *tuples* of votes, instead over just j and j'. While, for simplicity, in our case study we consider the version with one observed voter, our findings and theorems extend to the case with multiple observed voters.

We note that the above (cryptographic) definition of privacy is close in spirit to a definition in an abstract, Dolev-Yao style model [9]. Simulation-based definitions (see, e.g., [21]) are stronger, but often too strong to be applicable (e.g., in case of ThreeBallot and VAV).

Privacy of the Ideal Protocol. As we have already mentioned, the level δ of privacy is bigger than zero for virtually every voting protocol, as some information is always leaked by the result of the election, e.g., if one candidate got all votes—an event with non-negligible probability—, it is clear that everybody voted for this candidate. In order to have a lower bound on δ for all voting protocols (where the results are of the form considered below), we now determine the optimal value of δ for the ideal voting protocol. An *ideal voting protocol* collects the votes of all voters and then correctly publishes the result, where we assume that a result reveals the number of votes for every candidate. The argument sketched below is similar to the one for determining the level of *coercion-resistance* of the ideal voting protocol in [18].

As we will see, the level of privacy of the ideal voting protocol, denoted by $\delta_{Priv}(k,n,\vec{p})$, depends on the number k of candidates, on the number n of honest voters, and the probability distribution $\vec{p}$ on the candidates.

To define this function, we need the following terminology. Since the observer knows the votes of the dishonest voters, he can simply subtract these votes from the final result and obtain what we call the *pure result* $\vec{r} = (r_0,\dots,r_k)$ of the election, where r_i, $i \in \{1,\dots,k\}$, is the number of votes for candidate i casted by honest voters, and r_0 is the number of honest voters who abstained from voting. Note that $r_0 + \cdots + r_k = n+1$ (n honest voters plus the observed voter). We denote by Res the set of all pure results. Let $A^i_{\vec{r}}$ denote the probability that the choices made by the honest voters yield the pure result $\vec{r}$, given that the voter under observation votes for the i-th candidate. (Clearly, $A^i_{\vec{r}}$ depends on $\vec{p}$. However, we omit this in the notation.) Moreover, let $M^*_{j,j'} = \{\vec{r} \in Res : A^j_{\vec{r}} \leq A^{j'}_{\vec{r}}\}$. Now, the intuition behind the definition of $\delta_{Priv}(k,n,\vec{p})$ is as follows: If the observer, given a pure result $\vec{r}$, wants to decide whether the observed voter voted for candidate j or j', the best strategy of the observer is to opt for j' if $\vec{r} \in M^*_{j,j'}$, i.e., the pure result is more likely if the voter voted for candidate j'. This leads to the following definition:

$$\delta_{Priv}(n,k,\vec{p}) = \max_{j,j' \in \{1,\dots,k\}} \sum_{\vec{r} \in M^*_{j,j'}} (A^{j'}_{\vec{r}} - A^{j}_{\vec{r}}).$$

The following theorem states that $\delta_{Priv}(k,n,\vec{p})$ is indeed the optimal level of privacy, where VA denotes the trusted authority in the ideal voting protocol.

Theorem 2. *Let $S = \mathsf{P}_{\mathsf{ideal}}(\{VA\},q,n,k,\vec{p})$ be an instantiation of the ideal protocol and $\delta = \delta_{Priv}(n,k,\vec{p})$. Then S achieves δ-privacy. Moreover, S does not achieve δ'-privacy for any $\delta' < \delta$.*

Due to space limitations, we omit the proof in this extended abstract. Some values for $\delta_{Priv}(n,k,\vec{p})$ are depicted in Figure 3 (see the values for the ideal protocol).

B. Definition of Coercion-Resistance

We now briefly recall the definition of coercion-resistance from [18]. Since the overall setting for coercion-resistance is similar to that of privacy, we highlight the differences to privacy.

For the definition of coercion-resistance, the voter under observation considered for privacy is now replaced by a *voter under coercion*, also called a *coerced voter*. Unlike a voter under observation, a coerced voter does not have to follow the honest voting procedure but can deviate from it. We denote by V the set of all programs the coerced voter v can run. This set includes all probabilistic polynomial-time ITMs where the communication interface is that of an honest voter plus an input and output channel for communication with the coercer (see below). In particular, the set V 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.

The observer in the case of privacy is now replaced by the *coercer*. We denote by C the set of all programs a coercer can run, i.e., all probabilistic polynomial-time ITMs with a communication interface similar to that of observers, where in addition the coercer can communicate with the coerced voter.

Before recalling the formal definition of coercion-resistance, we provide some intuition. We imagine that the coercer demands full control over the voting interface of the coerced voter, i.e., the coercer wants the coerced voter to run the dummy strategy $\mathsf{dum} \in V$ instead of the program an honest voter would run. If the coerced voter in fact runs 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.

Now, informally speaking, a protocol is called coercion-resistant if the coerced voter, instead of running the dummy strategy, can run some *counter-strategy* $\tilde{v} \in V$ such that (i) by running this counter-strategy, the coerced voter achieves her own goal, e.g., votes for a specific candidate (see below), and (ii) the coercer is not able to distinguish whether the coerced voter followed his instructions (i.e., ran dum) or tried to achieve her own goal (by running $\tilde{v}$). 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: Even if the coerced voter tries to sell her vote by running dum, i.e., by following the coercer's instructions, the coercer is not able to tell whether the coerced voter is actually following the coercer's instructions or whether she is just trying to achieve her own goal. For the same reason, the coerced voter can safely run the counter-strategy and achieve her own goal, even if she is coerced into running dum.

The *goal* of the coerced voter is formalized by a set γ of runs. For example, if γ is supposed to express that the coerced voter wants to vote for a certain candidate, then γ would contain all runs in which the coerced voter (successfully) voted for this candidate.

In the formal definition of coercion-resistance, we write, analogously to the case of privacy, $\Pr[(\pi_\mathsf{c} \parallel \pi_\mathsf{v} \parallel \mathsf{e})^{(\ell)} \mapsto 1]$ for the probability that π_c outputs 1 in a run of the process $(\pi_\mathsf{c} \parallel \pi_\mathsf{v} \parallel \mathsf{e})$ with security parameter 1^ℓ. We write $\Pr[(\pi_\mathsf{c} \parallel \pi_\mathsf{v} \parallel \mathsf{e})^{(\ell)} \mapsto \gamma]$ for the probability that a run of $(\pi_\mathsf{c} \parallel \pi_\mathsf{v} \parallel \mathsf{e})$, with security parameter 1^ℓ, belongs to γ.

Definition 3 ([18])**.** Let $P^* = P(A_H, q, V_H, k, \vec{p})$ be a protocol instantiation and let V and C be sets of processes as above. Let $\delta \in [0,1]$ and let γ be a goal. Then, P^* is *δ-coercion-resistant w.r.t.* γ, if there exists $\tilde{v} \in V$ such that the following conditions are satisfied:

(i) $\Pr[(\pi_\mathsf{c} \parallel \tilde{v} \parallel \mathsf{e})^{(\ell)} \mapsto \gamma]$ is overwhelming, as a function of ℓ, for every $\pi_\mathsf{c} \in C$.

