AUT Journal of Mathematics and Computing

AUT J. Math. Comput., 6(2) (2025) 177-191
https://doi.org/10.22060/AJMC.2024.22420.1157

Original Article

A secure method of voting and planning based on quadratic voting

Hamid Devisti[a], Massoud Hadian Dehkordi[*a]

[a]*School of Mathematics and Computer Science, Iran University of Science and Technology, Tehran, Iran*

ABSTRACT: Recently, an innovative voting method named quadratic voting (QV) has been proposed which allows people to vote as much as they want, according to their preferences intensity. Little research has been done on the safe implementation of this method. In this paper, we first present a voting method based on QV. This method combines the voting and planning, and gives the ability to voters to express their opinions about the candidates programs in addition to voting. Then, a secure electronic voting protocol is proposed for implementing our method. This protocol gives the voters to check the verifiability of ballots and the safety of payment so that they would be sure that their votes are counted correctly.

Review History:
Received:20 May 2023
Revised:06 January 2024
Accepted:09 March 2024
Available Online:01 March 2025

Keywords:
Quadratic voting
Electronic voting
Verifiability

MSC (2020):
11T71; 14G50; 68P25

1. Introduction

In recent years, traditional voting based on paper ballots has been replaced by the electronic voting in many public surveys and messenger application. In the foreseeable future, election procedures will be internet-based with the help of artificial intelligence. This paper addresses the growing global concern regarding the diminishing public interest in elections, a foundational element of democratic societies. To tackle this challenge, two proposed solutions are discussed, focusing on the facilitation and enhancement of the electoral process. The first approach involves the implementation of electronic elections, aiming to reduce physical costs and save time. The second strategy suggests transforming elections into a social market accessible to the public, akin to a stock exchange. This paper aims to address the identified issues and provide solutions to reinvigorate public engagement in the electoral process.

The primary focus of this work revolves around the critical challenge associated with electronic elections: the preservation of voters' votes and the accurate counting of the total votes cast. Consequently, the paper centers on presenting an end-to-end verifiable electronic voting system as a viable solution. Informally, end-to-end verifiability encompasses three key characteristics, ensuring transparency and reliability throughout the voting process.

The outlined characteristics include:

- Each voter's ability to verify whether their vote has been cast.

*Corresponding author.
E-mail addresses: h_devisti@mathdep.iust.ac.ir, mhadian@iust.ac.ir

- Each voter's ability to confirm if their vote has been recorded as cast.
- The capacity for anyone to verify if all votes have been accurately tallied as recorded.

The paper delves into the technical aspects of achieving end-to-end verifiability, emphasizing the utilization of zero-knowledge proof algorithms and homomorphic encryption. zero-knowledge proof algorithms empower voters to authenticate the legitimacy of their votes, while homomorphic encryption provides assurance to third parties regarding the accuracy of the vote count. These cryptographic techniques play a pivotal role in addressing the identified challenges and ensuring the integrity of the electoral process. In conclusion, this paper contributes to the ongoing discourse on revitalizing democratic participation by proposing and exploring the implementation of end-to-end verifiable electronic voting systems. Through a comprehensive examination of cryptographic techniques, the paper aims to instill confidence in both voters and external entities, fostering a renewed interest and trust in the democratic electoral process.

Electronic voting combines the areas of software and hardware in such a way that an electoral process including registration, voting and counting of votes is done satisfactorily. Creating efficient and secure protocols is important in the software area. The following requirements must be met for an electronic voting protocol to be secure.

- *Eligibility:* Only eligible voters can vote in the election and every voter can cast only one vote.
- *Privacy:*The identity of voters and their votes must be stored secretly and not reveal their personal details.
- *Accuracy:* Voting protocols must be error-free. The votes must be correctly recorded and tallied. Incorrect votes should be discarded.
- *Verifiability:* Voters must be able to verify the correctness of their votes and final tallied result.
- *Integrity:* No one can modify/duplicate any ballot without being discovered.

1.1. contribution

In general, our contributions can be summarized as follows:

- We present a new method of voting based on quadratic voting (QV). We call it $QV-V\&P$ system (voting and planning based on quadratic voting) in the form of short. Two protocols zero-knowledge proof and proofs of partial knowledge are introduced to verify the content of ballots in section 3.
- We present two voting protocols that meet the security requirements for the voting method introduced in section 6.

The remainder of this paper is organized as follows. In section 2, the related works are reviewed. Section 3 is devoted to the preliminaries and setting requirements. In section 4, we deal with the definition and explanation of QV. In sections 5, we describe our $QV-V\&P$ System. In section 6, the implementation method and algorithms are presented, and finally, in section 7, we analyze the security of the proposed scheme.

2. Related Works

Over the last two decades, many papers have been published on electronic voting. The protocols presented in these papers focus on issues such as privacy, integrity, verifiability, robustness and receipt-freeness. The purpose of a voting protocol is to provide all of those properties, but this is not practical in the real world and some of these requirements are sometimes contradictory. The four key topics at the heart of the voting protocols are the cryptographic algorithm, the digital signature, the mix-net, and the proof of knowledge.

Cryptographic algorithms are widely used in all protocols in order to protect the ballot content and the personal information of voters. The cryptosystems, which is used in the voting protocol have the homomorphism property. Homomorphism property enables the voters and authorities to compute the results of election without disclosing the content of the ballots. Voters and authorities can also use this tool to create a common encryption public key. Papers [4, 11, 12, 19, 27, 30] have used the homomorphism property. We use homomorphism encryption to compute the sum of the votes and to build a common public key.

A blind signature is a form of digital signature in which the content of a message is concealed before singing. The resulting blind signature can be publicly verified against the original, unblinded message in the manner of a regular digital signature. Blind signatures are usually used in privacy-related protocols where the signer and message author are different parties. Generally the blind signature is used in voting protocol in order to conceal the identity of voters and the content of their ballots. Some voting protocols that use of the tool of blind signature are found in papers [8, 10, 20, 22, 34, 37].

Another technique, that is widely used in voting protocols is mix-net. A mix-net permute and modify the sequence of objects in order to disguise the relation between elements of original and final sequence. The proposed protocol in paper [3, 18, 25, 32] use mix-net to hide the the voter's information.

The proof of knowledge is a tool to check the validity of people votes and the way that they vote. In fact, proof of knowledge including zero-knowledge proof and proof of partial knowledge is an interactive protocol between two parties in which one party (prover) wants to prove the correctness of a statement to another (verifier) without revealing additional information. This feature is widely used in verifiable voting systems. A verifiable voting scheme gives assurances to voters that the election has been carried out on healthy way and their vote has been counted, without manipulating. Extensive research has been carried out on the field of verifiable voting systems. The papers [15, 21, 22, 23, 28, 33, 36] are related to verifiable voting system. The works that have been used more and their ideas have been generalized by us are summarized below. We use the works [6, 7, 17] to construct our proofs of knowledge. Tassa and et al proposed a secure voting protocol for score voting and ranked voting [16]. They applied the secret sharing scheme in a way that elections authorities could not get the results of the vote count alone. Yang and et al proposed a verifiable voting system based on web [36]. In their work, each ballot is considered as a squared matrix. We generalize this idea and use 3-dimensional matrix in our scheme.

