Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.084123

ARTICLE

Somewhat Deniable Voting: Coercion-Resistant Electronic Voting Scheme with Privacy Preservation Property

Mingxuan Jia, Chenglong Shi, Yang Ye, Wen Huang and Jian Peng*

College of Computer Science, Sichuan University, Chengdu, China

*Corresponding Author: Jian Peng. Email: jianpeng@scu.edu.cn

Received: 16 April 2026; Accepted: 11 July 2026; Published: 13 August 2026

ABSTRACT: With the development of electronic voting schemes, traditional on-site voting is gradually being replaced because of its organizational inconveniences. However, electronic voting takes place in an uncontrollable environment, which opens up the possibility of voter coercion. In this paper, we propose an electronic voting scheme with the property of coercion resistance and privacy preservation. In particular, we introduce the concept of somewhat deniable voting. Somewhat deniable voting gives up verifiability to some extent but not all in exchange for coercion resistance under the condition that the election result remains unchanged. Besides, a somewhat deniable voting scheme is a local differential privacy mechanism with an adjustable privacy guarantee from the perspective of local differential privacy. Thus, the somewhat deniable voting scheme is a new kind of voting scheme that lies between real-name voting schemes and anonymous voting schemes. Specifically, the somewhat deniable voting scheme gradually becomes an anonymous voting scheme from a real-name voting scheme by adjusting a parameter named deniability confidence. To demonstrate somewhat deniable voting, we design a scheme for somewhat deniable voting and perform extensive experiments to evaluate the proposed voting scheme.

KEYWORDS: E-voting; coercion resistance; privacy preservation; local differential privacy

1 Introduction

How to vote is a very important and serious topic in a democratic society [1]. For example, in the 2024 US election between Trump and Harris which has attracted the attention of the world, how to vote is an important dispute. In particular, the vote cast by post may be invalid for many reasons such as coercion, forgery, and so on. Meanwhile, traditional on-site voting suffers from inherent limitations, such as organizational complexity and interference susceptibility. These limitations motivate the need for more flexible approaches, namely electronic voting systems [2,3].

For the electronic voting schemes, the voting process takes place in an uncontrolled environment, which may expose voters to various forms of coercion [4]. For example, voters may be pressured by parents, friends, or other individuals to vote for a candidate whom voters do not actually support. Likewise, voters may face workplace-related coercion, where employers utilize employment as leverage to force voters vote for a particular candidate. Consequently, coercion resistance becomes an important property of electronic voting schemes.

To achieve coercion resistance, there are mainly three kinds of solutions, including fake credentials, deniable vote updating, and masking [5]. For fake credentials, voters can cast their votes using fake credentials without any authentication error if voters are coerced. Voters can cast their votes for their

preferred candidate using real credentials when adversaries are not present. For deniable vote updating, voters can cast another new vote to overwrite the previous vote that the voters cast due to a threat. For masking, voters can cast votes for their preferred candidate while letting adversaries think that the same vote is a vote for another candidate.

The above coercion-resistant methods have a common limitation: any counter-method used for coercion resistance is also known to adversaries. Specifically, no matter what kind of coercion-resistant method is applied in an electronic voting scheme, the used coercion-resistant method needs to be public for all voters so that voters are aware of the existence of a coercion-resistant method and how to use the coercion-resistant method. However, adversaries also know the existence of the coercion-resistant method and how to use the coercion-resistant method when voting.

The key point is whether voters have the ability to verify their votes. Assume that an adversary tries to force voters to cast their votes for the adversary's preferred candidate. The adversary knows the existence of a coercion-resistant method, and voters have the ability to verify their votes. The adversary naturally forces voters to verify their votes to ensure that the voters indeed follow the adversary's instructions. Therefore, the adversary can still coerce voters even if the coercion-resistant method is implemented by advanced cryptography methods as long as voters have the ability to verify their votes. The reason is that adversaries and voters have the same information aspect to coercion resistance. If any evidence confirms that a vote has been counted for a particular candidate, the adversary can utilize the evidence to verify whether voters follow his instructions. The key difference between the cryptography scenario and the coercion-resistant scenario is shown in Fig. 1.

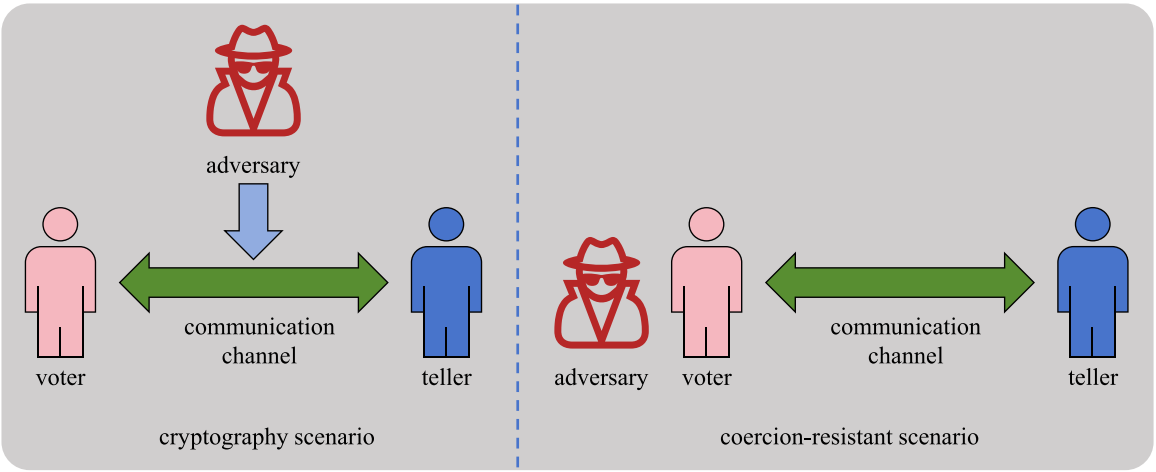

Figure 1: Difference between cryptography scenario and coercion-resistant scenario.

Based on the above observation, voters must give up verifiability for coercion resistance. However, verifiability is also an important property of an electronic voting scheme [6,7]. Therefore, we introduce a new concept of somewhat deniable voting to relieve the tension between verifiability and coercion resistance. Specifically, in a somewhat deniable voting scheme, each vote for one candidate may be changed to a vote for another candidate with a certain probability under the condition that the election results remain unchanged. So, the voters keep verifiability to some extent while reasonably denying their votes. For example, an adversary wants to force a voter to cast one vote for candidate Eve while the voter actually wants to cast one vote for candidate Alice. The voter can just cast his vote for Alice and has the ability to verify which candidate gets his vote. If the adversary forces the voter to verify the vote and the vote is proven for Alice,

not for Eve, in the verification process, the voter can reasonably deny it by saying that the voter did cast his vote for Eve, while the voter's vote is changed to a vote for Alice.

Furthermore, with the rapid development of differential privacy, an increasing number of privacy-preserving applications have adopted this theoretical framework to provide robust privacy protection across various domains, such as private image synthesis, model security auditing, vehicle driving safety, and healthcare data analytics. Inspired by these advancements, our electronic voting scheme bridges the concept of somewhat deniable voting with local differential privacy to formally quantify privacy guarantees of our scheme. Our main contributions are as following:

- We introduce the concept of somewhat deniable voting to balance coercion resistance and verifiability. The somewhat deniable voting gives up verifiability to some extent but not all in exchange for coercion resistance, under the condition that the election result remains unchanged. One concrete scheme of somewhat deniable voting is designed to demonstrate the concept and to evaluate the performance.
- The designed scheme is a new type of voting scheme. In particular, the designed scheme is not a real-name voting scheme or an anonymous voting scheme, but a voting scheme lying between a real-name voting scheme and an anonymous voting scheme. How close the designed scheme is to a real-name voting scheme or an anonymous voting scheme is controlled by a parameter. Therefore, the designed voting scheme is a very flexible scheme in terms of privacy preservation.
- By leveraging the rigorous mathematical theorems of local differential privacy, our electronic voting scheme provides formalized guarantees for different protection levels. Specifically, our scheme utilizes the differential privacy parameter to quantitatively control the adversary's ability to verify voter compliance, thereby explicitly characterizing the coercion resistance capability of the electronic voting scheme under the defined threat model.

2 Related Work

There are mainly three approaches to designing the electronic voting scheme with coercion resistance, including fake credentials, deniable vote updating, and vote masking.