(ii) $\Pr[(\pi_\mathsf{c} \parallel \mathsf{dum} \parallel \mathsf{e})^{(\ell)} \mapsto 1] - \Pr[(\pi_\mathsf{c} \parallel \tilde{v} \parallel \mathsf{e})^{(\ell)} \mapsto 1]$ is δ-bounded, as a function of ℓ, for every $\pi_\mathsf{c} \in C$.

Similar to the case of privacy, in Condition (ii) the difference is required to be δ-bounded instead of negligible, since there is always a non-negligible chance for the coercer to know for sure whether the coerced voter followed his instructions or not. For example, if one candidate got all votes, but the coercer told the coerced voter to vote for a different candidate, then the coercer knows that the coerced voter did not follow his instructions. In general, as in the case of privacy, δ is a function of $\vec{p}$, k, and the number $n = |V_H|$ of honest voters (see below and Section VI-D). Clearly, a small δ is preferable. Let us illustrate the meaning of δ by the following example. Assume that if π_c outputs 1, i.e., the coercer thinks that the coerced voter is following his instructions, then the coercer pays \$100 to the coerced voter, and otherwise, if π_c outputs 0, i.e., the coercer thinks that the coerced voter did not follow his instructions, he might harm the coerced voter. Now, if $\delta = 0.8$, then this means that if the coerced voter follows the instructions of the coercer, the coerced voter increases her chances of getting payed (not being harmed) by up to 80%. Conversely, by following the counter-strategy, the coerced voter drastically decreases her chances of getting payed and increases her chances of being harmed. This might be a strong incentive for the coerced voter to follow the instructions of the coercer.

While here we concentrated on the case for one coerced voter, the above definition in fact also applies to the setting of multiple coerced voters (see [18]).

Coercion-resistance of the Ideal Protocol. Since we will refer to the level of coercion-resistance of the ideal protocol in this paper, we recall the optimal level of coercion-resistance established in [18]. Similarly to the case of privacy, let $A^i_{\vec{r}}$ denote the probability 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. Also, let $M^*_{i,j} = \{\vec{r} \in Res : A^i_{\vec{r}} \leq A^j_{\vec{r}}\}$ and

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

Let γ_i be the goal of the coerced voter which is achieved if the coerced voter votes for candidate i, in case she is instructed by the coercer to vote (for some candidate). Note that coercion-resistance w.r.t. this goal does not imply that forced abstention attacks are prevented: If the coercer wants the coerced voter to abstain from voting, the coerced voter, when running her counter-strategy, does not need to vote in order to fulfil the goal. While for the ideal protocol a stronger goal which says that the coerced voter in any case votes for i could be considered, for ThreeBallot and VAV such a goal, which requires that forced abstention attacks are not possible, is too strong (see Section VI-D). The following theorem states that $\delta^i_{min}(n,k,\vec{p})$ is optimal for γ_i:

Theorem 3 ([18])**.** *Let $S = \mathsf{P}_{\mathsf{ideal}}(\{VA\}, q, n, k, \vec{p})$. Then, S is δ-coercion-resistant w.r.t. γ_i, where $\delta = \delta^i_{min}(n,k,\vec{p})$. Moreover, S is not δ'-coercion-resistant for any $\delta' < \delta$.*

We note that the level of privacy of the ideal protocol coincides with the level of coercion-resistance of the ideal protocol, if the goal of the coerced voter is to vote for the candidate with the smallest probability according to $\vec{p}$.

C. Privacy of ThreeBallot and VAV

In this section, we analyze the level of privacy provided by all variants of ThreeBallot and VAV described in Sections III and IV. In all cases, the presentation of the results follows the same structure: First, we define what we call an *essential view of the observer*, where we abstract away from some parts of the observer's full view in a given protocol run. Based on the notion of an essential view, we define the optimal level of privacy, δ, and state the result. Due to the similar structure, we introduce the necessary terminology and present the results "in parallel" for all protocol variants. We start with our modeling and security assumptions, which are largely the same for all variants.

Modeling and Security Assumptions. In our analysis, we assume that the observer can see whether a voter enters the voting booth. We also assume that an honest voter may reveal her (paper) receipt to the observer, after the voting phase is finished. However, to measure how much information an observer gains from the receipts of honest voters, we will also consider the case that the observer does not get to see the receipts of honest voters.

Moreover, we assume that the voting machine (the scanner) is honest; the bulletin board may be dishonest. Note that this assumption is indeed necessary for privacy: without this assumption, the observer gets to know how voters vote, as the voters disclose their votes to the machine. Even though the machine, at the moment a voter votes, might not know who the voter is, this information could be reconstructed from the order in which voters voted.

In our analysis of ThreeBallot, we focus on the case with two candidates, i.e., a case where the so-called short ballot assumption is fulfilled. It is well-known that without this assumption, ThreeBallot does not have a sufficient level of privacy and coercion-resistance (see, e.g., [26], [11]). The degradation of the level of coercion-resistance of the variant of ThreeBallot by Marneffe et al. in the multi-candidate case was formally studied in [18]. However, in our analysis of VAV we do not restrict the number of candidates.

By $\mathsf{P}^{s+}_{\mathsf{VAV}}$ and $\mathsf{P}^{s-}_{\mathsf{VAV}}$ we denote the simple variant of the VAV protocol (modeled as a protocol in the sense of Section II), where '$+$' and '$-$' indicate whether or not the honest voters reveal their receipts. Similarly, $\mathsf{P}^{p+}_{\mathsf{VAV}}$ and $\mathsf{P}^{p-}_{\mathsf{VAV}}$ denote the privacy enhanced variant of VAV, with/without receipts being revealed. As for ThreeBallot, we use $\mathsf{P}^{o+}_{\mathsf{TB}}$ and $\mathsf{P}^{o-}_{\mathsf{TB}}$ for the original variant and $\mathsf{P}^{p+}_{\mathsf{TB}}$ and $\mathsf{P}^{p-}_{\mathsf{TB}}$ for the variant by Marneffe et al.

Following our modeling and security assumptions, we consider instantiations of $\mathsf{P}^{s+}_{\mathsf{VAV}}$, $\mathsf{P}^{s-}_{\mathsf{VAV}}$, $\mathsf{P}^{p+}_{\mathsf{VAV}}$, $\mathsf{P}^{p-}_{\mathsf{VAV}}$, $\mathsf{P}^{o+}_{\mathsf{TB}}$, $\mathsf{P}^{o-}_{\mathsf{TB}}$, $\mathsf{P}^{p+}_{\mathsf{TB}}$ and $\mathsf{P}^{p-}_{\mathsf{TB}}$, where the parameters are chosen as follows: (i) the set $A_H = \{M\}$ of honest authorities contains the voting machine M only, (ii) the number q of (honest and dishonest) voters, (iii) some number n of honest voters, (iv) some number k of candidates, and (v) some probability distribution $\vec{p}$ on the candidates. In case of ThreeBallot, we assume that $k = 2$. For brevity of notation, we, for instance, simply write $\mathsf{P}^{s+}_{\mathsf{VAV}}$, instead of $\mathsf{P}^{s+}_{\mathsf{VAV}}(A_H, q, n, k, \vec{p})$.