Quadratic voting (QV) is the newest voting method which has been recently presented by Lalley and Weyl [29]. A full explanation of QV will be presented in Section 5. Many research has been done on the various aspects of this new method. For more study, the reader can refer to papers [24, 31, 35] Maskin study the relation between ranked-choice voting and quadratic voting. he believed [29] that the combination these two voting system can improve democracy. We present a new scheme by combining ranked-choice voting and quadratic voting. More details will be provided in Section 6 In implementation area, Park and Rivest Examine the Security Requirements of QV Implementation in election and survey setting [26]. They combined end-to-end verifiable voting method with anonymous payments and presented a new refund formula for QV. In this paper, we will deal with more details in implementation phase of a new version of QV. Our scheme surpasses the QV in two significant aspects. Firstly, the QV plan posits a singular issue for individuals to either agree or disagree with based on the outlined strategy in the article, determined through voting. In contrast, our proposed plan offers voters multiple programs to consider, allowing them to make choices based on the amount paid. This structure mirrors the advantages seen in plurality voting over majority voting, providing individuals with a broader spectrum of options. The second noteworthy distinction lies in the limited number of plans, to our knowledge, that have been developed for the secure implementation of this method in the context of electronic voting. Additionally, the primary article related to our work primarily explores the theoretical concept of QV voting, scrutinizing the issue through the lens of selection theory statistics. In contrast, our approach involves a practical implementation phase, setting our work apart by validating the proposed method in real-world scenarios.

3. Preliminaries

The important encryption tools in order to establish security in electronic voting scheme are discrete logarithm assumption (DLA) and argument of knowledge. Informally, DLA states that if the security parameters are chosen properly, it will be impossible for an attacker to solve a discrete logarithm equation. A zero-knowledge argument is a method by which one party (the prover) can prove to another (the verifier) that a given statement is true, while avoiding conveying to the verifier any information beyond the mere truth of the statement.

3.1. Discrete Logarithm Assumption

Suppose a probabilistic polynomial time (PPT) adversary $\mathcal{A}$ is a probabilistic interactive turing machine that runs in polynomial time in the security parameter λ. We will omit the security parameter λ from the notation when it is implicit.

Definition 3.1. *For all PPT adversaries $\mathcal{A}$ and for all $n \geq 2$ there exists a negligible function $\mu(\lambda)$ such that*

$$P\left[\begin{array}{l} \mathbb{G} = Setup(1^\lambda), \quad g_1, \ldots, g_n \xleftarrow{\$} \mathbb{G}; \\ a_1, \ldots, a_n \in \mathbb{Z}_p \leftarrow \mathcal{A}(\mathbb{G}, g_1, \ldots, g_n) \end{array} : \exists a_i \neq 0 \wedge \prod_{i=1}^{n} g_i^{a_i} = 1 \right] \leq \mu(\lambda)$$

We say $\prod_{i=1}^{n} g_i^{a_i} = 1$ is a non trivial discrete log relation between $g_1, \ldots, g_n$. The Discrete Log Relation assumption says that an adversary can't find a non-trivial relation between randomly selected group elements. For $n \neq 0$ this assumption is equivalent to the discrete-log assumption.

Notation	
N	number of voters
n	number of authorities
n_c	number of candidate
n_p	number of plans
n_v	number of votes
$V = \{V_i\}_{i=1}^{N}$	set of voters
pk_{V_i}	public key of V_i
sk_{V_i}	secret key of V_i
sign_{V_i}	blind signature of V_i
$A = \{A_i\}_{i=1}^{n}$	set of authorities
pk_{A_i}	public key of A_i
sk_{A_i}	secret key of A_i
sign_{A_i}	blind signature of A_i
B_i	submitted ballot matrix of V_i
$E(B_i)$	encrypted ballot matrix of V_i
x_{jk}^{i}	the value on position B_i
$(y_{jk}^{(i)}, y_{jk}^{(i)})$	the encrypted value of x_{jk}^{i}
$B_{j,k,l}^{(i)}$	the value position $(j,k,l)$of B_i
$C_{j,k,l}^{(i)}$	the encrypted value position (j,k,l) of $E(B_i)$
y	authorities common public key

3.2. Proof of Knowledge

The purpose of our proof of knowledge is providing an interactive method between voters (provers) and center (verifier) to ensure the correctness of the ballots. The voter prove to the election system that he knows the content of the ballot, without revealing additional information about it.

Zero-Knowledge Proof. Given a cyclic group $G = \langle g \rangle = \langle h \rangle$ and matrix $A_{n_c \times n_p}$ and $B_{n_c \times n_p}$ as public knowledge with arrays $g^{x_{ij}}$ and $h^{x_{ij}}$, where the values of x_{ij} are the the ballot plaintexts. The prover must convince the verifier that all corresponding arrays of $A_{n_c \times n_p}$ and $B_{n_c \times n_p}$ have the same exponentiation. The values of x_{ij} must be remain secret and the prover only knows these values. The steps of verification are as follows:

Step 1. Prover select a random matrix $R = (r_{ij})_{n_c \times n_p}$ whose arrays belong to $\mathbb{Z}_p$ and computes the matrix $T = (t_{ij})_{n_c \times n_p}$ and $T' = (t_{ij})_{n_c \times n_p}$ where $t_{ij} = g^{r_{ij}}$ and $t'_{ij} = h^{r_{ij}}$, then send T and T' to verifier.

Step 2. Verifier selects a random matrix $C = (c_{ij})_{n_c \times n_p}$, $(c_{ij} \in \mathbb{Z}_p)$ and sends C to prover.

Step 3. Prover computes matrix $S = X.C + R$ and sends S to verifier.

Step 4. Verifier verifies if $g^S = A^C.T$ and $h^S = B_2^C.T'$. In this equality, g^S, h^S, A^C, B^C are arranged in the form of a matrix shown below and equalities are checked in component-wise.

$$g^S = (g^{s_{ij}})_{n_c \times n_p}, \quad h^S = (h^{s_{ij}})_{n_c \times n_p}, \quad A^C = (a_{ij}^{c_{ij}})_{n_c \times n_p}, \quad B^C = (b_{ij}^{c_{ij}})_{n_c \times n_p}$$

Proof of Partial Knowledge. Our proof of partial knowledge consider each ballot as a set S of secret numbers, which its members are the arrays of ballot matrix. Suppose G is a cyclic group of prime order q and also suppose the sets $\{g_i\}_{i=1}^{l}$ and $\{h_i\}_{i=1}^{l}$ are distinct generators of G. The prover select a subset $X \subseteq S$ with $m = 2k$ members. He/She then divide up S into two separate subsets $X_1 = \{x_1, x_2, \ldots, x_k\}$ and $X_2 = \{x'_1, x'_2, \ldots, x'_k\}$ where $X = X_1 \cup X_2, X_1 \cap X_2 = \varnothing$. The prover must convince the verifier that either knows X_1 or X_2. suppose $A = \{A_i = g^{x_i}, \quad x_i \in X_1\}, B_i = \{h^{x'_i}, \quad x'_i \in X_2\}$ be the set of public knowledge. Prover must convince the verifier that she either knows the elements of X_1 or the elements of X_2. We assume the prover knows the X_1. The steps of verification are as follows:

Step 1. Prover chooses three sets $t = \{t_i\}_{i=1}^{k}$, $c = \{c_i\}_{i=1}^{k}$, $s = \{s_i\}_{i=1}^{k}$ where $t_i, c_i, s_i \in \mathbb{Z}_p$ and computes $T = \{T_i = g_i^{t_i}\}_{i=1}^{k}$, and $T' = \{T'_i = h^{s_i}/B_i^{c_i}\}_{i=1}^{k}$, then sends $T_1 = \prod_{i=1}^{k} T_i$ and $T_2 = \prod_{i=1}^{k} T'_i$ to verifier.