The idea of fake credentials is that voters can cast their votes with fake credentials if they are coerced to cast votes. In particular, voters obtain multiple credentials when they register, and only one real credential can authenticate voters to the voting system. Voters can cast votes with fake credentials if they are coerced by adversaries while the voting system does not give any authentication error. However, these votes cast using fake credentials will not be counted in the tallying procedure. Voters can cast votes for their preferred candidate with the real credential when they are not under the adversary's observation. There are many electronic voting schemes based on fake credentials. For example, Aziz [8] designed a voting scheme that provides voters with multiple voting login accounts. When under coercion, a voter can provide fake credentials by using them to cheat the adversary. Clarkson et al. [9] designed a method named Civitas with a fake credential generation algorithm. The method utilizes the voter's private designation key to produce credentials that are indistinguishable to the adversary. The schemes proposed by Iovino et al. [10] and Neji et al. [11] both adopt Civitas and allow voters to simply use a fake credential to cast a vote in the presence of an adversary during the voting phase. Zaghloul et al. [12] proposed a scheme for encrypting votes using private keys. When voters are coerced by adversaries, they can use a different private key to decrypt the vote and obtain a fake credential to deceive the adversary. Merino et al. [13] investigated whether ordinary voters can comprehend and effectively utilize fake credentials. They demonstrated the potential of fake credentials as a viable mitigation strategy to alleviate the vulnerability of online voting. Yin et al. [14] proposed a scalable coercion-resistant blockchain voting scheme. Their scheme introduces a novel technique named dummy voting power that associates encrypted fake credentials with zero voting power to achieve linear complexity

$O(n)$. Giustolisi and Garjan [15] proposed a scheme utilizing server-generated noise ballots and a novel verifiable re-randomization procedure. Their approach successfully excludes fake credentials without leaking any information to the adversary. Sarier [16] proposed a scheme in which every voter has a valid Bitcoin wallet and an invalid Bitcoin wallet. The voter can cast a vote using the invalid Bitcoin wallet as a fake credential to cheat the adversary under coercion.

The idea of deniable vote updating is that voters can cast a new vote to overwrite an old vote that was cast because of coercion. In particular, voters may cast a vote under the coercion of adversaries. However, the voters can cast another vote for their preferred candidates to overwrite the old vote, while the voters can deny the action of revoting. There are many electronic voting schemes based on deniable vote updating. For example, Bernhard et al. [17] investigated the formal definition of probabilistic receipt-freeness in electronic voting schemes with deniable voting updating, such as KTV-Helios [18]. Heiberg et al. [19] proposed a scheme with deniable voting updating in which the privacy of voters' votes is not released. This scheme has been applied to real national elections. Locher et al. [20] achieved coercion resistance based on a new mechanism for deniable vote updating. In particular, they utilized two types of mix networks to guarantee that vote updates remain undetectable by the adversary. Giustolisi et al. [21] proposed a novel technique named flexible vote updating. Their scheme makes deniable revoting schemes effective even if an adversary coerces a voter at the very last minute of the voting phase. Müller et al. [22] proposed a scheme named DeVoS, which utilizes a posting trustee to generate dummy votes. This scheme achieves both deniable vote updating and participation privacy while preserving verifiability. Farzaliyev and Garjan [23] proposed an end-to-end verifiable remote voting protocol that directly supports deniable vote updating through verifiable re-voting. Their concept implementation outperforms prior STARK-based protocols [24] in terms of scalability.

The idea of vote masking is: voters cast votes for their preferred candidates while letting adversaries think that the same vote is a vote for adversaries' preferred candidates [25]. For example, the voting system provides voters with a key b such that the same vote is for different candidates if the value of the key is different. In particular, voters can cast votes for their preferred candidate Alice by $vote = M(b, v_{alice})$ and then calculate b' such that $M(b, v_{Alice}) = M(b', v_{Eve})$. Voters can make adversaries believe that the voters cast a vote for Eve by claiming that the value of the key is b'. There are some electronic voting schemes with coercion resistance based on vote masking such as [26,27]. Chen et al. [28] proposed that the voter can intentionally input an altered PIN into the smart card to cast a masked vote that complies with the coercer's demand for masking the voter's true intention. Finogina and Herranz [29] proposed that the voter could utilize the simulatable property of zero-knowledge proofs to generate a computationally indistinguishable fake record based on the concept of vote masking. Consequently, this simulated record masks the underlying plaintext, achieving coercion resistance. Furthermore, Kho et al. [30] extended the interactive voting architecture from the work of Finogina and Herranz [29]. In their scheme, the voter can cast a true encrypted vote v to the bulletin board and avoid coercion by duplicating and forging the verification parameters to mask the plaintext in the publishing phase.

To better highlight the fundamental differences between our proposed scheme and existing approaches, we provide a concise comparison in Table 1. We select two representative schemes from each of the three kinds of coercion-resistant electronic voting schemes, including fake credentials, deniable vote updating, and vote masking. We evaluate them across three critical dimensions: voting confirmation, revoting, and verifiability.

As illustrated in Table 1, prior methods designed for coercion resistance, including fake credentials, deniable vote updating, and vote masking, have the property of verifiability. In contrast, our somewhat deniable voting trades off between verifiability and coercion resistance. Specifically, somewhat deniable

voting gives up verifiability to some extent but not all in exchange for coercion resistance under the condition that the election result remains unchanged.

Table 1: Comparison with representative coercion-resistant schemes.

Scheme Category	Representative Schemes	Voting Confirmation	Revoting	Verifiability
Fake Credentials	Aziz [8]	Yes	No	Yes
	Clarkson et al. [9]	Yes	Yes	Yes
Deniable Vote Updating	Bernhard et al. [17]	Yes	Yes	Yes
	Heiberg et al. [19]	Yes	Yes	Yes
Vote Masking	Chen et al. [28]	Yes	No	Yes
	Kho et al. [30]	Yes	No	Yes
Our Scheme	Somewhat Deniable Voting	Yes	Yes	No

3 Somewhat Deniable Voting

As mentioned before, there are two key observations regarding existing coercion-resistant voting schemes. First, the coercion-resistant method of an electronic voting scheme is public information for everyone, including adversaries. Thus, adversaries can stop voters from using the coercion-resistant method. Second, adversaries can force voters to verify their votes to prevent voters from using the coercion-resistant method if adversaries know that voters have the ability to verify their votes.

Due to the two observations, voters must give up the ability to verify their votes and the ability to convince others to believe in whom they cast their votes if the voting scheme has coercion resistance. That is, non-verifiability and deniability are necessary. In particular, a voting scheme is a non-verifiable scheme if voters have no way to check whether their votes are counted for a particular candidate. Meanwhile, a voting scheme is a deniable scheme if voters can not convince others to believe in whom they cast their votes, even if voters would like to prove it.

However, the voting schemes can not obtain coercion resistance at the cost of losing verifiability. The verifiability of each voter is an important property of voting schemes so that voters can trust the voting scheme. To relieve the tension between verifiability and coercion resistance, we introduce a new concept named somewhat deniable voting.

The idea of somewhat deniable voting is borrowed from the law. There is a principle in law: freedom cannot be achieved at the cost of freedom. To be more precise, some people's complete freedom can not be at the cost of others completely losing their freedom. The prohibition on the sale of organs is a good example. Some people may argue for organ trading by saying that: my body is mine, and I should be allowed to completely control my body in law, including that I can sell my organs legally. Surely, organ trading is prohibited, and the reason for the prohibition is the principle mentioned before. In specific, some people will become richer, and some people will become poorer no matter what kind of economic rules are used. The poorer people may sell their organs for various reasons. For example, some mothers may sell their organs for money to treat their sick child, and some people may sell their organs just for money to buy something cool. Organ trading could cause these poorer people to lose control of their own bodies, while the initial intention of the argument is to let people control their bodies completely.

Based on the same logic, if one voting scheme grants everyone the freedom to completely prove who they voted for, then adversaries could utilize this freedom to coerce or buy the votes of poor voters. Finally,

while every voter seemingly attains the freedom to prove their votes, some voters completely lose the freedom to vote according to their true and independent wishes. Therefore, the verifiability of some voters who are not coerced should not be at the cost of the voting rights of voters who are coerced. Furthermore, coercion exists widely in our daily life, such as social pressure from friends and relatives, pressure from employers in the workplace, moral pressure from religious beliefs, etc. Consequently, the concept of somewhat deniable voting is introduced to address these challenges.

