A Novel Quantum Voting System Based on Quantum Blind Signature without Entanglement

Yu-Yuan Chou[1], Wen-Ching Wu [2], Jue-Sam Chou [3*]

[1]Department of Physics, National Central University, Taiwan

warrior1819150@gmail.com

[2] Department of Information Management, Nanhua University, Taiwan

10769553@nhu.edu.tw

[3]Department of Information Management, Nanhua University, Taiwan

*: corresponding author: jschou@nhu.edu.tw; jschou54@gmail.com

Tel: 886+ (05)+272-1001 ext.56536

Abstract

In this paper, we specifically review Xu et al.'s quantum blind signature scheme for distributed e-voting systems, which primarily focuses on simulating real-life e-voting. The scheme aims to ensure voter anonymity in an e-voting system. However, we found that it not only suffers from identity impersonation attacks but also lacks the blindness property essential to a blind quantum signature. To address these shortcomings, we propose a new quantum blind signature scheme that leverages quantum mechanical properties and a one-way hash function. Considering that a voting scheme naturally involves an election committee member blindly signing a ballot embedded with the name of the selected candidate, we use our quantum blind signature as the foundation to design a quantum voting system. This system effectively prevents the repudiation and counterfeiting issues present in Xu et al.'s scheme. Additionally, we provide relevant security analyses to support our theoretical framework. The results demonstrate that our scheme outperforms existing literature not only in terms of e-voting security properties—such as undeniability, anonymity, and untraceability—but also in conceptual simplicity and computational efficiency.

Keywords: quantum signature, quantum blind signature, quantum voting system, quantum measurement.

1. Introduction

With the development of modern data science and technology, the importance of information security requirements has dramatically increased, especially in the context of user authentication for online electronic commerce transactions. This has led to a rising trend in the study of digital signature schemes. Signature schemes can be applied in various fields such as payment systems, commercial contracts, voting, and more. Many cryptographic researchers have contributed to this field, working on secure signature schemes ranging from general signatures [1–7] and proxy signatures [8–35] to their variants, such as deniable authentication with a designated verifier [36–51] and k-out-of-n oblivious transfer protocols [52–80]. All of these schemes enable a signer to sign a message that can then be verified either publicly or by a designated verifier [81–102].

Recently, several quantum blind signature schemes have been proposed [85, 92, 94], in which the verifier cannot discern the content of the signature. This property makes such schemes applicable to many areas, among which voting is one of the most important activities in a democratic country. When developing a voting system, maintaining the secrecy of the candidate selection is crucial. The ballot, embedded with the candidate's name, must be blindly signed by the committee and then anonymously transmitted to the ballot counting center. The center will verify its validity and increment the vote count for the corresponding candidate under the constraint that the voting content cannot be altered. Clearly, both the voter’s identity verification and the voting process correspond to signature verification for identity authentication and blind signing, respectively. That is, a voting system requires that the government agency cannot know the selected candidate on the ballot but must confirm that the ballot is genuinely from a legitimate voter. The only way to achieve this goal is by adopting a blind signature scheme in the cryptographic field. In 2011, Xu et al. [94] proposed a quantum group blind signature scheme without entanglement and claimed that their scheme meets the security requirements of a voting system. They asserted that the essential properties of a blind signature are (1) blindness, (2) unforgeability, (3) undeniability, and (4) anonymity. Blindness means that the signer can sign a message without any party being able to access the embedded voting information. However, upon examining the

protocol, we found that it does not satisfy the undeniability property because each involved pair must pre-share a common secret. This causes the scheme to suffer from a deniability problem: the original signer, Alice, can deny having signed the signature by claiming it was generated by the verifier, Bob, who can also use their shared session key K_{AB} to produce the same signature, despite the message having been signed by Alice herself. Therefore, to design a truly secure voting system, we propose a quantum blind signature scheme based on asymmetric quantum cryptography. Our proposed scheme not only overcomes the drawbacks identified in [94] but also ensures the secrecy and unforgeability of the voter's candidate selection, making it suitable for practical implementation.Aside from scheme [94], there were also several protocols proposed regarding voting scheme [113-128] for the last two decades. Among them, protocols [120-128] are based on conventional cryptography or blockchain security. Protocol [121] also use quantity key distribution (QKD) for the key sharing. And protocols [113-119] are quantum operation based. Due to the quantum computer has a massive parallel computing power by exponentially speed up over the classic computer, it will be a major trend for being adopted as a computing device in the near future. For this reason, in this paper we will explore the possibilities of using quantum computer to fulfil an election activity in a democratic country. We have surveyed the excellent literature and found that they each has important contributions to this field. However, there still exist several drawbacks in each scheme. We listed the limitations mentioned by the author in each of them as follows.