Step 2. The verifier select $r = \{r_i\}_{i=1}^{k}, r_i \in \mathbb{Z}_p$ and sends r to prover.

Step 3. The prover computes $d = \{d_i = c_i \oplus r_i\}_{i=1}^{k}$, $v = \{x_i.d_i + t_i\}_{l=1}^{k}$, sends c, r, v, s to verifier.($\oplus$ denotes XOR)

Step 4. The verifiers verify if the following conditions are satisfied.

$$\begin{cases} \prod_{i=1}^{k} g_i^{v_i} = \prod_{i=1}^{k} A_i^{d_i}.T_1 \\ \prod_{i=1}^{k} h_i^{s_i} = \prod_{i=1}^{k} B_i^{c_i}.T_2 \\ r = d \oplus c \end{cases}$$

3.3. Cryptographic Protocols

Most of the cryptosystems used in cryptographic protocols have the homomorphism property. This property enables us to encrypt the sum of two plaintext without additional computations and also, the sum of several plaintexts can be decrypted without revealing none of them. For example, consider ElGamal cryptosystem which is used in our protocol.

Distributed ELGamal Cryptosystem in $\mathbb{Z}_p$. *Suppose p be a prime such that the discrete logarithm problem in $(\mathbb{Z}_p^*, .)$ is infeasible, and suppose $G = \langle g \rangle$ be a cyclic subgruop of $\mathbb{Z}_p$ of prime order q, The plaintext and cyphertext space are respectively $\mathcal{P} = \mathbb{Z}_p^*$, $\mathcal{C} = \mathbb{Z}_p^* \times \mathbb{Z}_p^*$.*
The keyspace is as follows:

$$\mathcal{K} = \{(p, q, a, b, c) : \quad b \equiv a^c \pmod{p}\}$$

The values p, q, a and b are the public key, and c is the private key.
For $K = (p, q, a, c, b)$, and for a secret random number $k \in \mathbb{Z}_{p-1}$, define

$$E_K(x, k) = (y, y') \qquad \text{where} \qquad y = a^k \pmod{p}, \quad y' = g^x b^k.$$

For $y, y' \in \mathbb{Z}_p^*$, the encryption function is as follows:

$$D_K(y, y') = y'(y^c)^{-1} \pmod{p}$$

It is easily proven that ElGamal cryptosystem satisfy in homomorphism property. suppose $x_1, x_2 \in \mathbb{Z}_p$ and the corresponding encryption values are $E_K(x_1) = (a^{k_1}, g^{x_1} b^{k_1})$ and $E_K(x_2) = (a^{k_2}, g^{x_2} b^{k_2})$, then we have:

$$E(x_1) \times E(x_2) = (a^{k_1}, g^{x_1} b^{k_2}) \times (a^{k_2}, g^{x_2} b^{k_2}) = (a^{k_1+k_2}, g^{x_2+x_2} b^{k_1+k_2}) = E(x_1 + x_2)$$

4. Quadratic Voting

In the real world, there are several voting methods to choose the best option among existence choices. A voting system is a pair $E = (C, P)$ where C is the set of candidates or alternatives $\{c_1, \ldots, c_m\}$, $|C| = m$, and P is a profile consisting of a set of voters indexed by their preference orders,$\{\preceq_1, \ldots, \preceq_n\}$, $|P| = n$. Also, a voting function is defined as follows:

$$VF : C \to (\pi_C, S_{\pi_C})$$

where π_C is a permutation of C and is actually the order that the voter considers for the candidates and S_{π_C} is a vector whose elements are real numbers that voter assigns to each candidate. This is the most common form of defining a voting function.
The simplest method of voting is based on the rule of one-person-one-vote in which a person can choose one of the available options. Some voting method are proposed to improve the choices limitation problem. Approval voting, ranked voting, score voting and other methods give the voters the ability to have wide range of choosing.

Most voting methods focused on the rule of majority, which means that if the majority of people in the community agree on a person, that person will be formalized as a public choice. The tyranny of the majority is direct result of majority rule.

Various institutions have been tried over the years to solve this problem. Several suggestions including super-majority rule, weighted voting, cumulative voting, "mixed constitutions, and executive discretion are proposed to solving the problem. Approval voting, ranked voting, score voting and other methods gives the people the ability of selecting from a variety of options.
The main problem of these methods is that they do not apply voters' preferences adequately. A new method

presented by Posner and Weyl has been able to impose the intensity of people's preferences. They created a new framework of political decision making based on the theory of quadratic voting. They illustrated how quadratic voting solve the preference-aggregation problem by giving proper weight to preferences of varying intensity.
Quadratic Voting (QV) is a voting scheme for selecting one out of two candidates, which departs from the rule of *"one person, one vote"* and instead permits each eligible voter to cast multiple votes for any single candidate, thereby applying the intensity of his or her preference for the selected candidate.

According to 4, various voting systems can be provided. But in practice, only those systems can be implemented that satisfied criteria such as Majority criterion, Condorcet criterion, Monotonicity criterion and etc. QV is considered a new voting system from two aspects. First, it is relatively new among voting systems such as plurality voting, majority voting, score voting, and ranked voting. Secondly, this method has also been implemented in practice and meets the mentioned criteria of a voting system. The first distinguished feature of QV is the paying the price of votes, and the second is the increasing of price quadratically, that is, if buying one vote costs 1 \$, then buying two votes costs 4 \$, buying three votes costs 9 \$, and so on. After the election, the total income is redistributed equally among the voters. QV has two benefits simultaneously. The first benefit allows the voters to buy as many votes as they want and, the second benefit is the stabilizing of fairness principle, that is, the rich have to pay more to buy more votes. The reason for the quadratic increase in vote prices is explained in reference [29].

Weyl and Posner examine various aspects of QV. They show that quadratic voting achieves optimal efficiency in the sense that asymptotically, the system's utilitarian inefficiency tends to zero.

5. $QV - V\&P$ System

As mentioned in Section 4, QV permits people to express the intensity of their preferences and vote as much as they want provided that they pay the the square of the number of casted votes. Although the strength of QV is unlimited number of votes, it can sometimes appears as an weakness.