3.1 Threat Model

To formally analyze the coercion resistance of our somewhat deniable voting scheme, this section defines three key roles: the adversary, the voter, and the teller. Specifically, the adversary has the ability to coerce or bribe voters. The voter has the opportunity to cast their vote independently. Additionally, the teller is assumed to be trustworthy in our application scenario.

3.1.1 Adversary

It is assumed that the adversary attempts to break the coercion resistance. The adversary is assumed to have three capabilities: access to the confirmation message, access to parameters and outputs, and access to communication transcripts. First, the adversary can repeatedly demand that coerced voters request the confirmation message during the voting stage. Notably, the confirmation message received by voters is the string of "ack". Second, the adversary is assumed to have full knowledge of all scheme parameters, including the number of candidates m, the number of votes n, the deniability confidence k, and the number of injected dummy votes w. Furthermore, the adversary can observe all system outputs, including both the elected winner C and the final vote-candidate pairs $(vote_i, c_i)$. Finally, the adversary is assumed to have access to all encrypted network communication between voters and the teller. Although the adversary can eavesdrop on metadata including timestamps, IP addresses, operator identity, and the encrypted vote-candidate pairs $(vote_i, c_i)$, the adversary can not break the security of the encryption to recover the plaintext vote.

3.1.2 Voter

It is assumed that the voter has the opportunity to vote independently. Specifically, it is assumed that the physical safety of voters and the security of their voting devices are strictly guaranteed, and voters are not subjected to continuous on-site monitoring. This assumption ensures that voters have the opportunity to cast their votes without being under the supervision of the adversary. Notably, voters receive a confirmation message "ack" to verify that the teller has successfully received their votes during the active voting stage. During the publishing stage, voters could verify their votes. However, the candidate they truly supported may have been swapped by the telling algorithm with a certain probability.

3.1.3 Teller

In our scheme, the teller is assumed to be trustworthy. Firstly, the teller is an algorithm, not a person. That is, the teller actually references the telling algorithm, not just a person. Secondly, the telling algorithm could be implemented by Trusted Execution Environments such as SGX [31,32]. Therefore, the trust assumption is based on an algorithm that is implemented by Trusted Execution Environments. Thirdly, supported by strict code auditing and formal software verification in our application scenarios, the teller is assumed to be correctly implemented. Finally, in our application scenarios, the teller is assumed to be a strictly closed black box. The human operator or system administrator cannot manually modify the critical internal parameters or the execution workflow. Specifically, the parameters within the teller

are computed automatically. Therefore, the operator can neither observe the memory of the system nor manually modify the internal parameter values and execution logic of the program within the Trusted Execution Environments.

3.2 Definition of Our Scheme

Definition 1 ((m, n, k) Somewhat Deniable Voting Scheme): Assume that there are n voters and m candidates. A voting scheme M is a somewhat deniable voting scheme if the following equality holds under the condition that the elected candidate remains unchanged.

$$\begin{cases} P\{ver(M(c_i)) = c_i\} = \dfrac{n-k}{n} \\ P\{ver(M(c_i)) = c_j\} = \dfrac{k}{n(m-1)}, \quad \forall c_j \neq c_i \end{cases} \tag{1}$$

Here, the ver is the procedure to verify who a voter casts a vote for. The c_i represents a candidate. The k is an important parameter called deniability confidence.

Thus, for a somewhat deniable voting scheme, when a voter verifies whom he cast his vote for, the returned candidate is the candidate whom the voter did cast his vote for with probability $(n-k)/n$. The returned candidate is any other candidate whom the voter did not cast his vote for with probability $k/(n(m-1))$. In addition, the elected candidate needs to remain unchanged.

Any probability is between 0 and 1 so we have

$$0 < \frac{n-k}{n} < 1 \; and \; 0 < \frac{k}{n(m-1)} < 1 \tag{2}$$

namely,

$$0 < k < n \tag{3}$$

Note that $0 < k < n$ serves as the foundational boundary to maintain the mathematical validity of the probability distribution. As formally derived in Section 3.1, to satisfy the strict definition of Local Differential Privacy (i.e., privacy budget $\varepsilon > 0$), the parameter space is further refined.

Voters in a somewhat deniable voting scheme can reasonably deny who they actually cast votes for, no matter what the result of verification is. The reason is that the returned candidate is not the candidate for whom voters did cast votes with a certain probability. In other words, voters obtain deniability with a certain probability at the cost of giving up verifiability with a certain probability. If there is deniability in the voting stage, the somewhat deniable voting scheme has the coercion resistance as claimed in the next Theorem 1.

Theorem 1: *A somewhat deniable voting scheme with deniability in the voting phase has the property of coercion resistance.*

Proof: Firstly, deniability means that voters cannot prove to adversaries whom they cast their votes for in the voting phase, even if the voters would like to prove it. That is, voters can not show a third party, such as the adversary, for whom they cast their votes in the voting phase. In addition, somewhat deniable voting means that voters lose the ability to verify whom they cast their votes for.

Therefore, the adversary cannot be sure whether voters follow the order to cast a vote for the candidate supported by the adversary. The reason is that the voters themselves lose the ability to verify whom they cast

their votes for and the ability to prove to others whom they did cast their votes for with one hundred percent confidence. □

Notably, the above deniability is in the voting stage, and the non-verifiability is in the publishing stage. In some other contexts, the deniability is for all phases as a whole, which is different from our context. The coercion resistance of the entire scheme is achieved through the deniability in the voting stage and the non-verifiability in the publishing stage.

3.3 Analysis of Privacy Preservation

In this subsection, the proposed somewhat deniable voting scheme is analyzed from the perspective of local differential privacy, and the key parameter k is interpreted in detail.

Definition 2 (ε-Local Differential Privacy [33]): A randomized algorithm A satisfies ε-local differential privacy, if and only if for any pair of input values $v, v' \in D$, and any possible output $S \subseteq Range(M)$, the following inequality holds

$$\frac{Pr\{A(v) \in S\}}{Pr\{A(v') \in S\}} \leq e^{\varepsilon}. \tag{4}$$

Here, A is the local differential privacy mechanism, v, v' are data records of each participant, and Pr represents the probability that particular events happen.

The most local differential privacy mechanisms are based on randomized response technology, which was proposed by Warner as a survey technology to collect answers to embarrassing questions such as early pregnancy [34]. Kairouz et al. generalize the randomized response technology so that it can be applied in a larger answer domain, not just a binary answer domain [35].

Definition 3 (Generalized Randomized Response (GRR) [36]): For a user u with a value $v \in$ R, where R is a set with d possible values that a user can have. A random variable $\hat{t}$ represents the response of the user u with sample space R. The generalized randomized response works as follows:

$$Pr\{\hat{t} = v\} = \begin{cases} \dfrac{e^{\varepsilon}}{e^{\varepsilon} + d - 1}, if\ t = v \\ \dfrac{1}{e^{\varepsilon} + d - 1}, if\ t \neq v \end{cases} \tag{5}$$

The somewhat deniable voting scheme is a local differential privacy mechanism if it is checked from the perspective of local differential privacy as claimed in Theorem 2.

Theorem 2: *The (m, n, k) somewhat deniable voting scheme is a $\ln \frac{(n-k)(m-1)}{k}$-local differential privacy mechanism.*

Proof: Assume that a voter with identity number i casts his vote for a candidate c. By the definition of somewhat deniable voting, when the voter verifies whom he casts his vote for, the returned candidate $\hat{c}$ is c with probability k/n and the returned candidate is not c with probability $1 - k/n$. Assume that the returned candidate is a random variable denoted by t when the voter verifies whom he cast his vote for. Then, we have

$$Pr\{t = c\} = \frac{n - k}{n}, if\ \hat{c} = c \tag{6}$$

And, we know that there are m possible candidates, namely $m = d$, so we have

$$Pr\{t = c\} = \frac{k}{n(m-1)}, if\ \hat{c} \neq c \tag{7}$$

So, we have the following equation system

$$\begin{cases} \dfrac{n-k}{n} = \dfrac{e^{\varepsilon}}{e^{\varepsilon}+d-1} \\ \dfrac{k}{n(m-1)} = \dfrac{1}{e^{\varepsilon}+d-1} \end{cases} \tag{8}$$

By the equation system, we can obtain the ε as following

$$\begin{cases} d &= m \\ \varepsilon &= \ln \frac{(n-k)(m-1)}{k} \end{cases} \tag{9}$$

In a word, the (m, n, k) somewhat deniable voting scheme is a $\ln \frac{(n-k)(m-1)}{k}$-local differential privacy mechanism. □

The parameter k is the key parameter to determine the privacy guarantee of a somewhat deniable voting scheme. Let

$$\varepsilon = \ln \frac{(n-k)(m-1)}{k}. \tag{10}$$

By derivation, we know

$$\varepsilon' = \frac{-n}{k(n-k)}. \tag{11}$$

We know $0 < k < n$, so we have

$$\varepsilon' = \frac{-n}{k(n-k)} < 0. \tag{12}$$

So, ε is a monotonically decreasing function of k in the interval $(0, n)$.