Hillery Mark et al. [113] address two key requirements: privacy (anonymity) and security (prevention of double voting). However, their scheme cannot prevent attacks by colluding parties, illegal voting operations, or cheating authorities. Xue Peng and Xin Zhang [117] share the same drawbacks as [113] and additionally do not demonstrate the voter authentication process. Moreover, if the parameter m is not sufficiently large, their scheme is vulnerable to man-in-the-middle attacks. Shi Wei-Min et al. [116] require a secure channel, while Zheng Mengce et al. [119] rely on underlying quantum key distribution (QKD). However, as noted by Finogina Tamara and Javier Herranz [121], authenticating the quantum channel for QKD necessitates a pre-shared secret to agree upon common bases. Horoshko Dmitri and Sergei Kilin [115] have the limitation of being unable to guarantee anonymity for a single voting act and provide only probabilistic eavesdropping detection. Dolev Shahar, Itamar Pitowsky, and Boaz Tamir

[114] lack flexibility, as their scheme must be adapted to accommodate elections with three or more candidates. Vaccaro, Joan Alfina, Joseph Spring, and Anthony Chefles [118] face the issue that, as their scheme is modified to enhance vote privacy, the restrictions on possible individual vote values become weaker. Based on this survey of recent quantum voting literature and inspired by Wang Feihu et al. [130], who demonstrate that quantum rotation operations are highly feasible with current technology, and Piétri Yoann [129], who notes that hash-based cryptography is considered quantum-safe and has been selected by NIST, this paper, like Shi Wei-Min et al. [116], utilizes single-qubit rotations and a one-way hash function to design our protocol.

The rest of this article is organized as follows. In Section 2, we first review Xu et al.'s quantum group blind signature as an example to explore the necessary properties of a quantum e-voting system and to describe its weaknesses. Section 3 presents our two designed sub-schemes [107,108] and defines the roles used in our proposed quantum voting scheme. These two schemes include an undeniable quantum signature and a quantum blind signature. In Section 4, we propose our quantum voting system. The security analyses are provided in Section 5. Finally, a conclusion is given in Section 6.

2. Review of Xu et al's quantum group blind signature

In this section, we first provide a brief overview of Xu et al.'s quantum group blind signature scheme [94] in Section 2.1. Next, we discuss its vulnerabilities in Section 2.2. Following that, the voting environment scenario and the associated security requirements, as outlined by Xu et al., are presented in Sections 2.3 and 2.4, respectively. For further details, please refer to the original article [94].

2.1 Xu et al's scheme

It comprises four roles and five phases. The roles and their corresponding definitions are listed in Table 1, shown below.

Table 1. The notations used in Xu et al's scheme

Roles	Member	Definition
A	Alice	The voter, who was part of the voting system managed by Bob, wanted Bob to

		blindly sign her candidate selection message.
B	Bob	The agency whose responsibilities include: (1) Managing the committee responsible for authenticating the voter's identity when voter A joins by verifying A's signature. (2) Blindly signing the voter A's candidate choice information.
C	Charlie	The trust inspector verifies the blind signature, where the original message was first blinded by A and then signed by B. The original message corresponds to the ballot containing the candidate name selected by A. After verification, C must reveal A's selection in the ballot message to update the candidate's vote count, thereby reflecting A's choice.
T	Trent	The superintendent monitored the entire system to prevent any counterfeit voting.

There are five phases in their scheme. We have depicted them in Figure 1 and describe each phase below.