The creators of QV claimed that it is likely to be superior to a society employing one-person-one-vote majority rule, but they have examined it from the standpoint of distributive justice. It is only a part of fairness in the election. Another important part of the justice is about the investment opportunities, which means that all people have the equal position and situation to invest in the projects. As we know, this is not the case in most industrial project and owners of big companies have more chance than small companies and usual people.

Big companies compete to invest in the projects and get more profits, so they have more incentive to pay more amount of money and cast more votes in the election.

In the societies with high Class conflict, The value of the money is not the same for all people. For the middle class the value of money is related to basic needs, but for the rich class it is related to unnecessary needs and investment costs.

The reports have recently shown the top 0.01% richest individuals—the 520,000 people who have at least \$19 million now hold 11% of the world's wealth. Meanwhile, the share of global wealth owned by billionaires has grown from 1% in 1995 to 3% in 2021 poor and wealth [9]. So we think there should be some restrictions on QV. If there is an upper bound for the number of votes, the election will be more realistic and will prevent the vote-buying. This upper bound can be set based on criterion such as the Gini coefficient and per capita income.

Our proposed scheme is based on QV and contains several important features. As we know, the aim of each election is electing the most desirable people for executing projects in the best possible way.

In each election, people's opinions about the candidates are asked directly or indirectly, but people's opinions about the plans are ignored. It is usually assumed that the people coincide with the candidates goals and what people want is similar to what candidates want.

People outsource their wishes to the candidates of their choice and expect the candidates to follow their favorite programs. These are not completely equivalence in the real world and there are differences between peoples interests and candidates programs. We can have more democratic elections if people can be consulted about plans and goals. Our scheme combines the voting for selecting people and individual participation in decision-making. First, the main goals and candidates are published. Each candidate then ranks the goals based on their priorities and send this ranking to the bulletin board. Then, people vote based on their favorite candidates, their priorities and candidate's priorities toward announced goals. Finally, Each person decides to cast a number of votes in election and pay the square of the number of votes. If a person agrees with the plan, she will pay a number of votes for it, and if she disagrees with it, she will not pay any money.

We explain our scheme in the form of an example. Suppose four people compete for mayor seat in a municipal election and also suppose that there are four main goals for the coming years. These goals are building a park at city center, modernization of the public transport bus fleet, establishing an environmental association and creating a highway between the two points of the city. In addition to the preferred candidate, each person considers their own interests in relation to these four goals. Everyone can at most cast v votes and distributes v votes between

candidates and plans based on preferences and priorities of each voter. We suggest the maximum value $v = n_v \times n_c$ for the number of votes that everyone can cast. The purpose of this proposal is to give the ability to the voters to cast one vote for each pair (candidate, plane) providing that the equal tendency toward all candidates.

Table 1: The public election information

Plan	Candidate	Priority
building a park at city center	Alice	$(1, 3, 2, 4)$
modernization of the Public Transport Bus Fleet	Bob	$(1, 4, 3, 2)$
Establishing an environmental association	Candy	$(1, 2, 4, 3)$
creating a highway between the two points of the city	Dorbas	$(1, 2, 3, 4)$

5.1. Fairness and Financial Issues

At first glance, the elections in which votes are bought and sold, may be considered unfair, but clearly this problem also has a solution. QV's financial strategy consists of two t phases. The initial purchase phase operates under the principle of the square of the number of votes, while the subsequent phase involves distributing the collected amount from the elections among all voters. The refund amount is equal to the total amount collected divided by the number of voters. This approach, validated by Lalley and Weyl, establishes that, for the majority of participating voters the difference between the amount paid and the amount received is positive: That is, if v_P is the amount paid and v_R is the amount received by voter V, then $v_R - v_P \geq 0$. QV system and $QV - V\&P$ System have three advantageous features.

Empowering the Minority: QV allows the minority of voters to aspire to win the election by offering the possibility of securing victory through a higher monetary contribution.

Reassurance for the Majority: The majority of voters can be confident that they will not face defeat in the election, adding a layer of assurance to their participation.

Balanced Distribution: The total amount collected from the majority of voters, who purchase a smaller number of votes, is juxtaposed against the total amount contributed by the minority of voters purchasing a larger number of votes. This balance ensures that both factions possess a viable chance of winning the elections. Contrary to initial impressions, affluent individuals do not have an outright ability to dictate the election outcome.

Several instances of quadratic voting in real-world scenarios include:

1. Colorado House of Representatives: In April 2019, the Democratic caucus of the Colorado House of Representatives conducted a quadratic voting experiment during their money elections [13].
2. Volt Germany's Party Congress in Leipzig: Volt Germany, a pan-European party, utilized quadratic voting in their second party congress in Leipzig to determine the most valued topics in their party manifesto among members [5].
3. Gramado City Council: The city council of Gramado in Brazil employed quadratic voting to establish priorities for the year and to achieve consensus on tax amendments [1].

5.2. The structure of ballots

By ballot we mean the electronic ballot that can be displayed on a PC or mobile and voters complete it by checking mark or entering the numbers. Ballot used in elections can be created in the form of a matrix. Each voter can enter the number of votes in positions of the matrix based on his/her preference. The voters must ensure that the distribution of votes is done correctly. As mentioned, we set an upper bound for the number of votes in our scheme and the voter must be convinced that the summing up of their votes should not be more than v. An example of such ballot is as follows: Ballots can be designed in other ways for the convenience of voters, the security issues and computational simplification. The numbers 1 to v are recorded below each plan and voters are asked to specify the desired number of votes based on the candidate and the plan by filling the circles. The ballot is a corresponding for the previous ballot.

Table 2: Ballot of the first type

Candidate	Plan			
	Plan 1	Plan 2	Plan 3	Plan 4
Alice	4	0	0	0
Bob	1	1	0	0
Candy	0	0	0	2
Dorbas	0	1	0	0

Table 3: Ballot of the second type

	Plan 1				Plan 2				Plan 3			
	1	2	3	4	1	2	3	4	1	2	3	4
Alice	○	○	○	●	○	○	○	○	○	○	○	○
Bob	●	○	○	○	●	○	○	○	○	○	○	○
Candy	○	○	○	○	○	○	○	○	○	○	○	●
Dorbas	○	○	○	○	●	○	○	○	○	○	○	○

6. A Verifiable $QV - V\&P$ System

In this section, we present $QV - V\&P$ (Quadratic Voting and Planing) system and explain the rest of the details.

- *Voter:* Eligible voters can cast their ballots by paying the cost of votes according to payment rule.
- *Plan:* Prior to the election, the plans are determined by a poll or a general council, such as the municipality council or parliament, so that candidates can choose their priorities among them, then the plan is sent to the bulletin board.
- *Candidates:* Candidates that participate in the elections declare their priorities and send them to the bulletin board so that voters can make decisions about them.
- *Public bulletin board:* Essential information concerning the election is published on a secure public bulletin board. The content of the bulletin board includes the public key of authorities and voters, approved plans, submitted votes and other items. Every one can access the contents of the bulletin board at anytime, but no-one can alter or distort the existing data on it. Our system consists of the preparation phase, registration phase, ballot casting phase, ballot verification phase and tally stage.