We denote the set of all the instantiations described above by $\mathscr{S}$.

Views and essential views. The *view* of the observer in a protocol run contains (1) the random coins generated by the observer, (2) optionally, depending on the case under consideration, the receipts of the honest voters, after the voting phase is finished, and (3) all messages received from the interface of the dishonest parties (which the observer controls). The latter includes all dishonest voters and the bulletin board, containing the shuffled simple ballots with serial numbers. Note that the observer cannot directly see the information the honest voters obtain or the actions they perform in the voting booth.

In an *essential view* of the observer we abstract away from those parts of his view which are not relevant for distinguishing how voters vote, e.g., the serial numbers on the simple ballots, the order of the simple ballots on the bulletin board or the simple ballots of the dishonest voters (which are determined by the observer). The crucial part of the proof of Theorem 4, stated below, is to show that, indeed, the observer can without loss of generality base his decision solely on such essential views.

More precisely, if the observer cannot see the receipts of the voters, the essential view is defined to be the pure result of the election, as defined in Section VI-A. If the observer can see the receipts, the definition of an essential view depends on the system under consideration:

– *ThreeBallot:* The essential view of the observer consists of (i) three integers $n_{\substack{\times\\\times}}$, $n_{\substack{\times\\\circ}}$, and $n_{\substack{\circ\\\times}}$ indicating the number of the respective simple ballots on the bulletin board cast by honest voters, including the observed voter, and (ii) the receipt r of the voter under observation and (iii) integers $r_{\substack{\times\\\times}}$, $r_{\substack{\times\\\circ}}$, and $r_{\substack{\circ\\\times}}$, indicating the number of the respective receipts taken by the honest voters.
Note that from these numbers the number of $\substack{\circ\\\circ}$-ballots on the bulletin board and the number of $\substack{\circ\\\circ}$-receipts can easily be computed.

– *Privacy enhanced variant of VAV:* The essential view is just the pure result.

– *Simple variant of VAV:* The essential view of the observer consists of (i) integers n^A_j, n^V_j, for each candidate j, indicating the number of A- and V-ballots, respectively, on the bulletin board on which candidate j is marked, (ii) the receipt r of the voter under observation and (iii) integers r^A_j and r^V_j, indicating the number of A- and V-ballots, respectively, taken by the honest voters as receipts on which candidate j is marked.

By EV^S we denote the set of all essential views of the observer for the instantiation $S \in \mathscr{S}$.

Level of privacy. Let $S \in \mathscr{S}$. For an essential view $\rho \in EV^S$, and a candidate i, let $A^S_{\rho,i}$ denote the probability that, in a run of S, the essential view of the observer is ρ, given that the observed voter votes for i. For $i, j \in \{1, \dots, k\}$, let $M^S_{j,i} = \{\rho \in EV^S : A^S_{\rho,j} \le A^S_{\rho,i}\}$. Similar to the case of privacy for the ideal protocol, the intuition behind the definition of δ, given below, is the following: If the observer, given an essential view ρ, wants to decide whether the observed voter voted for candidate i or j, the best strategy for the observer is to opt for i, if $\rho \in M^S_{j,i}$, i.e. his (essential) view is more likely if the voter voted for candidate i.

Now, we are ready to express the function representing the level of privacy in the instantiation $S \in \mathscr{S}$:

$$\delta^S_{Priv}(n, k, \vec{p}) = \max_{i,j=1,\dots,k} \sum_{\rho \in M^S_{j,i}} \left(A^S_{\rho,i} - A^S_{\rho,j} \right).$$

The following theorem shows that $\delta^S_{Priv} = \delta^S_{Priv}(n, k, \vec{p})$ indeed is the optimal level of privacy achieved by the considered instantiations S of ThreeBallot and VAV.

Theorem 4. *Let $S \in \mathscr{S}$. The instantiation S achieves δ^S_{Priv}-privacy. Moreover, S does not achieve δ'-privacy for any $\delta' < \delta^S_{Priv}$.*

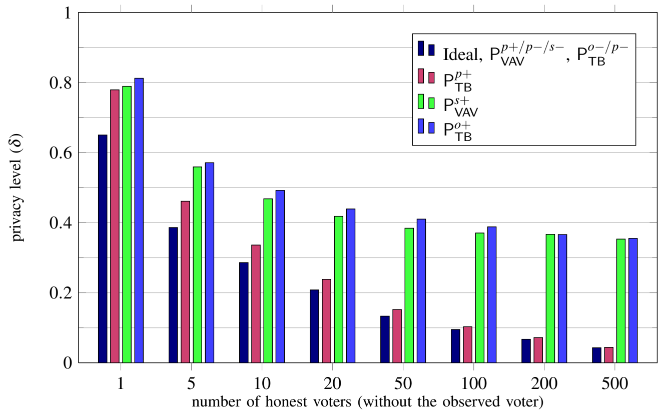

Fig. 3. Level of privacy (the smaller δ the higher privacy) for the considered variants of ThreeBallot with two candidates, $p_0 = 0.3$, $p_1 = 0.35$, $p_2 = 0.35$.

While the representation of δ^S_{Priv} is the same for every instantiation (except that the definitions of essential views differ), the proofs of Theorem 4 for the various instantiations differ significantly. In each case, we first show that the view of the observer can indeed be reduced to the corresponding essential view, and then, by combinatorial arguments, we show that δ^S_{Priv} is optimal. Due to space limitations, we do not present this proof here.

For every $S \in \mathscr{S}$, we have developed concrete formulas for $A^S_{\rho,i}$, which, in some cases, involved non-trivial combinatorial reasoning. These formulas allowed us to compute concrete values for δ^S_{Priv}, as depicted in Figure 3 for the case of two candidates. Note that election results are often published per polling station, with just a few hundred voters each.

As can be seen from Figure 3, the variants of the protocols where the observer does not get to see receipts of voters provide the ideal level of privacy. The privacy enhanced variant of VAV with receipts being revealed is ideal too: Intuitively, the reason for this is that the receipts taken by the honest voters are picked independently of the chosen candidates. Furthermore, the bulletin board does not leak any information about how a given voter voted, except for the bare result—the ballots of type A and V which cancel out are chosen independently of the voters' choices. The variant of ThreeBallot by Marneffe et al. is close to ideal. However, the level of privacy of the original variant of ThreeBallot and the simple variant of VAV is unacceptable. This is due to the receipts which, for these variants of the protocols, leak a lot of information about a vote. In case of VAV, for example, it is easy to see that with probability $\frac{1}{3}$, a voter takes the simple ballot as a receipt which exactly shows her choice, hence, δ can never drop below $\frac{1}{3}$.