Furthermore, since the privacy budget ε must be strictly positive (i.e., $\varepsilon > 0$), the argument inside the logarithm must satisfy:

$$\frac{(n-k)(m-1)}{k} > 1. \tag{13}$$

By rearranging and simplifying terms, we obtain the strict upper bound for the deniability confidence parameter k:

$$k < \frac{n(m-1)}{m}. \tag{14}$$

As k increases from 0 to the upper bound $n(m-1)/m$, the privacy guarantee of somewhat deniable voting scheme becomes stronger. Specifically, when k goes to 0, $\varepsilon = \ln \frac{(n-k)(m-1)}{k}$ goes to positive infinity, namely

$$\lim_{k \to 0} \varepsilon = \lim_{k \to 0} \ln \frac{(n-k)(m-1)}{k} = +\infty. \tag{15}$$

That is, the somewhat deniable voting becomes a real-name voting scheme, resulting in that any voter has no ability to deny his vote. On the contrary, when k goes to the upper bound $n(m-1)/m$, $\varepsilon = \ln \frac{(n-k)(m-1)}{k}$ goes to 0, namely

$$\lim_{k \to \frac{n(m-1)}{m}} \varepsilon = \lim_{k \to \frac{n(m-1)}{m}} \ln \frac{(n-k)(m-1)}{k} = 0. \tag{16}$$

That is, the somewhat deniable voting gradually becomes an anonymous voting scheme, resulting in each voter being able to deny his vote with one hundred percent confidence. So, we name k the deniability confidence.

In a word, the somewhat deniable voting is one kind of voting concept lying between real-name voting and anonymous voting, and the somewhat deniable voting changes from a real-name voting scheme to an anonymous voting scheme as k increases from 0 to the upper bound $n(m-1)/m$. That is, the ability to deny becomes stronger as deniability confidence k increases from 0 to the upper bound $n(m-1)/m$.

Theorem 3: *In the (m, n, k) somewhat deniable voting scheme, the adversary's verification ability regarding voter compliance is mathematically formulated as*

$$\frac{1}{1 + \frac{1-p}{(\frac{n}{k}-1)(m-1)p}}, \tag{17}$$

where p represents the probability that the voter complies with the adversary by casting a vote for the specified candidate.

Proof: The adversary's ability to verify voter compliance is formally defined as the posterior probability that the voter casts a vote for a particular candidate T according to the adversary's orders, given the published vote results from the teller. Let E denote the event that the voter cast a vote for candidate T according to the published votes. Let H_1 be the event that the voter follows the adversary's order (with a prior probability of $\Pr[H_1] = p$), and H_2 be the event that the voter does not follow the adversary's order and chooses an alternative candidate other than T (with a probability of $\Pr[H_2] = 1 - p$).

According to Bayes' theorem, the verification ability of the adversary can be formulated as follows:

$$\Pr[H_1 \mid E] = \frac{\Pr[E \mid H_1] \cdot \Pr[H_1]}{\Pr[E]}. \tag{18}$$

By applying the law of total probability, the marginal probability $\Pr[E]$ can be expanded as:

$$\Pr[E] = \Pr[E \mid H_1] \Pr[H_1] + \Pr[E \mid H_2] \Pr[H_2]. \tag{19}$$

Substitution yields:

$$\Pr[H_1 \mid E] = \frac{\Pr[E \mid H_1] \cdot \Pr[H_1]}{\Pr[E \mid H_1] \Pr[H_1] + \Pr[E \mid H_2] \Pr[H_2]}. \tag{20}$$

Based on our previous derivation of the conditional probabilities, we have $\Pr[E \mid H_1] = 1 - \frac{k}{n}$ and $\Pr[E \mid H_2] = \frac{k}{n} \cdot \frac{1}{m-1}$. By substituting these probabilities into the equation, we obtain:

$$\Pr[H_1 \mid E] = \frac{\left(1 - \frac{k}{n}\right) p}{\left(1 - \frac{k}{n}\right) p + \frac{k}{n} \cdot \frac{1}{m-1} \cdot (1-p)}. \tag{21}$$

Through algebraic simplification, the adversary's verification ability is finally established as:

$$\Pr[H_1 \mid E] = \frac{1}{1 + \frac{1-p}{\left(\frac{n}{k}-1\right)(m-1)p}}. \tag{22}$$

□

In our scheme, the expression of $\Pr[H_1 \mid E]$ demonstrates that the adversary's verification ability of voter compliance is inversely proportional to the parameter k. As k approaches 0, the privacy budget $\varepsilon \to +\infty$. That is, no votes are shuttled during the telling stage, so who the voter casts his vote for is completely exposed. The adversary can directly infer from the published results whether the voter cast his vote for a particular candidate. In this scenario, the adversary's verification ability regarding voter compliance reaches 100%. Therefore, based on our mathematical formulation:

$$\lim_{k\to 0} \Pr[H_1 \mid E] = \lim_{k\to 0} \frac{1}{1 + \frac{1-p}{\left(\frac{n}{k}-1\right)(m-1)p}} = 1. \tag{23}$$

Conversely, as k approaches the theoretical upper bound $\frac{n(m-1)}{m}$, $\varepsilon \to 0$. The strongest privacy preservation for the voters is achieved and the adversary's verification ability is minimal. That is:

$$\lim_{k\to \frac{n(m-1)}{m}} \Pr[H_1 \mid E] = \lim_{k\to \frac{n(m-1)}{m}} \frac{1}{1 + \frac{1-p}{\left(\frac{n}{k}-1\right)(m-1)p}} = p. \tag{24}$$

In this case, $\Pr[H_1 \mid E] = \Pr[H_1]$, meaning the posterior probability is exactly equal to the prior probability. The adversary can not extract any effective information from the published vote results to improve their verification ability, thereby achieving "Zero Information Gain" [37]. Consequently, the adversary's verification ability regarding voter compliance becomes weaker as deniability confidence k increases from 0 to the upper bound $n(m-1)/m$.

4 Implement of Somewhat Deniable Scheme

This section presents an implementation of the proposed somewhat deniable voting scheme. There are three phases in the implemented scheme. Specifically, the voting system is initialized in the first phase, including choosing all parameters for the system and distributing keys. Voters cast their votes in the second phase. The teller counts votes and publishes final results in the third phase. The system structure is shown in Fig. 2.

Figure 2: System structure.

4.1 Setup

The deniable authenticated encryption scheme (DAES) is chosen [38] as the encryption scheme used for communication between voters and the teller. The sub-algorithm keygen of DAES is invoked to generate private keys together with public keys for voters and the teller. Voters send their public keys to the teller, and the teller sends its public key to all voters. Suppose that n voters and m candidates participate in the election. These candidates are randomly assigned a unique identity number from 0 to $m-1$. Choose a value less than $n(m-1)/m$ for deniability confidence k according to the requirements of application scenarios.

The extent to which voters can deny their votes is controlled by the deniability confidence k. Deniability confidence is a flexible parameter and is dependent on the application scenarios, so the chosen value of deniability confidence is determined by the needs of the application scenarios. Notably, deniability confidence k must be less than the upper bound $n(m-1)/m$ as mentioned in the last section.

The DAES is chosen because it provides deniability. Specifically, deniability ensures that the communication participants cannot convince a third party to believe the actual communication content if the third party did not participate in the communication [38]. That is, the participants of communication via DAES have no ability to persuasively show others the communication content. Therefore, for our implemented scheme, voters cannot prove to others which candidate they actually cast their vote for if these votes are encrypted by DAES.