6.1. preparation Phase

Authorities and voters generate their public and private keys. The common public key is computed using the authorities public keys, which is sent on the public bulletin board in order to encrypt each ballot before submission. Authorities choose their public and private key pairs $(x_i, y_i = g^{x_i})$ according to ElGamal cryptosystem and the public key of election is $y = \prod_{i=1}^{n} y_i$, where $x_i \in G = \langle g \rangle$. Voters use the public key of election y to encrypts their ballots.

6.2. Registration Phase

In the registration phase, each voter must visit a registration station in person and present her ID document to station authorities. The authorities prepare the list of eligible voters prior to voting phase and transfer it in the bulletin board. Validity of this list can be checked by the third party. The reason for this process is that in the tallying phase, only ballots will be counted that encrypted by the existence public key in table. So the valid votes are only casted by the eligible voters. Another reason is that each voter can only vote once and any attempt to re-vote will be revealed. The eligible voters must generate a key pair including a public key pk_{V_i} and a private key sk_{V_i}. After verification by the authorities, the eligible voters should upload their to the public bulletin board. Then, all authorized voters must sign their submissions using their sk_{V_i}, and others can verify their identities by using corresponding pk_{V_i}. In this case, Voters uses Digital Signature Algorithm (DSA) in order to guarantee the security of submitted ballot and resist to forgery.

6.3. Payment Phase

After registration, Each voter must pay the price of the votes according to the payment rule. One of the security requirements of this method is that the identity of the voter should be kept secret. Therefore the ATMs and banks payment systems is not reliable because transactions and the name of voters are recorded and information made available to the government.

An alternative to traditional payment systems is to use cryptocurrencies such as Bitcoin. Here we present a payment system based on Bitcoin.

Bitcoin Protocol. The Bitcoin protocol is a peer-to-peer electronic cash system which no trusted third party interfere in transactions between users [2]. In this online payment system, a payer (voter) sends a payment directly to a payee (financial unit of election system). Bitcoin uses cryptographic principles to protect transactions security and users identity.

The basic feature of bitcoin is that the users' identities are always hidden and all transactions are visible, so bitcoin can be a reliable financial system to implement QV-based methods. In order to use Bitcoin protocol the voters need to create a Bitcoin account and a wallet. The basic components for the cryptographic in the bitcoin protocol is summarized as follows:

- *One-way hash function*: The double-SHA256 hashing algorithm is used to hash transactions and to solve proof-of-work puzzles.
- *E-wallet* The bitcoins and their corresponding transaction hashes are stored in a database named, e-wallet . It is usually referred to as the bitcoin wallet
- *Bitcoin*: Bitcoin is a digital currency interchanged between users over the Bitcoin network. BTC or B is the currency symbol for bitcoin.
- *Proof of work :* is a form of cryptographic proof in which one party (the prover) proves to others (the verifier) that a certain amount of a specific computational effort has been expended. Verifier can subsequently confirm this expense with minimal effort on their part.

  Miners perform proof-of-work calculations on transaction blocks. This process is called mining and is performed by mining software. Mining yields bitcoins for the miner as a reward.
- *Block-Chain:* The block-chain is a public financial center which stores processed transactions. It is necessary to make payment, the users create a transaction with all pertaining information including the address of the payee, the amount of bitcoins and a challenge. The transaction is distributed to all network nodes. The node puts this transaction into a block and attempt to solve the proof-of work for this block.If a node discovers a solution to the challenge, it distributes it to all other nodes and If all transactions are valid and not spent before, then the other nodes accept the block and start working on the next block in the chain. It is sometimes possible that the chain encounter the forks because two blocks were mined and broadcasted simultaneously. The longest block-chain is considered to be the correct by the nodes. Coin-base and regular transactions are two types of transactions.
- *Transactions:* Coin-base transactions are used for new bitcoins, and regular transactions are used for transferring of bitcoins between users.

This procedure is done with the cooperation of voters and authorities. As was said, voters must produce a pair of public and private keys to perform transactions. Here we will explain how to pay for the vote with the help of ElGamal blind signature [14]. V_i selects the number of votes v_i and published the commitment $a_i = E(v_i, k)$ by using his private key. Then, V_i create blind message $a_i' = \text{blind}_{V_i}(a_i, r_i)$, where r_i is random blinding factor. Then, he/she go to the cashier and asks her a signature for a_i'. The cashier checks the identity of V_i and give him the signature $d_i = \text{sign}_A(a_i')$, where $\text{sign}_A(a_i')$ is election signature scheme produced by A. Next, cashier sends d_i back to V_i At the end of process, the list of voters' details and signatures will be sent to the bulletin board. and each voter can see his commitment $(ID_i, a_i')$and others commitment on the public board.

Voter V_i retrieves the signature $b_i = \text{unblind}(d_i, r_i)$ for the commitment value of a_i. He/She verifies if y_i is A's signature for the commitment a_i. If the verification reject, V_i can claim the invalid signature by showing that (a_i, b_i) is invalid.

When the election system issue PBCs, V_i received one of these PBCs. the voter checks the validity of the bitcoin address include in the PBCs contain coins. Then, V_i open and extract the covered private key which allows to him the access to the corresponding Bitcoin address in the PBC. The voter V_i generate a pair of private key V_iPBK and a Bitcoin address V_iBA that will be used for voting. Only V_i can perform transactions, and so, no one except, V_i can link the identity of the owner of V_iBA to the bitcoin address in the PBC.

At the beginning of election, A has produced a pair of private key $A.BPK$ and a Bitcoin address $A.BA$ which

published publicly. The eligible voters who received a signature from election system should register the Bitcoin Address that they use it for voting.
Voter V_i produce a Bitcoin transaction using V_iBA as input address nad $A.BA$ as output address. The voter sends an arbitrary amount of BTC and the pair of (a_i, v_i) to A. After confirming the details of the transaction, V_i sends the transaction to the bitcoin network. Authorities A can check the validity (a_i, y_i) by having the public information of each voter. If the validation is successful, A published a list all of the Bitcoin address that send the correct signature y_i of the commitment a_i given by (V_iBA, a_i, b_i). At the end of this stage, the number of entries in the list that contain (ID_i, a_i') should be equal to the number of entries in the list that contains (V_iBA, a_i, b_i). Since a_i contains the vote v_i, A can just check and collect the list that contains (V_iBA, a_i, b_i) Sinse a_i contain the vote v_i., election authorities can just check and collect the list contains (V_iBA, a_i, v_i).