D. *Coercion-Resistance of ThreeBallot and VAV*

In this section, we analyze the level of coercion-resistance provided by all variants of ThreeBallot and VAV described in Sections III and IV. We note that the level of coercion-resistance of the variant by Marneffe et al. [8] has already been established in [18]. However, the results for VAV and the one for the original variant of ThreeBallot are new.

As in the case of privacy, the presentation of the results follows the same structure for all protocol variants, which is why we again introduce the necessary terminology and present the results "in parallel" for all these variants. For coercion-resistance, we also use the notion of an essential view (although defined differently). In addition, we have to define the goal of the coerced voter and the counter-strategy.

Modeling and security assumptions. We make the same modeling and security assumptions as in the case of privacy and consider the same set $\mathscr{S}$ of concrete instantiations.

The goal of the coerced voter. Our analysis is w.r.t. the goal γ_i, for $i \in \{1,\ldots,k\}$, which is met if the coerced voter votes for candidate i, in case she is instructed by the coercer to vote for some candidate. Note that if the coerced voter is not instructed to vote, she cannot vote, as this fact would be observed by the coercer, who sees if the voter enters the voting booth (forced-abstention attack). Recall that for ThreeBallot, we assume $k = 2$.

Counter-strategy. We define the counter-strategy of the coerced voter for an instance $S \in \mathscr{S}$ as follows: The counter-strategy coincides with the dummy strategy dum 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 $C^S(Z,i)$, as defined next. (Recall that the goal of the coerced voter is to vote for i.) We define $C^S(Z,i)$ to be Z, if the pattern Z forms a valid vote for i. Otherwise, the definition of $C^S(Z,i)$ depends on the protocol under consideration:

- *ThreeBallot:* We define $C^S(Z,i)$ in such a way that it yields the same receipt as Z does. Moreover, it adjusts the two remaining ballots in such a way that the resulting multi-ballot is a valid vote for candidate i. By this requirement, $C^S(Z,i)$ is uniquely determined, except for two cases: $C^S((\substack{\times\\ \underline{\circ}}, \substack{\circ\\ \times}, \substack{\circ\\ \times}),1)$ and $C^S((\substack{\circ\\ \underline{\times}}, \substack{\times\\ \circ}, \substack{\times\\ \circ}),2)$. In the former case, for instance, one can take $(\substack{\times\\ \underline{\circ}}, \substack{\times\\ \circ}, \substack{\circ\\ \times})$ or $(\substack{\times\\ \underline{\circ}}, \substack{\times\\ \times}, \substack{\circ\\ \circ})$, or randomly pick one of the two, possibly based on some further information. For these cases, we define $C^S(Z,i)$ to choose one of the possible patterns uniformly at random.
- *VAV:* Similarly as in the case for ThreeBallot, $C^S(Z,i)$ is defined in such a way that it yields the same receipt as Z does. It adjusts the two remaining ballots in such a way that the resulting multi-ballot is a valid vote for candidate i. By this requirement, $C^S(Z,i)$ is uniquely determined, except for the case where the coercer demands a V-receipt for candidate i. In this case, one can mark an arbitrary candidate on the remaining two ballots. We demand that $C^S(Z,i)$ then marks candidate i also on the remaining ballots.

We use these strategies in the proof of Theorem 5. From the proof of this theorem it follows that these counter-strategies achieve the maximal level of coercion-resistance and, in this sense, are optimal for the coerced voter.

Essential views. The *essential view of the coercer* is defined as follows:

- *ThreeBallot:* If the coercer can see the receipts of honest voters, the essential view is defined just like the essential view of the observer in case of privacy (see Section VI-C), except that it does not contain the receipt of the coerced voter (as the coerced voter returns always the receipt demanded by the coercer). If the coercer cannot see the receipts, the essential view consists of the integers $n_{\substack{\times\\\times}}$, $n_{\substack{\times\\\circ}}$, and $n_{\substack{\circ\\\times}}$, representing the numbers of the respective simple ballots on the bulletin board.

- *Privacy enhanced variant of VAV:* Regardless of whether the coercer can see the receipts or not, an essential view of the coercer consists of two integers n_j^A and n_j^V for each candidate j, indicating the number of A- and V-ballots on the bulletin board, respectively, with candidate j marked.
- *Simple variant of VAV:* The essential view of the coercer consists of integers n_j^A and n_j^V, for each candidate j, as defined above and, if the coercer can see the receipts, additionally, two integers r_j^A and r_j^V, for each candidate j, indicating the number of A- and V-ballots, respectively, taken by the honest voters as receipts with candidate j marked.

By EV^S we denote the set of all essential views of the coercer for the instantiation $S \in \mathscr{S}$.

Level of coercion-resistance. Let $S \in \mathscr{S}$ and $\rho \in EV^S$. We define $A^S_{\rho,Z}$ to be the probability that, in a run of S, the choices made by the honest voters and the coerced voter result in the essential view ρ, given that the coerced voter chooses the pattern Z. The definition of the level of coercion-resistance, δ, now follows the same idea as in the case of privacy. We define $M^S_{Z,i} = \{\rho \in EV^S : A^S_{\rho,C(Z,i)} \leq A^S_{\rho,Z}\}$ to be the set of those essential views for which the coercer should accept the run and we define

$$\delta^S_{CR}(n,k,\vec{p}) = \max_Z \sum_{\rho \in M^S_{Z,i}} (A^S_{\rho,Z} - A^S_{\rho,C(Z,i)}). \qquad (2)$$

The following theorem shows that $\delta^S_{CR} = \delta^S_{CR}(n,k,\vec{p})$ indeed is the optimal level of coercion-resistance for the instantiation S, where the case in which S is the variant of ThreeBallot by Marneffe et al. was shown in [18].

Theorem 5. *Let $S \in \mathscr{S}$. Then S is δ^S_{CR}-coercion-resistant. Moreover, S is is not δ'-coercion-resistant for any $\delta' < \delta^S_{CR}$.*

Similar to the case of privacy, the details of the proofs for the different variants S of the protocols differ significantly. Due to lack of space, we omit the proofs in this extended abstract.

We developed concrete formulas for $A^S_{\rho,i}$ and $A^S_{\rho,C(Z,i)}$, which involved non-trivial combinatorial arguments, but allowed us to compute concrete values for δ^S_{CR}, as depicted in Figure 4 for the case of two candidates. To put these values in context, we present also the corresponding values for the variant of ThreeBallot and the ideal protocol studied in [18]. As we can see in Figure 4, for each protocol, with the exception of $\mathsf{P}^{p+}_{\mathsf{VAV}}$ and $\mathsf{P}^{p-}_{\mathsf{VAV}}$, the level of coercion-resistance is lower if the coercer can see the receipts. For $\mathsf{P}^{p+}_{\mathsf{VAV}}$ and $\mathsf{P}^{p-}_{\mathsf{VAV}}$ the level of coercion-resistance is the same. Intuitively, the reason for this is that, in $\mathsf{P}^{p+}_{\mathsf{VAV}}$, the information printed on a receipt is independent of the chosen candidate, which is also the case for $\mathsf{P}^{p+}_{\mathsf{TB}}$, but unlike $\mathsf{P}^{p+}_{\mathsf{TB}}$, no further information can be derived from the receipt in conjunction with the bulletin board. Altogether, under the same assumptions, VAV provides a better level of coercion-resistance than ThreeBallot, but both are still worse than the ideal protocol.