Within the scope of the assumed threat model, deniability strengthens coercion resistance in the voting scheme. Specifically, voters cannot prove to adversaries which candidate voters actually voted for. Even if voters are coerced or bribed to reveal their votes, the deniability property prevents them from providing verifiable evidence of their true votes. Consequently, adversaries cannot determine whether voters complied with instructions to vote for a particular candidate. Since voter compliance can not be verified, coercion and bribery become ineffective for influencing election outcomes. Under the defined threat model, adversaries therefore have little incentive to threaten or bribe voters, because they cannot effectively assess whether their demands have been followed. Notably, this coercion resistance is achieved by limiting the adversary's ability to verify voter compliance after the voting stage. This argument assumes the absence of a stronger attack that directly reveals the voter's vote. Therefore, credential transfer, delegated voting, device seizure, and key disclosure are excluded from the threat model.

There is a task in the setup step, namely distributing keys for voters and the teller. This is a necessary task that needs to be done even for a non-electronic voting scheme. For example, in the national election, the government officers need to identify who has the right to vote and distribute the legal vote to legal citizens. The process of distributing keys is just like the process of distributing ballots. The process of distributing keys can be done by the Diffie-Hellman key exchange protocol [39]. If the proposed scheme is for application scenarios where there are only dozens or hundreds of persons, the task of distributing keys can also be done by distributing a vote support device with embedded keys to each voter.

There are still three things to explain. First, the identity number of candidates starts from 0, not 1. The reason is that the modular operation is used to determine the identity number of candidates whose votes need to be processed in the third phase. So, it is convenient in terms of math if the identity number starts from 0. Second, k is less than n. In our design, k voters' votes could be processed in the third phase, and the number of votes that could be processed needs to be less than the number of voters. So, k must be less than n. As mentioned before, how to determine the value of k depends on application scenarios. Third, the setup step is executed by the teller who is supposed to be trustworthy.

4.2 Casting

Each voter i chooses his favorite candidate with the identity number c_i and randomly chooses k integer numbers denoted by $R_{i,0}, R_{i,1}, \ldots, R_{i,k-1}$. The voter i constructs a message.

$$M_i = c_i||R_{i,0}||\ldots||R_{i,k-1} \tag{25}$$

The message M_i is encrypted into $vote_i$ as the vote of voter i by DAES. The $vote_i$ is sent to the teller. When the teller receives the $vote_i$, the teller sends the "ack" in encrypted form as the return if the $vote_i$ is legal. The voter can resend another vote if he changes his decision. The new vote overwrites the old vote.

These k random integer numbers are used to determine the identity numbers of voters whose c_i is operated in the third phase. For example, $R_{i,j}$ partly determines the identity number of voters whose c_i is operated in the third phase. The identity number is calculated from the random numbers from all voters, resulting in two advantages. Firstly, the calculation is in a distributed manner so that no one can control the generated identity numbers. Thus, the distributed manner of generating the identity number makes the generated identity number more trustworthy. Secondly, although some voters may obtain low-quality random numbers, the generated identity number is a high-quality random number even if there is only one voter who can generate high-quality random numbers.

It is acceptable as long as the number $R_{i,j}$ is a random integer. Specifically, the random number $R_{i,j}$ can be drawn from any distribution because a randomly drawn number for any voter only has a partial influence on the final generated identity number. It is acceptable that the random number $R_{i,j}$ comes from any interval because the calculation result is mapped to the right range by the modular operation, no matter how big or small the calculation result is.

The DAES has the authentication function. Therefore, the teller can ensure that votes come from legal voters, and voters can make sure that the confirmation message "ack" comes from the teller. That is, the DAES ensures that the votes are indeed cast by the legal voters themselves under the defined threat model.

Notably, voters could cast their votes many times before the deadline for casting. The option to resend votes makes the proposed schemes more convenient and flexible. For example, the chance to resend a vote makes it possible to update voters' choices if voters get some new information. In addition, voters can correct their vote if there are some mistakes, such as typos or errors caused by software bugs.

4.3 Telling

The teller decrypts all received votes and performs calculations according to Algorithm 1. The teller publishes all voters' votes in pairs $(vote_i, c_i)$.

In the telling algorithm, the first task is to select k voters whose votes are changed later. Thus, each voter's vote is changed with probability k/n. When voters verify which candidate they cast their votes for, the returned candidate is the candidate for whom the voter did cast the vote with probability $1 - k/n = (n - k)/n$. In particular, steps from 5 to 11 are to choose the k voters, and steps from 7 to 9 make sure that no voter is chosen twice.

In the telling algorithm, the second task is to change the selected k voters' votes under the condition that the elected candidate remains unchanged. To that end, the teller constructs w/m votes for each candidate, resulting in w new votes. The constructed w votes and selected k votes are shuttled such that the supported candidate of each vote is changed to another one.

Algorithm 1: Telling

Require:

$vote_1, vote_2, \ldots, vote_n$;

Ensure:

pair $(vote_1, c_1), (vote_2, c_2), \ldots, (vote_n, c_n)$;

election winner C;

for i=0 to n-1 **do**
  decrypt $vote_i$ as $M_i = c_i||R_{i,0}||\ldots||R_{i,k-1}$;
end for
let $I = \varnothing$; // Initialize the set for transformed candidate values
let $S = \varnothing$; // Initialize the set for selected voter index
for j=0 to k-1 **do**
  $index = (\sum_{i=0}^{n} R_{i,j})\%n$;
  while $index \in S$ **do**
    $index = index + 1$;
  end while
  put $index$ in S;
  put c_{index} in I;
end for
let w is the smallest number such that $w \geq \frac{km}{(m-2)}$ and $w\%m = 0$;
for t=0 to w-1 **do**
  $c_{n+t} = t\%m$;
  put c_{n+t} in I;
end for
swap the value of elements in set I such that all elements' value are changed.
select the value with maximum frequency from $c_0, c_1, \ldots, c_{n+w-1}$ as C;
for i=0 to n-1 **do**
  $M_i = c_i||R_{i,0}||\ldots||R_{i,k-1}$;
  M_i is encrypted into $vote_i$ again by the teller;
end for
for i=n to n+w-1 **do**
  select k random integer number $R_{i,0}, R_{i,1}, \ldots, R_{i,k-1}$,
  $M_i = c_i||R_{i,0}||\ldots||R_{i,k-1}$;
  M_i is encrypted into $vote_i$ by the teller;
end for
return C and $(vote_1, c_1)$,$(vote_2, c_2), \ldots, (vote_n, c_n), \ldots$,$(vote_{n+w-1}, c_{n+w-1})$;

The elected candidate remains unchanged, although the number of votes is larger than the number of voters. Specifically, the teller casts an extra w/m votes for each candidate. Thus, the number of votes for each candidate increases by w/m. The election result where each candidate obtains extra w/m votes is the same as the election result where each candidate does not have the extra w/m votes.

There are two requirements for w, the number of votes added by the teller. Firstly, the number of candidates m is a factor of w such that the added votes for each candidate are the same, namely $w\%m = 0$. Secondly, it is achievable to shuttle values of selected k votes and w votes added such that each vote's supported candidate is swapped to another candidate. In particular, the worst case is the case where the selected k votes are votes for the same candidate. So, the next inequality needs to hold

$$\frac{w}{m}(m-1) \geq k + \frac{w}{m} \tag{26}$$

Rearranging this inequality, we have

$$w \geq \frac{km}{m-2} \tag{27}$$

This inequality (27) establishes the necessary condition under which all selected real votes are swapped during the shuttle phase. Although each candidate's vote count increases by exactly w/m, the relative size of their vote count remains unchanged. Thus, the elected winner remains unchanged. Notably, the published vote distribution in our scheme is a transformed distribution rather than the original vote distribution.

In certain scenarios, two or more candidates may obtain the highest number of votes at the same time. How to deal with such cases depends on the rules of the election. For example, during the Electoral College vote in the 2020 US presidential election, presidential candidates Donald Trump and Joe Biden may theoretically obtain the same number of votes. If such a tie occurs in the Electoral College, the President is elected by the House of Representatives according to the US Constitution (12th Amendment). Consequently, election rules play a key role in such cases, not technology.

5 Evaluation

In this section, detailed information on extensive experiments is given. Our goal is to evaluate the implemented scheme in terms of privacy guarantee, time consumption, and coercion resistance. Three key factors are considered in the experiments, including the number of candidates, the number of voters, and deniability confidence. All experiments are conducted on a PC equipped with an AMD Ryzen 9 8945HX CPU, 16 GB RAM, a 1 TB NVMe hard disk, and a 64-bit Windows 11 operating system. The experiments are implemented in Python 3.10.18.