6.4. Ballot Casting Phase

The Voters who take part in the election, pay a fee for the number of votes which they have bought and then distribute votes among the candidates and plans. Suppose there are n_c candidates and n_p plans in the election. The input values of the ballot matrix the input must be encrypted using the ElGamal cryptosystem. In order to hide the ballot content, each voters must encrypt the ballots using the public key PK and send it to the center.
Suppose the voter V_i, decide to cast a ballot in the voting phase and sending his vote to the center. For this purpose, V_i selects $n_c \times n_p$ random numbers r_{ij} from the $\mathbb{Z}_p$ and encrypts each position of ballot matrix according to the ElGamal cryptosystem. For the ballot B_i described in Section 5, we have:

$$B_i = \begin{pmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{pmatrix}$$

where the encrypted values of B_i according to to ElGamal cryptosystem are as follow:

$$(y_{jk}, y'_{jk}) = (g^{r_{jk}}, g^{x_{jk}} y^{r_{jk}}), \quad r_{jk} \in \mathbb{Z}_p, \quad 1 \le j \le n_c \qquad 1 \le k \le n_p$$

$$E(B_i) = \begin{pmatrix} (y^i_{11}, y'_{11}) & (y_{12}, y'_{12}) & (y_{13}, y'_{13}) & (y_{14}, y'_{14}) \\ (y_{21}, y'_{21}) & (y_{22}, y'_{22}) & (y_{23}, y'_{23}) & (y_{23}, y'_{23}) \\ (y_{31}, y'_{31}) & (y_{32}, y'_{32}) & (y_{33}, y'_{33}) & (y_{34}, y'_{34}) \\ (y_{41}, y'_{41}) & (y_{42}, y'_{42}) & (y_{43}, y'_{43}) & (y_{44}, y'_{44}) \end{pmatrix}$$

Before explaining the voting algorithm, we should note that $E(B_i)$ is shown according to its column. The verification can be carried out base on rows of $E(B_i)$ or any family of subsets containing elemnts of $E(B_i)$, provided that the number of elements of each subset be even.

$$A = (X_1, X_2, \ldots, X_n),$$

We consider each column of $E(B_i)$ as a set and then divide it into two separate sets X_k^1 and X_k^2 so that the following conditions are satisfied

$$X_k^1 \cap X_k^2 = \emptyset, \qquad X_k = X_k^1 \cup X_k^2$$

In algorithm 1, we describe the voting method and proof of partial knowledge when the ballots are in the form of table 1. We are now present it for the case that the ballots are in the form of table 2. Here, each ballot is in the form of 3-dimentional matrix $B^i_{j\times k\times l}$, where the position (j, k, l) shows whether the voter i devote to candidate j for plan k, l votes or not. If the answer is yes, he/she fills the corresponding circle and $B^i_{j\times k\times l} = 1$, otherwise $B^i_{j\times k\times l} = 0$.

7. Tallying phase

After the voting process is done completely, The election authorities are present to count the votes. Because all the authorities have declared their public key and details to the electoral system in the registration stage, no one can deceive others or change their private key. The counting of votes and the final results are obtained by multiplying the ballot matrices component-wise. This is due to the homomorphism property of ElGamal cryptosystem. As was seen in section 1, we have

$$E(x_{jk}) = (y^{(i)}_{kj}, y^{(i)}_{kj}) = (g^{r_{jk}}, g^{x_{jk}} y^{r_{jk}}) \quad 1 \le j \le n_c, 1 \le k \le n_p$$

Algorithm 1: Voting Algorithm 1

Input: $V_i, (ID_i, a_i), B_i, X_k = (X_k^1, X_k^2)$ Hash function $h : \{0,1\}^* \to G$
Output: encrypted ballot $E(B_i)$, proofs of $E(B_i)$

begin
  for $j \leftarrow 1$ ***to*** n_c **do**
    for $k \leftarrow 1$ ***to*** n_p **do**
      $E(x_{jk}) = (y_{jk}, y'_{jk}) = (g^{r_{jk}}, g^{x_{jk}} y^{r_{jk}}), \quad r_{jk} \in Z_q$
      $t = (t_i)_{i=1}^{l}, c = (c_i)_{i=1}^{l}, s = (s_i)_{i=1}^{l}, r = (r_i)_{i=1}^{l}$
      where $t_i, c_i, s_i, r_i, x_i \in \mathbb{Z}_p$
      if *Verifier choose* X_k^1 **then**
        $A_k^1 = \{A_i = g_i^{x_{ij}}, \quad x_{ij} \in X_k^1\}, \quad B_k^2 = \{B_i = h_i^{x_{ij}}, \quad x_{ij} \in X_k^2\}$
        $T = \{T_i = g^{t_i}\}_{i=1}^{l}, \quad T' = \{T'_i = h^{t_i}/B_i^{c_i}\}_{i=1}^{l},$
        $T_1 = \prod_{i=1}^{l} T_i, \quad T_2 = \prod_{i=1}^{l} T'_i$
        $\prod_{i=1}^{l} g_l^{v_i} = \prod_{i=1}^{l} A_i^{d_i}.T_1 \quad \prod_{1}^{l} h_i^{s_i} = \prod_{i=1}^{l} B_i^{c_i}.T_2, \quad r = d \oplus c \quad \Rightarrow \quad$ verifying $\quad X_k$
      end if
      else if *Verifier choose* X_k^2 **then**
        $T = \{T_i = h^{t_i}\}_{i=1}^{l}, \quad T' = \{T'_i = g^{t_i}/A_i^{c_i}\}_{i=1}^{l}$
        $T_1 = \prod_{i=1}^{l} T_i \quad , T_2 = \prod_{i=1}^{l} T'_i$
        $\prod_{i=1}^{l} h^{v_i} = \prod_{i=1}^{l} B_i^{d_i}.T_1 \quad \prod_{i=1}^{l} h^{s_i} \prod_{i=1}^{l} B_i^{c_i}.T_2, \quad r = d \oplus c \quad \Rightarrow \quad$ verifying $\quad X_k$
      end if
      $r = \{hash(X_1^k(i) \parallel X_k^2(i) \parallel T_i \parallel T'_i)\}_{i=1}^{l}$ $d = \{d_i = c_i \oplus r_i\}_{i=1}^{k}$, $v = \{x_i.d_i + t_i\}_{l=1}^{k}$
      $PC_k^i = PoPK\{X_k, T, T', r, c, s, v\}$ proof of columnX_k
    end for
  end for
end
digital signature: $S_{V_i} = \text{sign}_{V_i}(E(B_i) \parallel PoPKX_k{}^{(i)}, sk_{V_i})$
submitted document: $E(B_i), PC_k^i, S_{V_i}, k \in [1, n_p]$

by component-wise multiplying of all $E(B_i)$'s elements, we have

$$E(\sum_{i=1}^{N} x_{jk}) = \prod_{i=1}^{N} E^{(i)}(x_{jk})$$

then, the result was obtained as follows:

$$D(E(\sum_{i=1}^{N} x_{jk})) = \frac{\prod_{i=1}^{N} g^{x_{jk}} (\prod_{i=1}^{n} y_i)^{r_{kj}}}{\prod_{i=1}^{n} (\prod_{i=1}^{N} y^{r_{kj}})^{x_i}}$$

which simplifies to

$$D(E(\sum_{i=1}^{N} x_{jk})) = g^{\sum_{i=1}^{N} x_{jk}}$$

where the result is revealed by computing a discrete logarithm. and eventually, the candidate who received the most votes will be selected as the winner, and the plan that received the most votes,will be introduced as the selected plan to be implemented.