Some selected values for the multi-candidate case, namely 10 candidates, are depicted in Figure 5. To put these values in context, we present also the corresponding values for the ideal protocol and the variant of ThreeBallot by Marneffe et al. as studied in [18]. We can see that the (privacy enhanced) variant of VAV handles the case of multiple candidates much better than ThreeBallot, which for 10 candidate basically does not provide any coercion-resistance. (Recall that δ close to 1 means that the coercer can tell almost for sure whether the coerced voter followed his instructions or not.)

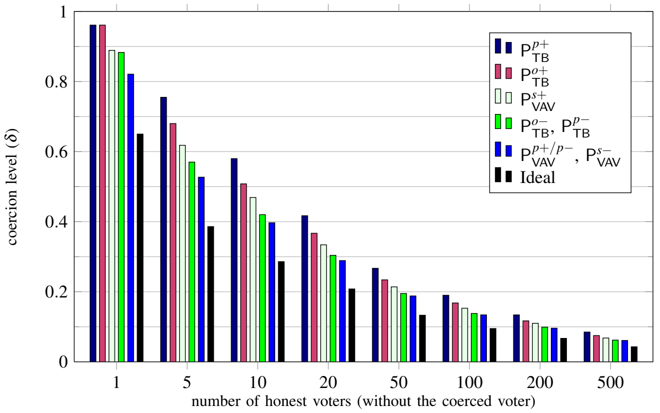

Fig. 4. Level of coercion (δ) for different protocols with two candidates, $p_0 = 0.3$, $p_1 = p_2 = 0.35$. The goal of the coerced voter is to vote for candidate 1.

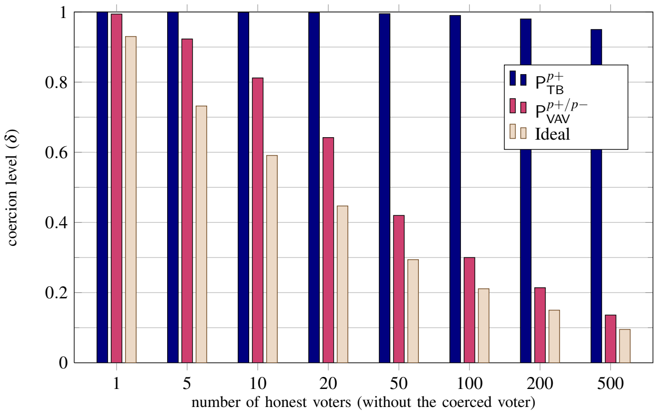

Fig. 5. The lower-bound of coercion-resistance (δ) for ThreeBallot in the variant by Marneffe et al. and the precise values for VAV in the privacy enhanced variant and the ideal voting protocol with 10 candidates, where an honest voter abstains from voting with probability $p_0 = 0.3$ and she chooses a candidate with probability $((1 - p_0)/10)$.

E. Relationship between Privacy and Coercion-resistance

As already mentioned in the introduction, one would expect that privacy and coercion-resistance are closely connected: If a protocol provides a poor level of privacy, i.e., an observer has a good chance of distinguishing whether an honest voter voted for candidate j or j', then this should give the coercer leverage to distinguish whether the coerced voter followed the coercer's instructions or not. Indeed some works suggest a close connection between coercion-resistance and privacy, most notable the already mentioned work by Moran and Naor [21] and work in an abstract, Dolev-Yao style approach [10], which, however, puts strong restrictions on the coercer and counter-strategies. The definitions used in these works yield yes/no-answers, rather than measuring the level of coercion-resistance and privacy.

Our case study of ThreeBallot and VAV demonstrates that the connection between privacy and coercion-resistance, in particular when it comes to measuring the level of privacy and coercion-resistance, is more subtle than what can be gathered from existing work.

One observation that comes out of our case study is that improving the level of privacy of a protocol in a natural way (e.g., by changing the way honest voters fill out ballots) can lead to a *worse* level of coercion-resistance. This is the case when going from the original variant of ThreeBallot to the "privacy enhanced" variant by de Marneffe et al., as can be seen from the results in Sections VI-C and VI-D (compare the levels of privacy of the systems $\mathsf{S}^{o+}_{\mathsf{TB}}$ and $\mathsf{S}^{p+}_{\mathsf{TB}}$, given in Figure 3, with those for coercion-resistance of the same systems in Figure 4). Clearly, in general, one does not expect privacy to imply coercion-resistance. One might, however, expect that improving privacy also improves coercion-resistance. In this sense, the illustrated effect is surprising. At the end of this section, we propose another (though artificial) variant of ThreeBallot which better explains and amplifies the described effect.

Another finding that comes out of our case study, which maybe more unexpected, is that the level of privacy of a protocol can be much lower than its level of coercion-resistance. This is so for the original variant of ThreeBallot and the simple variant of VAV, as can be seen from the results in Sections VI-C and VI-D (compare the level of privacy of $\mathsf{S}^{o+}_{\mathsf{TB}}$, given in Figure 3, with the level of coercion resistance of this system, given in Figure 4; similarly for the system $\mathsf{S}^{s+}_{\mathsf{VAV}}$). The reason behind this phenomenon is basically that the counter-strategy hides the behavior of the coerced voter, including her vote, better than the honest voting program hides the vote. Conversely, one could say that the honest voting program is "suboptimal" in hiding the way the voter voted. In the original variant of ThreeBallot and the simple variant of VAV, a receipt an honest voter obtains indeed discloses more information than necessary. The following simple, but unlike ThreeBallot and VAV, artificial example, carries this effect to extremes: Consider the ideal voting protocol which collects all votes and publishes the correct result. Now imagine a voting protocol in which voters use the ideal voting protocol to cast their vote, but where half of the voters publish how they voted (e.g., based on a coin flip). Clearly, the level of privacy this protocol provides is very low, namely $\delta \geq \frac{1}{2}$. However, a coerced voter can be more clever and simply lie about how she voted. This protocol indeed provides a high level of coercion-resistance, namely $\delta \approx \delta^{i}_{min}(n/2, k, \vec{p})$ (see Section VI-B). Below we also provide a slightly more subtle example based on ThreeBallot.

In case the counter-strategy does not "outperform" the honest voting program (or conversely, the honest voting program does not leak more information than the counter-strategy), one would expect that if a voting system provides a certain level of coercion-resistance, then it provides at least the same level of privacy. We now show that this is indeed true.

We first have to define what it means for the counter-strategy to outperform the honest voting program.