5.1 Evaluation on Privacy Guarantee

In the first experiment, the privacy guarantee of the implemented scheme is demonstrated. The graph of function $\varepsilon = \ln \frac{(n-k)(m-1)}{k}$ is shown in Fig. 3. As can be seen, when deniability confidence goes to 0, the privacy budget becomes bigger, resulting in the privacy guarantee becoming weaker and the implemented scheme gradually becoming closer to a real-name voting scheme. On the contrary, when deniability confidence goes to the number of voters, the privacy budget becomes smaller, resulting in the privacy guarantee becoming stronger and the implemented scheme gradually becoming closer to an anonymous voting scheme. In terms of the number of candidates, the privacy budget increases as the number of candidates increases, which means that the privacy guarantee becomes weaker as the number of candidates increases.

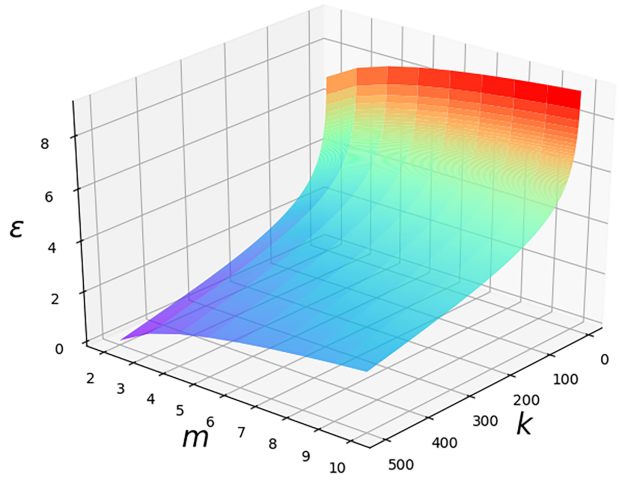

Figure 3: Privacy guarantee. The number of voters n is 1000, m is the number of candidates, k is deniability confidence, and ε is privacy budget which quantifies the privacy guarantee of differential privacy mechanisms.

5.2 Evaluation on Time Consumption

Next, the focus is on time consumption. The theoretical analysis of time consumption is shown in Table 2. In particular, n is the number of voters. m is the number of candidates. k is the deniability confidence. w denotes the number of dummy votes, defined as the smallest integer satisfying $w \geq \frac{km}{m-2}$. T_h represents the time to execute a hash function. T_e represents the time to execute a modular exponentiation operation. T_m represents the time to execute a modular multiplication operation. T_i represents the time to execute a modular inverse operation.

Table 2: Theoretical analysis of time consumption.

	Setup	Casting	Telling
Time	$(n+1+w)T_e$	$(2T_h + 2T_e + T_m)n$	$(4T_h + 3T_e + 3T_m + T_i)n$ $+ (2T_h + 2T_e + 2T_m + T_i)w$

In the second experiment, the influence of deniability confidence on time consumption is evaluated. In particular, the number of voters n is 1000, and the number of candidates m is 10. Deniability confidence is from 100 to 500. The experimental results are shown in Fig. 4.

According to Fig. 4, the theoretical analysis is consistent with experimental results. In particular, the setup time consumption increases slowly as deniability confidence increases. The reason is that the setup algorithm needs to generate more pairs of keys as deniability confidence increases. The casting time consumption remains unchanged because the casting time consumption is only related to the number of voters. The telling time consumption increases as deniability confidence increases. The increase in deniability confidence leads to more dummy votes that need to be generated by the teller, resulting in an increase in the time consumption of the telling algorithm.

In the third experiment, the number of candidates' influence on time consumption is evaluated. In particular, the number of voters n is 1000 and the deniability confidence k is 10. The number of candidates is from 100 to 500. The experimental results are shown in Fig. 5.

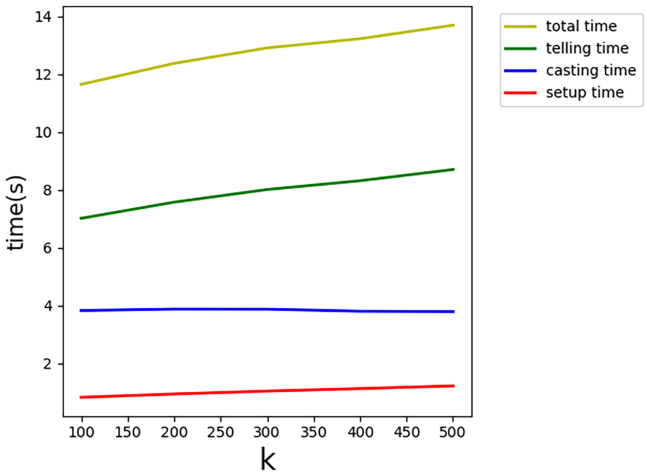

Figure 4: Deniability confidence's influence on time consumption.

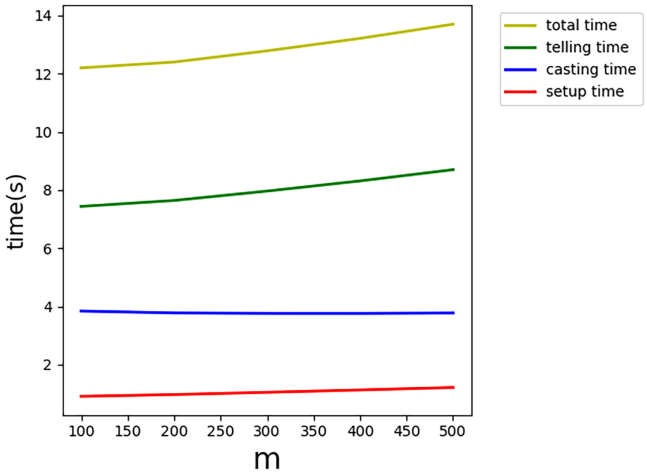

Figure 5: Number of candidates' influence on time consumption.

According to Fig. 5, the theoretical analysis is consistent with experimental results. In particular, the time consumption of setup and telling increases as the number of candidates increases. The number of candidates has no influence on the time consumption of casting.

In the fourth experiment, the number of voters' influence on time consumption is evaluated. In particular, the number of candidates m is 10 and the deniability confidence k is 10. The number of voters is from 100 to 1000. The experimental results are shown in Fig. 6.

According to Fig. 6, the theoretical analysis is consistent with experimental results. In particular, the time consumption of setup, casting, and telling increases as the number of voters increases.

5.3 Evaluation on Coercion Resistance

In the final experiment, the coercion resistance of somewhat deniable voting scheme is evaluated by analyzing the adversary's verification ability regarding voter compliance. As discussed in Theorem 3, this ability is related to deniability confidence k. Furthermore, since privacy budget ε can be mathematically derived from k, given the number of voters n and candidates m, it demonstrates that the adversary's verification ability fundamentally depends on the system's privacy budget ε.

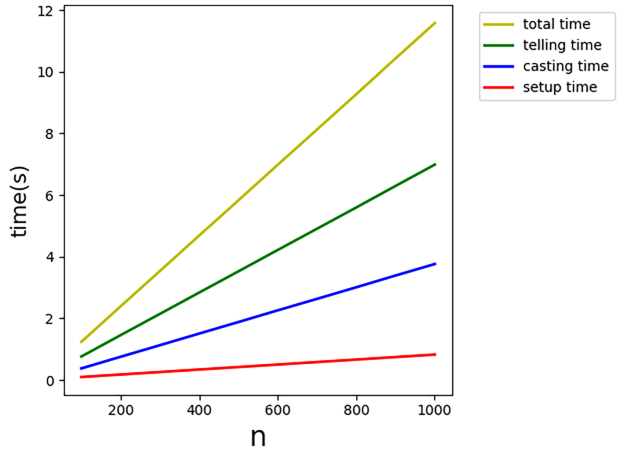

Figure 6: Number of voters' influence on time consumption.

Fig. 7 demonstrates the experimental results regarding the relationship between the scheme's coercion resistance and the deniability confidence k. In this experiment, the number of voters n and candidates m are set to 1000 and 10, respectively. Coercion resistance is quantified by the adversary's verification ability regarding voter compliance, represented by the y-axis. The deniability confidence k is represented by the x-axis. The evaluation is conducted across varying prior compliance probabilities, specifically p = 40%, 50%, and 60%. These probabilities represent the likelihood that a voter complies with the adversary's demands. When k = 100, the verification ability across all three curves is close to 100%, demonstrating that the scheme degrades toward a real-name voting system when fewer votes are shuttled. Conversely, as k gradually increases, all three curves drop quickly, demonstrating that the adversary gradually loses verification ability and that the system's coercion resistance gradually increases.