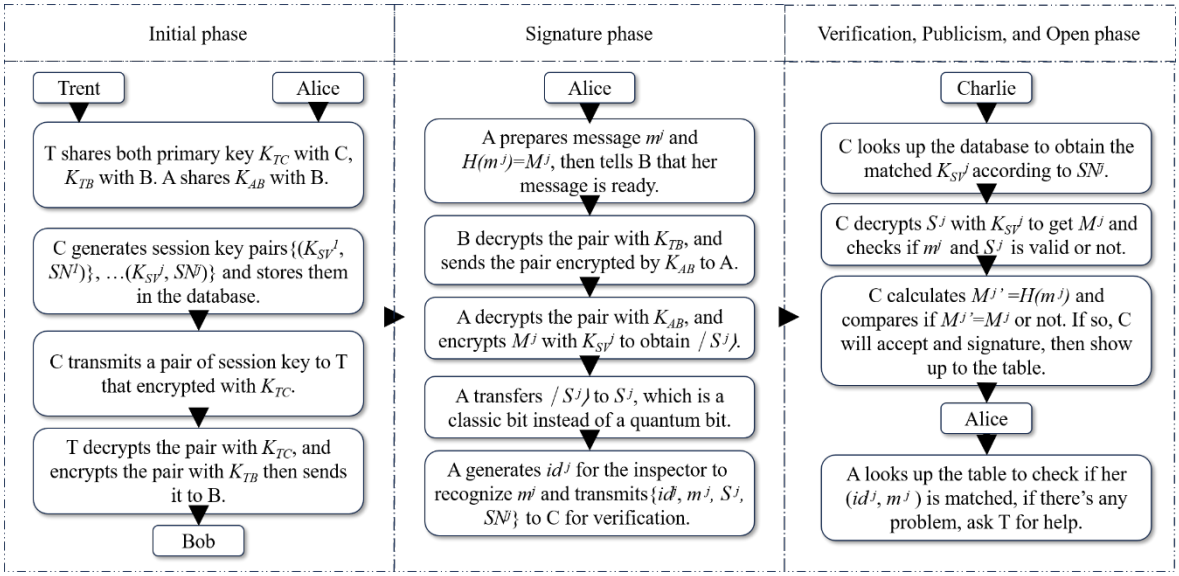

Figure 1. The five phases of Xu et al's scheme

a. Initial phase

In this phase, each pair of roles must share a common secret key prior to the voting process. We describe the process as follows.

(1) T shares a primary key, K_{TC}, with C. Additionally, they share a database where T can store the voter's information and the corresponding shared session keys.

(2) Once B has applied to become a member of the system, T shares a primary key K_{TB} with him after successful identity authentication, enabling B to begin verifying voter A's identity.

(3) Before preparing her candidate selection message, A first shares a primary key K_{AB} with B. Afterward, when she transmits her selection message encrypted with K_{AB} to B, B signs it.

(4) C generates the session key pairs { $(K_{SV}^1, SN^1), \ldots (K_{SV}^N, SN^N)$ } shared with B, where SN^j is the serial number in the *j-th* session (for $j=1$ to N) used to associate the session with the corresponding session key K_{SV}^j. Then, C stores the session key pairs in the session key database.

(5) C uses K_{TC} to encrypt a pair of session keys (K_{SV}^j, SN^j), which becomes $(E_{KTC}(K_{SV}^j), E_{KTC}(SN^j))$, and transmits the ncrypted keys to T.

(6) After receiving the message $(E_{KTC}(K_{SV}^j), E_{KTC}(SN^j))$ from C, T decrypts it using K_{TC}, and then encrypts the resulting pair with K_{TB}, producing $(E_{KTB}(K_{SV}^j), E_{KTB}(SN^j))$. T subsequently sends this to B.

(7) B decrypts $(E_{KTB}(K_{SV}^j), E_{KTB}(SN^j))$ with K_{TB} and obtains (K_{SV}^j, SN^j). The serial number SN^j is used to match each session with its corresponding session key, while the session key K_{SV}^j is used for signing and verification.

b. Signature Phase

In this phase, A prepares message m^j and and computes its hash, denoted as $H(m^j)=M^j$, where m^j represents the candidate's name chosen by A. Then, A informs B that she is ready.

(1) After receiving A's notification, B sends (K_{SV}^j, SN^j) encrypted with K_{AB}, as $(E_{KAB}(K_{SV}^j), E_{KAB}(SN^j))$ to A via a classical channel.

(2) A decrypts $(E_{KAB}(K_{SV}^{j})$ and $E_{KAB}(SN^{j}))$ with K_{AB} to obtain (K_{SV}^{j}, SN^{j}) for signature generation.