For this purpose, let $P^* = P(A_H, q, V_H, k, \vec{p})$ be a protocol instantiation, with sets V and C of processes as in Definition 3. As usual, with $\hat{\pi}_{\mathsf{v}}(j)$ we denote the honest voting program voting for candidate j and with γ_i we denote the goal which contains all runs in which the coerced voter voted for candidate i (or the weaker goal, where this is required, only if the voter is instructed by the coercer to vote for some candidate). Let $\tilde{v}_i$ be a counter-strategy that tries to achieve this goal. Let π^{j}_{c} be a process of the coercer which only connects to dum, but does not use any other part of the interface the coercer can connect to, and which simply simulates the program of an honest voter voting for candidate j. Clearly, the systems $(\pi^{j}_{\mathsf{c}} \parallel \mathsf{dum})$ and $\hat{\pi}_{\mathsf{v}}(j)$ are identical from the point of view of an external observer.

Now, informally speaking, for $\tilde{v}_i$ to *not* outperform $\hat{\pi}_{\mathsf{v}}(i)$ we require that from the point of view of an external observer, $\tilde{v}_i$, if instructed to vote for some candidate j by following the honest program, behaves like $\hat{\pi}_{\mathsf{v}}(i)$. Recall that, the program π_{o} of an *external observer* may output 0 or 1 on some designated channel and may use the same communication interface as the coercer, except for connecting to dum, i.e., there is no direct communication between an external observer and the coerced voter. Now, formally we say that $\tilde{v}_i$ *does not outperform* $\hat{\pi}_{\mathsf{v}}(i)$, if for all programs π_{o} of the external observer we have that

$$\Pr[(\pi_{\mathsf{o}} \parallel \pi^{j}_{\mathsf{c}} \parallel \tilde{v}_i \parallel \mathsf{e})^{(\ell)} \mapsto 1] - \Pr[(\pi_{\mathsf{o}} \parallel \hat{\pi}_{\mathsf{v}}(i) \parallel \mathsf{e})^{(\ell)} \mapsto 1]$$

is negligible as a function in the security parameter 1^{ℓ}.

Now, we can state that, under the assumption that the counter-strategy does not outperform the honest voting program, we have that if a protocol is δ-coercion-resistant, then it also achieves δ-privacy. In the following theorem, we say that P^* *is δ-coercion-resistant w.r.t. γ_i and $\tilde{v}_i$*, if $\tilde{v}_i$ can be used to show δ-coercion-resistant of P^* w.r.t. γ_i. For simplicity, we focus here on the case of single-voter coercion-resistance/privacy, but this result can easily be lifted to the multi-voter case.

Theorem 6. *Let P^* be δ-coercion-resistant w.r.t. γ_i and $\tilde{v}_i$, for every candidates i. Then, if $\tilde{v}_i$ does not outperform $\hat{\pi}_{\mathsf{v}}(i)$, for every candidate i, then P^* achieves δ-privacy.*

Proof sketch: Let $j, j' \in \{1,\ldots,k\}$ and let π_{o} be a program of the observer, in the sense of Section VI-A. We define $\pi_c = (\pi_{\mathsf{o}} \parallel \pi_{\mathsf{c}}^{j})$ to be a coercer program. Recall from Section VI-B that the communication interface of a coercer is that of an observer, except that a coercer can also communicate with the coerced voter. Therefore, π_c indeed is a coercer program. Since P^* is δ-coercion-resistant w.r.t. $\gamma_{j'}$ and $\tilde{v}_{j'}$, we know that $\mathsf{Pr}[T \mapsto 1] - \mathsf{Pr}[\tilde{T} \mapsto 1]$ is δ-bounded for $T = (\pi_c \parallel \mathsf{dum} \parallel \mathsf{e})^{(\ell)}$ and $\tilde{T} = (\pi_c \parallel \tilde{v}_{j'} \parallel \mathsf{e})^{(\ell)}$.

Now, by the definition of π_{c}^{j}, we know that $\mathsf{Pr}[T \mapsto 1] = \mathsf{Pr}[T' \mapsto 1]$, for $T' = (\pi_{\mathsf{o}} \parallel \hat{\pi}_{\mathsf{v}}(j) \parallel \mathsf{e})^{(\ell)}$. Moreover, because $\tilde{v}_{j'}$ does not outperform $\hat{\pi}_{\mathsf{v}}(j')$, we have that $\mathsf{Pr}[\tilde{T} \mapsto 1] - \mathsf{Pr}[\tilde{T}' \mapsto 1]$ is negligible for $\tilde{T}' = (\pi_{\mathsf{o}} \parallel \hat{\pi}_{\mathsf{v}}(j') \parallel \mathsf{e})^{(\ell)}$. It follows that $\mathsf{Pr}[T' \mapsto 1] - \mathsf{Pr}[\tilde{T}' \mapsto 1]$ is δ-bounded, which proves that P^* achieves δ-privacy. ■

It turns out that for many voting protocol which have been analyzed with respect to coercion-resistance, the (optimal) counter-strategies indeed do not outperform the honest voting program of the respective protocol. In particular, it is not hard to check that at least the following protocols satisfy the condition in Theorem 6: the Bingo voting system [3] (see [18]), Scantegrity II [5] (see [17]), the JCJ voting protocol [13] and the Civitas voting system [7] (see also [16]). Also the "privacy enhanced" variants of ThreeBallot and VAV satisfy the condition (but clearly not the other variants of ThreeBallot and VAV we considered, since, as mentioned, for these variants the level of coercion-resistance is higher—δ is smaller—than the level of privacy.)

For such protocols, once δ-coercion-resistance is proven, by Theorem 6 we obtain δ-privacy for free. In case the level of coercion-resistance corresponds to the ideal one—as, e.g., proven for the Bingo voting system and Scantegrity II in [18] and [17]—, by Theorems 2 and 3, the level of privacy is ideal as well. However, in general, the actual level of privacy might be better than what can be concluded from the theorem, with the "privacy enhanced" variants of ThreeBallot and VAV being examples.

We conclude this section with the postponed examples mentioned above.