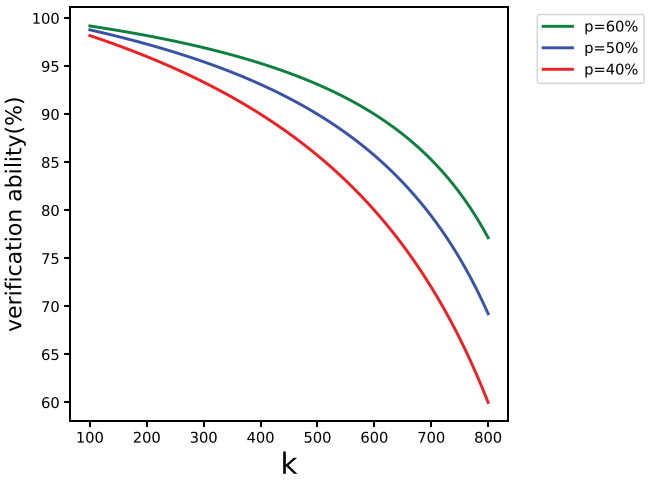

Figure 7: Deniability confidence's influence on verification ability.

Fig. 8 demonstrates the relationship between coercion resistance and the privacy budget ε. In this experiment, the number of voters n and candidates m are set to 1000 and 10, respectively. The adversary's verification ability is represented by the y-axis, where the value of 100% demonstrates a complete loss of coercion resistance. The privacy parameter ε is represented by the x-axis. The experiment evaluates the scheme under different prior compliance probabilities, p = 40%, 50%, and 60%. When a loose privacy

budget is applied, e.g., $\varepsilon = 5.0$, the adversary's verification ability is close to 100%, nearly a real-name voting environment. However, as the privacy budget decreases, the verification ability drops quickly. At a strict privacy budget of $\varepsilon = 0.1$, the verification ability is close to its empirical minimum, where each curve gradually converges to its respective prior compliance probability p. This demonstrates that the published vote results have nearly zero information gain for the adversary, who can not infer voter compliance from the voting outputs. In this case, the scheme is close to an anonymous voting scheme.

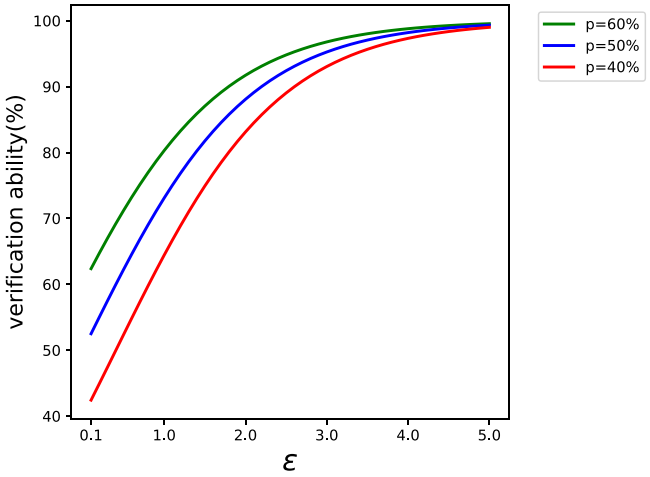

Figure 8: Privacy budget's influence on verification ability.

Based on the above discussion, the experimental results demonstrate that our somewhat deniable voting scheme, which satisfies $\ln \frac{(n-k)(m-1)}{k}$-local differential privacy mechanism, is a new kind of voting scheme lying between a real-name voting scheme and an anonymous voting scheme. By appropriately configuring the parameter k, our somewhat deniable voting scheme provides a degree of coercion resistance under the defined threat model.

6 Limitations

Although our proposed somewhat deniable voting scheme effectively prevents voters from being coerced within the defined threat model, certain limitations remain. Specifically, the scheme cannot directly protect votes from the risks of physical coercion. If an adversary physically seizes the private key, voting device, authentication token, or even conducts on-site monitoring, the coercion resistance may fail. Additionally, the overall security of our scheme relies on the correct implementation of the teller. If the teller is malicious, compromised, or incorrectly implemented, it introduces a critical risk of privacy leakage. Furthermore, the overall security of our scheme fundamentally depends on the reliability of the Trusted Execution Environments. If the Trusted Execution Environment is broken, the decrypted plaintext votes would be directly exposed to the adversary, and the claimed privacy and coercion-resistance guarantees may no longer hold.

Moreover, it is crucial to clarify the relationship between local differential privacy and voting security. Our local differential privacy analysis provides a mathematical explanation for deniability confidence and the adversary's verification ability regarding voter compliance, which is useful for providing voting security. However, voting security relies not only on the mathematical local differential privacy guarantees but also strictly depends on the full voting protocol, authentication records, transcripts, timing information, and device security. These aspects remain to be addressed in future work.

Consequently, our electronic voting scheme can be applied to many scenarios, such as the board elections of large companies and national elections. As long as the physical safety of voters and the Trusted Execution Environment can be guaranteed, our scheme can be adopted. For example, in the elections of large companies or national elections, voters are usually not subjected to physical coercion from adversaries directly. Although voters might still face pressure from employers, friends, or relatives, the legal framework ensures that they are not under continuous, on-site monitoring. Consequently, voters can easily find opportunities to cast their votes independently, such as inside their personal vehicles after work or within a private room at home. Additionally, these scenarios can provide code auditing, formal software verification, and the deployment of physically secured Trusted Execution Environment hardware, ensuring that the theoretical trust assumptions are well maintained in practice.

7 Conclusion

In this paper, we introduce a new concept of somewhat deniable voting. A somewhat deniable voting scheme has coercion resistance and privacy preservation properties. Specifically, somewhat deniable voting gives up verifiability to some extent but not all in exchange for coercion resistance under the condition that the election result remains unchanged. A somewhat deniable voting scheme is a local differential privacy mechanism, which means that the somewhat deniable voting scheme is a new kind of voting scheme lying between a real-name voting scheme and an anonymous voting scheme. An implemented scheme of somewhat deniable voting is given, and extensive experiments are conducted to evaluate performance.

Acknowledgement: None.

Funding Statement: This work was supported in part by the Advanced Materials-National Science and Technology Major Project under Grant 2024ZD0608200, in part by the National Natural Science Foundation of China under Grant 82474394, in part by Sichuan Science and Technology Program under Grant 2026NSFSC1462, in part by the MOE (Ministry of Education in China) Liberal arts and Social Sciences Foundation under Grant 24XJCZH004, and in part by the Sichuan University Postdoctoral Interdisciplinary Innovation Fund.

Author Contributions: The authors confirm their contributions to the paper as follows: Conceptualization: Mingxuan Jia, Wen Huang, Jian Peng; software: Wen Huang; supervision: Wen Huang, Jian Peng; validation: Mingxuan Jia, Chenglong Shi; writing—original draft: Mingxuan Jia, Yang Ye; writing—review & editing: Mingxuan Jia, Yang Ye, Wen Huang. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the Corresponding Author, Jian Peng, upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Wang B, Guo F, Liu Y, Li B, Yuan Y. An efficient and versatile e-voting scheme on blockchain. Cybersecurity. 2024;7(1):62. doi:10.1186/s42400-024-00226-8.
2. Shubhranshu K, Verma HK. Privacy-preserving and coercion-resistant e-voting via unlinkable biometric credentials on a hybrid permissioned blockchain. In: Proceedings of the IEEE 4th International Conference on Device Intelligence, Computing and Communication Technologies (DICCT); 2026 Apr 24–25; Dehradun, India. p. 267–72. doi:10.1109/dicct69099.2026.11535975.