8. Security Analysis

The first question is, how Proof of knowledge 3.2 can prove the correctness of the ballot matrix. In order to answer this question, we must emphasize that a ballot matrix is valid whenever each of its arrays is correct and the sum

Algorithm 2: Voting Algorithm 2

Input: V_i, B_i, PK Hash function $h : \{0,1\}^* \to G$
Output: encrypted ballot $E(B_i)$, proofs of $E(B_i)$.

begin
  for $j \leftarrow 1$ ***to*** n_c **do**
    for $k \leftarrow 1$ ***to*** n_p **do**
      for $l \leftarrow 1$ ***to*** n_v **do**
        $r, t, v, s \in \mathbf{Z_q}, T_0 = g^t, T_1 = y^t$
        if $B^i_{j\times k\times l} = 1$ **then**
          $C^{(i)}_{j,k,l} = E(1) = (c_1, c_2) = \{g^r, g.y^r\}, T_2 = y^s/c_2^v$
        end if
        else if $B^i_{j\times k\times l} = 0$ **then**
          $C^{(i)}_{j,k,l} = E(0) = (c_1, c_2) = \{g^r, y^r\}, T_2 = g^v.y^s/c_2^v$
        end if
        $v = hash(c_1||c_2||T_0||T_1||T_2), \quad w = v \oplus u, \quad s = r.w + t$
        $PC^{(i)}_{j,k,l} = PoPK\{C^{(i)}_{j,k,l}, T_0, T_1, T_2, u, w, s, v\} \Rightarrow$ proof of ciphertext$C^{(i)}_{j,k,l}$
      end for
    end for
  end for
  for $j \leftarrow 1$ ***to*** n_c **do**
    for $k \leftarrow 1$ ***to*** n_p **do**
      $A_{n_c\times n_p}(j,k) = g^{sum_{j,k}}, B_{n_c\times n_p}(j,k) = h$
      $sum_{j,k} = \sum_{l=1}^{n_v} B_{j,k,l}$
      $R = (r_{jk})_{n_c\times n_p}, \quad T = (t_{jk})_{n_c\times n_p}, \quad T' = (t'_{jk})_{n_c\times n_p}$
      $C = (c_{jk})_{n_c\times n_p}, S = A.C + R$ where $r_{jk}, t_{jk}, t'_{jk}, c_{jk} \in \mathcal{Z}_{\text{II}}$
      if $g^S = A^C.T \quad \& \quad h^S = B^C.T'$ **then**
        The statement $(sum_{j,k} = 1)$ is verified
      end if
      else if then
        The verification is failed
      end if
      $P^{(i)}_{j,k} = PoZK^{(i)}_{j,k}\{A, B, T, T', S\} \Rightarrow$ proof the$(g^{sum_{j,k}} = h)$
    end for
  end for
end
digital signature: $S_{V_i} = \text{sign}_{V_i}(E(B_i) \parallel PoPKX_k{}^{(i)} \parallel PoZK^{(i)}_{j,k}, sk_{V_i})$
submitted document: $E(B_i), PC^{(i)}_{j,k,l}, P^{(i)}_{j,k}, S_{V_i}, j \in [1, n_c], k \in [1, n_p]$

of all arrays is equal to the number of votes purchased by the voter. It is therefore required to carry out $n_c \times n_p$ proof of partial knowledge and one zero-knowledge proof between voter and election system to check the validity of the ballot matrix. We show that this can be done by reducing the number of transactions to $n_p + 1$.

Theorem 8.1. *The proof of partial knowledge introduced in 3.2 most likely guarantees the validity of the ballot matrix.*

Proof. We assumed that each column k of matrix $E(B_i)$ has $n_p = 2l$ elements. The voter V_i should published the public information associated with each value of x_{jk}, including A, B, g_i, h_i. In the proposed proof of partial knowledge, we give the ability to the election system to choose two subset with l elements of column k at random. The election system asks the user to publish the relevant commitments and randomly selects one of two subset again. The possible choices for X^1_k and X^2_k is $\binom{2l}{l}$ and the probability of selecting X^1_k by the center is $\frac{1}{2}\frac{1}{\binom{2l}{l}}$. Obviously, if the prover knows the value of all x_{jk}, then he/she can convince the verifier that he/she knows the sum of x_{jk}'s, but the reverse of the statement is not necessarily true, that is, if one party knows the sum of x_{jk}'s, he/she may not know each of those values. That's why we give the verifier the option to choose any subset of X_k, elements that he she wants. If the prover really knows all the values of x_i, then he can answer any of questions about the sum of desired set elements. When the number of versifier's questions about the values of x_{jk} exceed the number of n_p, he/she can behave in such a way that form a system of linear equations consists of n_p variables and n_r (the

number of transaction) equations as follows:

$$\begin{cases} x_1^1 + x_2^1 + \cdots + x_l^1 = y_1 & (\text{iteration1}) \\ \vdots & \vdots \\ x_1^i + x_2^i + \cdots + x_l^i = y_i & (\text{iteration } i) \\ \vdots & \vdots \\ x_1^{n_r} + x_2^{n_r} + \cdots + x_l^{n_r} = y_{n_r} & (\text{iteration } n_r) \end{cases}$$

where, $x_i^j \in X_k$. The prover has indirectly proved that she knows the answer of this linear equations system and consequently he/she knows all the value of x_{jk}. □

Theorem 8.2. *In our proposed scheme, an attacker cannot extract the information of ballot B_i belonging to voter V_i.*

Proof. If an attacker wants to access the content of ballot B_i, then he needs to decrypt $E(B_i)$. Decrypting $E(B_i)$ requires solving $2 \times n_c \times n_p$ discrete logarithm equations, which is impossible due to DLA introduced in 3.1. □

Conclusions

In this paper, we present a method of voting and planning based on quadratic voting and propose a secure voting protocol for it . This method can improve quadratic voting. The proof of knowledge presented in this paper reduces the number of transactions between the voter and the election system and has the ability to confirm the validity of the ballot with a smaller number of transactions between the two parties.

References

[1] *The mathematical method that could offer a fairer way to vote.* https://www.economist.com/christmas-specials/2021/12/18/the-mathematical-method-that-could-offer-a-fairer-way-to-vote. The Economist, 2021.

[2] V. Augoye, *Electronic voting: An electronic voting scheme using the secure payment card system*, tech. rep., Information Security Group Royal Holloway, University of London Egham, Surrey TW20 0EX, UK, 2013.

[3] V. Augoye and A. Tomlinson, *Mutual authentication in electronic voting schemes*, in 16th Annual Conference on Privacy, Security and Trust (PST), 2018, pp. 1–2.

[4] O. Baudron, P.-A. Fouque, D. Pointcheval, J. Stern, and G. Poupard, *Practical multi-candidate election system*, in Proceedings of the Twentieth Annual ACM Symposium on Principles of Distributed Computing, PODC '01, New York, NY, USA, 2001, Association for Computing Machinery, pp. 274–283.

[5] J. Bone, *The new voting system that could save our democracies.* https://www.nesta.org.uk/feature/ten-predictions-2020/new-voting-system-could-save-our-democracies, 2019.

[6] B. Bünz, J. Bootle, D. Boneh, A. Poelstra, P. Wuille, and G. Maxwell, *Bulletproofs: Short proofs for confidential transactions and more*, in 2018 IEEE Symposium on Security and Privacy (SP), 2018, pp. 315–334.