Example (Improving Privacy Significantly Lowers the Level of Coercion-Resistance). We consider the following variant of ThreeBallot. An honest voter is supposed to submit either $\left(\genfrac{}{}{0pt}{}{\mathsf{x}}{\underline{\mathsf{x}}},\genfrac{}{}{0pt}{}{\mathsf{x}}{\mathsf{o}},\genfrac{}{}{0pt}{}{\mathsf{o}}{\mathsf{o}}\right)$ or $\left(\genfrac{}{}{0pt}{}{\mathsf{x}}{\underline{\mathsf{x}}},\genfrac{}{}{0pt}{}{\mathsf{o}}{\mathsf{x}},\genfrac{}{}{0pt}{}{\mathsf{o}}{\mathsf{o}}\right)$ (according to her favorite candidate). This scheme is ideal in terms of privacy, because the bulletin board and the receipts do not leak any information apart from the pure result of the election. However, this scheme provides no coercion-resistance whatsoever: When the coerced voter is instructed to submit $\left(\genfrac{}{}{0pt}{}{\mathsf{o}}{\underline{\mathsf{x}}},\genfrac{}{}{0pt}{}{\mathsf{x}}{\mathsf{x}},\genfrac{}{}{0pt}{}{\mathsf{o}}{\mathsf{o}}\right)$ (which is allowed but never done by honest voters), but wants to vote for candidate A, she would have to submit $\left(\genfrac{}{}{0pt}{}{\mathsf{o}}{\underline{\mathsf{x}}},\genfrac{}{}{0pt}{}{\mathsf{x}}{\mathsf{o}},\genfrac{}{}{0pt}{}{\mathsf{x}}{\mathsf{o}}\right)$. But then, as all the honest voters submit $\left(\genfrac{}{}{0pt}{}{\mathsf{x}}{\underline{\mathsf{x}}},\genfrac{}{}{0pt}{}{\mathsf{x}}{\mathsf{o}},\genfrac{}{}{0pt}{}{\mathsf{o}}{\mathsf{o}}\right)$ or $\left(\genfrac{}{}{0pt}{}{\mathsf{x}}{\underline{\mathsf{x}}},\genfrac{}{}{0pt}{}{\mathsf{o}}{\mathsf{x}},\genfrac{}{}{0pt}{}{\overline{\mathsf{o}}}{\mathsf{o}}\right)$, the coercer could easily detect that he was cheated, by counting the number of ballots of type $\genfrac{}{}{0pt}{}{\mathsf{o}}{\mathsf{o}}$ on the bulletin board.

Example (Improving Coercion Resistance Significantly Lowers the Level of Privacy). We consider the following variant of ThreeBallot. In order to vote for candidate A, an honest voter is supposed to submit $\left(\genfrac{}{}{0pt}{}{\mathsf{x}}{\underline{\mathsf{x}}},\genfrac{}{}{0pt}{}{\mathsf{x}}{\mathsf{o}},\genfrac{}{}{0pt}{}{\mathsf{o}}{\mathsf{o}}\right)$ or $\left(\genfrac{}{}{0pt}{}{\mathsf{x}}{\underline{\mathsf{o}}},\genfrac{}{}{0pt}{}{\mathsf{x}}{\mathsf{o}},\genfrac{}{}{0pt}{}{\mathsf{o}}{\mathsf{x}}\right)$, each with probability $\frac{1}{2}$. Analogously for a voter that wants to vote for candidate 2. By this, in 50% of the cases, an honest voter reveals her vote, namley in the case where she does not have $\genfrac{}{}{0pt}{}{\mathsf{x}}{\mathsf{x}}$ as receipt. That means that this variant provides a very low level of privacy ($\delta \geq 0.5$), i.e. the observer can with quite high probability tell which candidate a given voter voted for. However, this variant is not that bad in terms of coercion-resistance, as here, the coerced voter is not bound to follow the honest strategy and can choose patterns in a more clever way. In fact, we can use here the same counter-strategy we used previously—take the receipt required by the coercer and adjust the remaining ballots to form a valid vote for the favorite candidate. Note that, following this strategy, the coerced voter may submit patterns which are valid but never chosen by the program of an honest voter. For this counter-strategy, although the coercer might learn approximatively half of the votes of the honest voters, the actual vote of the coerced voter is still hidden behind the votes of the honest voters that submitted $\left(\genfrac{}{}{0pt}{}{\mathsf{x}}{\underline{\mathsf{x}}},\genfrac{}{}{0pt}{}{\mathsf{x}}{\mathsf{o}},\genfrac{}{}{0pt}{}{\mathsf{o}}{\mathsf{o}}\right)$ or $\left(\genfrac{}{}{0pt}{}{\mathsf{x}}{\underline{\mathsf{x}}},\genfrac{}{}{0pt}{}{\mathsf{o}}{\mathsf{x}},\genfrac{}{}{0pt}{}{\mathsf{o}}{\mathsf{o}}\right)$, i.e. that did not reveal their votes to the coercer. This results in a reasonably small δ.

VII. Conclusion

In this paper, we presented new insights into central security properties, namely verifiability, privacy, and coercion-resistance. Our findings, in part, come from a case study, in which we precisely measure the level of verifiability, privacy, and coercion-resistance of different variants of ThreeBallot and VAV proposed in the literature.

For verifiability we have demonstrated that the combination of individual and universal verifiability is, unlike commonly believed, insufficient to provide overall/global verifiability. Our case study shows that the main problem with individual and universal verifiability is that these notions ignore that dishonest authorities/voters can break the integrity of ballots of honest voters by ill-formed ballots. We therefore advocate the concept of *global verifiability*, as captured by the definition of verifiability in [19] and used in the present paper.

We also demonstrated that the relationship between privacy and coercion-resistance is more subtle than what can be gathered from the literature. Our case study highlighted interesting phenomena for existing protocols: i) improving privacy may degrade the level of coercion-resistance, ii) the level of coercion-resistance may be higher than the level of privacy. The latter is due to the fact that the counter-strategy a coerced voter uses maybe "smarter" in hiding information than the honest voting program. For the case that this is not so, we were able to prove that δ-coercion-resistance implies δ-privacy. As discussed in Section VI-E, for many protocols, the counter-strategy does indeed not outperform the honest voting program. We conjecture that if it does, then it should be possible to improve the honest voting program.

Besides these general findings on verifiability, privacy, and coercion-resistance, our case study also provides the first comprehensive picture on the security of prominent voting systems, ThreeBallot and VAV.

Acknowledgment. This work was partially supported by *Deutsche Forschungsgemeinschaft* (DFG) under Grant KU 1434/5-1 and KU 1434/6-1.

References

[1] B. Adida and C.A. Neff. Ballot Casting Assurance. In *USENIX/ACCURATE Electronic Voting Technology (EVT 2006)*, 2006.

[2] 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.

[3] 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.

[4] D. Chaum. Elections with unconditionally-secret ballots and disruption equivalent to breaking rsa. In *Advances in Cryptology – Eurocrypt '88*, volume 330 of *Lecture Notes in Computer Science*, pages 177–182. Springer, 1988.

[5] 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.

[6] D. Chaum, P.Y.A. Ryan, and S. Schneider. A Practical, Voter-verifiable Election Scheme. In *Proceedings of the 10th European Symposium on Research in Computer Security (ESORICS 2005)*, volume 3679 of *Lecture Notes in Computer Science*, pages 118–139. Springer, 2005.

[7] 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.

[8] 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.

[9] S. Delaune, S. Kremer, and M. D. Ryan. Verifying Privacy-type Properties of Electronic Voting Protocols. *Journal of Computer Security*, 17(4):435–487, 2009.

[10] 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.

[11] 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.

[12] M. Hirt and K. Sako. Efficient receipt-free voting based on homomorphic encryption. In B. Preneel, editor, *Advances in Cryptology – EUROCRYPT 2000*, volume 1807 of *Lecture Notes in Computer Science*, pages 539 – 556. Springer, 2000.