(3) A encrypts her hashed message M^{j} using K_{SV}^{j} to obtain the quantum state $|s^{j}\rangle = EK_{SV}^{j}(M^{j})$, and she translates $|s^{j}\rangle$ to s^{j}, which is a string of classical bits instead of the original quantum bits, by using the transformation rules $|0\rangle \rightarrow 00$, $|1\rangle \rightarrow 01$, $|+\rangle \rightarrow 10$, $|-\rangle \rightarrow 11$.

(4) A generates an ID id^{j}, which is an alias used by A to search for the corresponding item in the table, allowing the inspector C to recognize m^{j}.

(5) A transmits $\{id^{j}, m^{j}, s^{j}, SN^{j}\}$ to C for verification via anonymous message communion to ensure that her identification remains confidential.

c. Verification phase

After receiving the anonymous message $\{id^{j}, m^{j}, s^{j}, SN^{j}\}$ from A (C doesn't know the sender's identity), C verifies the anonymous ballot message to determine its validity by using the following steps:

(1) By using SN^{j}, C looks up the database to obtain the match K_{SV}^{j}. If a match is found, C will decrypt $s^{j}(|s^{j}\rangle)$ with K_{SV}^{j}, obtaining M^{j}. Then, C cancels the match.

(2) C uses the received m^{j} to compute $M^{j}=H(m^{j})$ and checks whether this computed M^{j} matches the decrypted value. If they are equal, C accepts s^{j} as a valid signature for M^{j}; otherwise, he rejects the signature.

d. Publication and identification phase

(1) After verifying the signature s^{j} on m^{j}, C displays m^{j} and id^{j} on the display board.

(2) Alice looks up the display board for her identifier id^{j} and the corresponding message m^{j}. If id^{j} exists on the board and the message m^{j} on the board matches her own choice m^{j}, Alice can be confident that her message is accepted without any forgery.

e. Open phase

When a dispute arises, T will receive a signature and its serial number, (s^j, SN^j), from A. Then, T checks the database to determine who signed the signature s^j.

2.2 The weakness of the scheme

Xu et al's scheme is not only conceptually complex but also vulnerable to identity impersonation attacks. Specifically, role A can be impersonated by B during the voting process because all the secrets used by A are also shared with B. This means their scheme is susceptible to a deniability problem if B acts dishonestly, as B possesses the key K_{AB} required for encryption. B can simply sign his own message, which was intended to be blindly signed by himself (B), instead of A's originally chosen message. This leads to a deniability issue. For example, if B signs his own chosen message rather than A's message and then forwards it to C, C cannot detect that the message was actually forged by B. Furthermore, if the message passes C's verification, then according to the rules specified in their scheme, all data legally generated and transferred by B cannot be distinguished from those sent by A. Therefore, in the event of a dispute, although T can be consulted for assistance, if the problem arises from B's forgery, it is impossible for T to detect any irregularity. This is because A cannot prove that the signature was not made by herself, and B can perform the same signing action.

2.3 The requested scenario of a voting environment asserted in [111,112]

The voting environment described in [111, 112] requires voters to go to a designated polling place, where the use of cellphones or any other electronic devices is strictly prohibited to prevent the disclosure of their votes, such as sharing screenshots of their ballots on social media platforms like Facebook or Instagram. In some cases, voters' physical information security is compromised because their ballots may be inspected by higher authorities. Such threats to voter safety must be eliminated.

2.4 The security requirements of a voting system

There are four security requirements for a voting system as outlined in Xu et al.'s scheme, which we present below.

a. Blindness

Trent needs to verify Alice's signature to confirm the sender of the message. Bob can also verify Alice's identity, but he cannot access Alice's voting information. Therefore, the system requires Bob to perform blind signing. This is the concept of blindness.

b. Anonymity

This property ensures that no one can determine the identity of the voter after examining all transmitted parameters. It is a crucial aspect of a voting system.

c. Unforgeability

No one can forge a legal vote because each ballot contains a secret known only to the voter. The legality of the vote can be verified by an authorized verifier, ensuring that the vote cannot be tampered with or forged by others.

d. Undeniability

The voter cannot deny having cast their own ballot. In the event of a dispute, the voter's identity would be disclosed to confirm that the ballot was indeed cast by them.