[7] J. Camenisch and M. Stadler, *Proof systems for general statements about discrete logarithms*, tech. rep., ETH Zurich, Department of Computer Science, 1997.

[8] M. Chaieb, M. Koscina, S. Yousfi, P. Lafourcade, and R. Robbana, *Dabsters: Distributed authorities using blind signature to effect robust security in e-voting*, in Proceedings of the 16th International Joint Conference on e-Business and Telecommunications - SECRYPT, INSTICC, SciTePress, 2019, pp. 228–235.

[9] L. Chancel, T. Piketty, E. Saez, and G. Zucman, *World inequality report 2022*, Harvard University Press, 2022.

[10] D. Chaum, *Blind signatures for untraceable payments*, in Advances in Cryptology, D. Chaum, R. L. Rivest, and A. T. Sherman, eds., Boston, MA, 1983, Springer US, pp. 199–203.

[11] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène, *A homomorphic lwe based e-voting scheme*, in Proceedings of the 7th International Workshop on Post-Quantum Cryptography - Volume 9606, PQCrypto 2016, Berlin, Heidelberg, 2016, Springer-Verlag, p. 245–265.

[12] J. D. Cohen and M. J. Fischer, *A robust and verifiable cryptographically secure election scheme*, in 26th Annual Symposium on Foundations of Computer Science (sfcs 1985), 1985, pp. 372–382.

[13] P. Coy, *A new way of voting that makes zealotry expensive*, Bloomberg L.P., (2019).

[14] J. P. Cruz and Y. Kaji, *E-voting system based on the Bitcoin protocol and blind signatures*, IPSJ Transactions on Mathematical Modeling and Its Applications, 10 (2017), pp. 14–22.

[15] C. Culnane and S. Schneider, *A peered bulletin board for robust use in verifiable voting systems*, in IEEE 27th Computer Security Foundations Symposium, 2014, pp. 169–183.

[16] L. Dery, T. Tassa, and A. Yanai, *Fear not, vote truthfully: Secure multiparty computation of score based rules*, Expert Systems with Applications, 168 (2021), p. 114434.

[17] Y. G. Desmedt, ed., *Proofs of Partial Knowledge and Simplified Design of Witness Hiding Protocols*, Springer, Berlin, Heidelberg, 1994.

[18] W. Y.-m. GAO Hu-ming, WANG Ji-lin, *An electronic voting scheme based on a new mix net*, Acta Electronica Sinica, 32 (2004), pp. 1047–1049.

[19] P. Grontas, A. Pagourtzis, and A. Zacharakis, *Coercion resistance in a practical secret voting scheme for large scale elections*, in 2017 14th International Symposium on Pervasive Systems, Algorithms and Networks & 2017 11th International Conference on Frontier of Computer Science and Technology & 2017 Third International Symposium of Creative Computing (ISPAN-FCST-ISCC), 2017, pp. 514–519.

[20] S. James, N. Gayathri, and P. V. Reddy, *Pairing free identity-based blind signature scheme with message recovery*, Cryptography, 2 (2018).

[21] R. Joaquim, P. Ferreira, and C. Ribeiro, *Eviv: An end-to-end verifiable internet voting system*, Computers & Security, 32 (2013), pp. 170–191.

[22] M. Kumar, S. Chand, and C. P. Katti, *A secure end-to-end verifiable internet-voting system using identity-based blind signature*, IEEE Systems Journal, 14 (2020), pp. 2032–2041.

[23] R. Küsters, J. Liedtke, J. Müller, D. Rausch, and A. Vogt, *Ordinos: A verifiable tally-hiding e-voting system*, in 2020 IEEE European Symposium on Security and Privacy (EuroS&P), 2020, pp. 216–235.

[24] S. P. Lalley and E. G. Weyl, *Quadratic voting: How mechanism design can radicalize democracy*, AEA Papers and Proceedings, 108 (2018), pp. 33–37.

[25] M. Ohkubo, F. Miura, M. Abe, A. Fujioka, and T. Okamoto, *An improvement on a practical secret voting scheme*, in Information Security. ISW 1999, M. Mambo and Y. Zheng, eds., 1999, pp. 225–234.

[26] S. Park and R. L. Rivest, *Towards secure quadratic voting*, Public Choice, 172 (2017), pp. 151–175.

[27] K. Peng, R. Aditya, C. Boyd, E. Dawson, and B. Lee, *Multiplicative homomorphic e-voting*, in Proceedings of the 5th International Conference on Cryptology in India, INDOCRYPT'04, Berlin, Heidelberg, 2004, Springer-Verlag, pp. 61–72.

[28] A. J. Perez and E. N. Ceesay, *Improving end-to-end verifiable voting systems with blockchain technologies*, in IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2018, pp. 1108–1115.

[29] E. A. Posner and E. G. Weyl, *Voting squared: Quadratic voting in democratic politics*, Vanderbilt Law Review, 68 (2015), pp. 441–500.

[30] H. Pu, Z. Cui, and T. Liu, *An electronic voting scheme using secure multi-party computation based on secret sharing*, International Journal of Network Security, 23 (2021), pp. 997–1004.

[31] D. Quarfoot, D. von Kohorn, K. Slavin, R. Sutherland, D. Goldstein, and E. Konar, *Quadratic voting in the wild: real people, real votes*, Public Choice, 172 (2017), pp. 283–303.

[32] K. Sako and J. Kilian, *Receipt-free mix-type voting scheme: a practical solution to the implementation of a voting booth*, in Proceedings of the 14th Annual International Conference on Theory and Application of Cryptographic Techniques, EUROCRYPT'95, Berlin, Heidelberg, 1995, Springer-Verlag, pp. 393–403.

[33] B. Smyth, *Mind the gap: Individual- and universal-verifiability plus cast-as-intended don't yield verifiable voting systems*. Cryptology ePrint Archive, 2020.

[34] G. K. Verma, B. Singh, and H. Singh, *Provably secure certificate-based proxy blind signature scheme from pairings*, Information Sciences, 468 (2018), pp. 1–13.

[35] E. G. Weyl, *The robustness of quadratic voting*, Public Choice, 172 (2017), pp. 75–107.

[36] X. Yang, X. Yi, C. Ryan, R. van Schyndel, F. Han, S. Nepal, and A. Song, *A verifiable ranked choice internet voting system*, in Web Information Systems Engineering – WISE 2017, A. Bouguettaya, Y. Gao, A. Klimenko, L. Chen, X. Zhang, F. Dzerzhinskiy, W. Jia, S. V. Klimenko, and Q. Li, eds., vol. 6, Springer International Publishing, 2017, pp. 490–501.

[37] H. Zhu, Y.-a. Tan, L. Zhu, Q. Zhang, and Y. Li, *An efficient identity-based proxy blind signature for semioffline services*, Wireless Communications and Mobile Computing, 2018 (2018), p. 5401890.

Please cite this article using:

Hamid Devisti, Massoud Hadian Dehkordi, A secure method of voting and planning based on quadratic voting, AUT J. Math. Comput., 6(2) (2025) 177-191
https://doi.org/10.22060/AJMC.2024.22420.1157