[13] 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.

[14] Steve Kremer, Mark Ryan, and Ben Smyth. Election Verifiability in Electronic Voting Protocols. In Dimitris Gritzalis, Bart Preneel, and Marianthi Theoharidou, editors, *15th European Symposium on Research in Computer Security (ESORICS2010)*, volume 6345 of *Lecture Notes in Computer Science*, pages 389–404. Springer, 2010.

[15] 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.

[16] R. Küsters and T. 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.

[17] R. Küsters, T. Truderung, and A. 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.

[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. Verifiability, Privacy, and Coercion-Resistance: New Insights from a Case Study. Technical report, University of Trier, 2011. Available at http://infsec.uni-trier.de/publications.html.

[21] 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.

[22] 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.

[23] 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.

[24] B. Riva and A. Ta-Shma. Bare-Handed Electronic Voting with Preprocessing. In *USENIX/ACCURATE Electronic Voting Technology (EVT 2007)*, 2007.

[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] Charlie E. M. Strauss. A critical review of the triple ballot voting system, part 2: Cracking the triple ballot encryption. http://www.cs.princeton.edu/~appel/voting/Strauss-ThreeBallotCritique2v1.5.pdf, October 8, 2006. Draft V1.5.

Appendix

A. Proof Sketch of Theorem 1

For the first condition of Definition 1, we have to show that whenever, in a run of the system, the machine and the bulletin board are honest, *ver* accepts this run with overwhelming probability. This is easy to see: If the machine and the bulletin board are honest in a run of the system, only well-formed ballots are sent to the bulletin board, (with overwhelming probability) no serial numbers occur twice, and the bulletin board correctly displays the ballots received from the machine. Now, by the definition of the verifier, it follows that *ver* accepts such a run.

For the second condition of Definition 1, we have to show that the probability that the system produces a run which is accepted by *ver*, even though the goal is violated, is bounded by δ^{p}_{Ver} or δ^{o}_{Ver}, respectively. In such a run, since it is accepted by *ver*, the bulletin board must be consistent. Furthermore, because the goal is violated, there must exist a candidate, say candidate i, such that the sum of all votes of honest voters for all candidates except i is at least $\ell+1$.

As we have already shown, the machine can safely change m votes. Therefore, in order to violate the goal γ_ℓ, it remains to change $k' = \ell + 1 - min(\ell+1, m)$ votes of honest voters that did not vote for candidate i. One can verify that the best (the safest) way of doing this is to change k' multi-ballots cast by k' different honest voters, who voted for some $j \neq i$, in the following way: A dishonest party (the voting machine or the bulletin board) chooses one simple ballot of such a voter with marked j-th position, but not i-th position, and replaces it by a similar ballot, but with the markings on the i-th and j-th position swapped (one can show that this is always possible). Every time this is done, the probability that this is detected by an honest voter is $\frac{1}{6} \cdot p_{check}$ in the system $\mathsf{S}^{p}_{\mathsf{TB}}$ and $\frac{1}{3} \cdot p_{check}$ in $\mathsf{S}^{o}_{\mathsf{TB}}$. These probabilities can be computed by some elementary calculation. As it must be done k' times and, as mentioned, there must exist $\ell + 1$ voters who voted not for candidate i, we conclude that the probability that the goal γ_ℓ is violated and the observer accepts the run is bounded by δ^{p}_{Ver} and δ^{o}_{Ver}, respectively, and that these bounds are optimal.

B. Verifiability of VAV

Let $\mathsf{P}^{s}_{\mathsf{VAV}}$ and $\mathsf{P}^{p}_{\mathsf{VAV}}$ denote the VAV protocol in the simple variant and the privacy enhanced variant, respectively. Based on analogous assumptions as those for ThreeBallot (see Section V-B), it is straightforward to formally define the protocol instantiations $\mathsf{S}^{s}_{\mathsf{VAV}} = \mathsf{P}^{s}_{\mathsf{VAV}}(\{ver\}, q, V_H, k, \vec{p})$ of $\mathsf{P}^{s}_{\mathsf{VAV}}$ and $\mathsf{S}^{p}_{\mathsf{VAV}} = \mathsf{P}^{p}_{\mathsf{TB}}(\{ver\}, q, V_H, k, \vec{p})$ of $\mathsf{P}^{p}_{\mathsf{VAV}}$.

To state the following theorem, we need to introduce the following notation. For a given run of the protocol, we denote by A the set of those candidates j for which the sum of all votes of honest voters for all candidates except j is at least $\ell + 1$. Moreover, by X_j we denote the number of multi-ballots in a run submitted by honest voters for which the following holds: i) The multi-ballot forms a vote for a candidate different from j and ii) On the multi-ballot, not the same candidate is marked on all three simple ballots; these multi-ballots can safely be changed to votes for j, as explained in Section IV. Finally, we define $p_r = \mathsf{Pr}[A \neq \emptyset \text{ and } \max_{j \in A} X_j = r]$, where the probability is over runs of the protocol. (Note that p_r only depends on choices made by honest voters.)

Theorem 7. *Let B be the set containing the voting machine and the bulletin board and $x \in \{s, p\}$. The goal γ_ℓ is guaranteed in $\mathsf{S}^{x}_{\mathsf{VAV}}$ by B and δ^{x}_{VerVAV}-verifiable by a, where $\delta^{s}_{VerVAV} = \delta^{o}_{Ver}$ with δ^{o}_{Ver} as in Theorem 1 and*

$$\delta^{p}_{VerVAV} = \sum_{r=0}^{n} p_r \left(1 - \frac{1}{4} \cdot p_{ckeck}\right)^{\max(\ell+1-r-m,0)} ,$$

where m is the number of dishonest voters and p_r is defined as above.

The statement for $\mathsf{S}^{s}_{\mathsf{VAV}}$ follows as in the proof of Theorem 1. The intuition behind the statement for $\mathsf{S}^{p}_{\mathsf{VAV}}$ is the following. The best strategy of the adversary for violating the goal in a given run is as follows: First, he determines those candidates j such that the number of submitted multi-ballots not for j is bigger than ℓ. (Those candidates form the set A introduced above). Second, among those candidates, he determines a candidate j for which the number X_j is maximal. The probability of this number being r is p_r. Now, if $X_j = r$, then the adversary can safely change r votes. If there are some further votes to be changed (i.e. if $r < \ell + 1$), then the adversary can use dishonest voters to change additional m votes, as described in Section V-B. Only if there are still some votes to be changed (i.e. $r + m < \ell + 1$), the adversary has to change further ballots, namely $\ell + 1 - r - m$, which is detected with probability $\frac{1}{4} \cdot p_{check}$ for each ballot. Hence, the probability that the adversary goes undetected when changing $\ell + 1 - r - m$ ballots is $\left(1 - \frac{1}{4} \cdot p_{ckeck}\right)^{\ell+1-r-m}$.