3. Swearingen N, Zou X, Li N. Fully transparent, privacy-preserving yet verifiable, attack-resistant, and practical remote electronic voting rendering assured and fair elections. IEEE Trans Priv. 2025;2:105–18. doi:10.1109/tp.2025.3603141.
4. Merino LH, Colombo S, Reyes R, Azhir A, Mishra S, Tennage P, et al. TRIP: coercion-resistant registration for e-voting with verifiability and usability in Votegral. In: Proceedings of the ACM SIGOPS 31st Symposium on Operating Systems Principles; 2025 Oct 13–16; Seoul, Republic of Korea. p. 834–74. doi:10.1145/3731569.3764837.
5. Kulyk O, Neumann S. Human factors in coercion resistant internet voting—a review of existing solutions and open challenges. In: Proceedings of the Fifth International Joint Conference on Electronic Voting; 2020 Oct 6–9; Bregenz, Austria. p. 189–204.
6. Cortier V, Galindo D, Kusters R, Mueller J, Truderung T. Sok: verifiability notions for e-voting protocols. In: Proceedings of the 2016 IEEE Symposium on Security and Privacy (SP); 2016 May 23–25; San Jose, CA, USA. p. 779–98. doi:10.1109/SP.2016.52.
7. Alown M, Kiraz MS, Bingol MA. Enhancing democratic processes: a survey of DRE, internet, and blockchain in electronic voting systems. IEEE Access. 2025;13(2):20512–45. doi:10.1109/access.2025.3531349.
8. Aziz A. Coercion-resistant e-voting scheme with blind signatures. In: Proceedings of the 2019 Cybersecurity and Cyberforensics Conference (CCC); 2019 May 8–9; Melbourne, Australia. p. 143–51. doi:10.1109/CCC.2019.00009.
9. Clarkson MR, Chong S, Myers AC. Civitas: toward a secure voting system. In: Proceedings of the 2008 IEEE Symposium on Security and Privacy; 2008 May 18–21; Oakland, CA, USA. p. 354–68. doi:10.1109/SP.2008.32.
10. Iovino V, Rial A, Ronne PB, Ryan PY. Using Selene to verify your vote in JCJ. Lect Notes Comput Sci. 2017;10323(4):385–403. doi:10.1007/978-3-319-70278-0_24.
11. Neji W, Blibech K, Rajeb NB. Incoercible fully-remote electronic voting protocol. Lect Notes Comput Sci. 2017;10299(1):355–69. doi:10.1007/978-3-319-59647-1_26.
12. Zaghloul E, Li T, Ren J. Anonymous and coercion-resistant distributed electronic voting. In: Proceedings of the 2020 International Conference on Computing, Networking and Communications (ICNC); 2020 Feb 17–20; Big Island, HI, USA. p. 389–93. doi:10.1109/icnc47757.2020.9049653.
13. Merino LH, Azhir A, Zhang H, Colombo S, Tellenbach B, Estrada-Galiñanes V, et al. E-vote your conscience: perceptions of coercion and vote buying, and the usability of fake credentials in online voting. In: Proceedings of the 2024 IEEE Symposium on Security and Privacy (SP); 2024 May 20–24; San Francisco, CA, USA. p. 3478–96. doi:10.1109/sp54263.2024.00252.
14. Yin Z, Zhang B, Nastenko A, Oliynykov R, Ren K. A Scalable coercion-resistant voting scheme for blockchain decision-making. IEEE Trans Dependable Secur Comput. 2026;23(3):5090–105. doi:10.1109/tdsc.2026.3651473.
15. Giustolisi R, Garjan MS. Efficient cleansing in coercion-resistant voting. Lect Notes Comput Sci. 2024;15014:72–88. doi:10.1007/978-3-031-72244-8_5.
16. Sarier ND. Efficient, usable and coercion-resistant blockchain-based e-voting. J Inf Secur Appl. 2025;92(8):104074. doi:10.1016/j.jisa.2025.104074.
17. Bernhard D, Kulyk O, Volkamer M. Security proofs for particPipation privacy, receipt-freeness and ballot privacy for the helios voting scheme. In: Proceedings of the 12th International Conference on Availability, Reliability and Security; 2017 Aug 29–Sep 1; Reggio Calabria, Italy. p. 1–10. doi:10.1145/3098954.3098990.
18. Kulyk O, Teague V, Volkamer M. Extending helios towards private eligibility verifiability. Lect Notes Comput Sci. 2015;9269(1):57–73. doi:10.1007/978-3-319-22270-7_4.
19. Heiberg S, Martens T, Vinkel P, Willemson J. Improving the verifiability of the Estonian internet voting scheme. Lect Notes Comput Sci. 2016;10141(1):92–107. doi:10.1007/978-3-319-52240-1_6.
20. Locher P, Haenni R, Koenig RE. Coercion-resistant internet voting with everlasting privacy. Lect Notes Comput Sci. 2016;9604:161–75. doi:10.1007/978-3-662-53357-4_11.
21. Giustolisi R, Garjan MS, Schuermann C. Thwarting last-minute voter coercion. In: Proceedings of the 2024 IEEE Symposium on Security and Privacy (SP); 2024 May 19–24; San Francisco, CA, USA. p. 3423–39. doi:10.1109/sp54263.2024.00112.
22. Müller J, Pejó B, Pryvalov I. Devos: deniable yet verifiable vote updating. Proc Priv Enhancing Technol. 2024;1(1):357–78. doi:10.56553/popets-2024-0021.

23. Farzaliyev V, Willemson J. End-to-End verifiable internet voting with partially private bulletin boards. Lect Notes Comput Sci. 2025;16028(5):73–89. doi:10.1007/978-3-032-05036-6_5.
24. Ben-Sasson E, Bentov I, Horesh Y, Riabzev M. Scalable zero knowledge with no trusted setup. Lect Notes Comput Sci. 2019;11694(3):701–32. doi:10.1007/978-3-030-26954-8_23.
25. Barelli R, D'Onghia M, Longari S. Towards secure electronic voting: a survey on e-voting systems and attacks. IEEE Access. 2025;13(1):89600–26. doi:10.1109/access.2025.3569334.
26. Sako K, Kilian J. Receipt-free mix-type voting scheme. Lect Notes Comput Sci. 1995;921:393–403. doi:10.1007/3-540-49264-X_32.
27. Xia Z, Tong Z, Xiao M, Chang CC. Framework for practical and receipt-free remote voting. IET Inf Secur. 2018;12(4):326–31. doi:10.1049/iet-ifs.2017.0213.
28. Chen TL, Liu CH, Ou YH, Huang YM, Wu ZY. An improved and efficient coercion-resistant measure for electronic voting system. Int J Inf Secur. 2024;23(4):2637–54. doi:10.1007/s10207-024-00852-w.
29. Finogina T, Herranz J. On remote electronic voting with both coercion resistance and cast-as-intended verifiability. J Inf Secur Appl. 2023;76(6):103554. doi:10.1016/j.jisa.2023.103554.
30. Kho YX, Heng SH, Tan SY, Chin JJ. A provably secure coercion-resistant e-voting scheme with confidentiality, anonymity, unforgeability, and CAI verifiability. PLoS One. 2025;20(6):e0324182. doi:10.1371/journal.pone.0324182.
31. Costan V, Devadas S. Intel SGX explained. [cited 2026 Jan 1]. Available from: https://eprint.iacr.org/2016/086.
32. Zheng W, Wu Y, Wu X, Feng C, Sui Y, Luo X, et al. A survey of Intel SGX and its applications. Front Comput Sci. 2021;15(3):153808. doi:10.1007/s11704-019-9096-y.
33. Duchi JC, Jordan MI, Wainwright MJ. Local privacy and statistical minimax rates. In: Proceedings of the 2013 IEEE 54th Annual Symposium on Foundations of Computer Science; 2013 Oct 26–29; Berkeley, CA, USA. p. 429–38. doi:10.1109/FOCS.2013.53.
34. Warner SL. Randomized response: a survey technique for eliminating evasive answer bias. J Am Stat Assoc. 1965;60(309):63–9. doi:10.1080/01621459.1965.10480775.
35. Kairouz P, Bonawitz K, Ramage D. Discrete distribution estimation under local privacy. arXiv:1602.07387. 2016. doi:10.48550/arxiv.1602.07387.
36. Yang M, Lyu L, Zhao J, Zhu T, Lam KY. Local differential privacy and its applications: a comprehensive survey. arXiv:2008.03686. 2020. doi:10.48550/arxiv.2008.03686.
37. Goldwasser S, Micali S. Probabilistic encryption. J Comput Syst Sci. 1984;28(2):270–99. doi:10.1016/0022-0000(84)90070-9.
38. Huang W, Liao Y, Zhou S, Chen H. An efficient deniable authenticated encryption scheme for privacy protection. IEEE Access. 2019;7:43453–61. doi:10.1109/access.2019.2907250.
39. Diffie W, Hellman M. New directions in cryptography. IEEE Trans Inf Theory. 1976;22(6):644–54. doi:10.1109/TIT.1976.1055638.