3. The essential components of the proposed scheme

We designed two essential components of a voting system : (1) quantum signature, and (2) quantum blind signature for the proposed voting system. These original methods are detailed in references [107] and [108], respectively. In this section, we provide an overview of these two components as applied to the proposed quantum voting system (QVS). First, Section 3.1 outlines the roles and definitions within the QVS. Next, Section 3.2 describes the quantum signature scheme [107] used to authenticate voters' identities. Finally, Section 3.3 introduces the adaptation of the quantum blind signature scheme [108].

3.1 The roles and definitions used in our proposed scheme (QVS)

From the discussion in Section 2, we know that Xu et al.'s quantum group blind signature applied in the voting scheme is flawed because the signer and verifier must pre-share a secret key before voting. This requirement renders their scheme deniable.

Furthermore, it cannot be publicly verified due to the inherent properties of the shared key. For these reasons, and inspired by the controversies surrounding election agencies in Taiwan's 2018 election [109,110], this study first adopts the quantum signature scheme [107] for Trent to authenticate voters' identities in the quantum voting system (QVS). We then modify the undeniable quantum blind signature scheme [108] and use it as a foundation to design our quantum voting system. In our design, we adopt the same key generation phase as the basis for QVS, following the approaches of Kaushik et al. and Shi et al. [105,106] quantum signatures. Specifically, we design the QVS by integrating adaptations of our quantum signature scheme and the quantum blind signature scheme [107,108]. These will be described in detail in Section 4. Before that, we present the definitions and roles in QVS in Table 2 and Table 3, respectively.

Table 2 Symbols defined in our scheme

Symbols	Definitions
ID_A, ID_T	The identifies of role A and role T
m_{ij}	The message transmitted between party i and j in the system
$C1$, $C2$	The candidates' name in the system
r_1, r_2, r_a, r_T, r_0, r_C	The random numbers chosen in the system
$H(.)$	A hash function maps arbitrary-size data into a fixed-size output.
$S_j\theta_n$	The private key of each role in the system.
$\|0z\rangle$	The initial state of a quantum state in the Z measurement.
$\|\varphi_{pk}\rangle_i$	The public key of role i in the system.
(r_i, J_i)	The random pair in the random tables set in the initialization.
$\|Sig\rangle_A$, $\|Sig\rangle_T$	The quantum signature signed by roles A and T, respectively.
$\|BSig\rangle_B$	The quantum blind signature signed by B.
W_1, hq, X_1, X_2, Q,	The intermediate parameters produced during the role's calculations

QX_1X_2, W, hw, hrs, hwr, sr, srh, Y, P_1, P_2, Y_B, $H(Y_B)$, $Htot$, K_{PA}, K_{PB}, K_{PC}, K_A, K_B, K_C, M_B, SM_B, $H(Y_A)$, Y_A, $Ex_{CB...}$and other parameters	

Table 3 The roles and their responsibilities in our scheme

Roles	Members	Responsibility
A	Alice	The voter's identity must first be verified by Trent. Only after passing Trent's verification can Alice prepare the candidate selection message to be blindly signed by Bob during the voting process.
B	Bob	B verifies the signature generated by T to validate voter A's identity. The role that B plays is similar to that of a staff member in the government election committee.
C	Charlie	Acts as the ballot opening center to verify the validity of each ballot. If the ballot is valid, C increments the count for the corresponding candidate by one.
T	Trent	A fair third party like the government agency that can authenticate the voter's identity and generate a signature for the voter to be identified as qualified when she votes. In addition, T also can prevent a malicious open counter C from wrong candidate counting.

3.2 The quantum signature scheme [107], which will be used in QVS to authenticate the voter's identity

There are three phases in our quantum signature scheme[107] for T and B to authenticate the voter: (a) public / private key pair generation phase, (b) the signing phase, and (c) the verification phase. We describe each of them below.

a. Public / private key pair generation

We adopt the same key pair generation method, as illustrated in Kaushik et al.'s scheme [105], in our proposed scheme [107]. The system generates a public/private key pair for each member and prepares N-qubit states $|0_z\rangle^{\otimes N}$. The private key $S_j\theta_n$, referred to as S_A, is applied to the state $|0_z\rangle^{\otimes N}$ for member A to generate his/ her public key by rotating an angle $(S_j\theta_n)_A$ from the initial state $|0_z\rangle$. That is, $|\varphi_{pk}\rangle_A = \bigotimes_{j=1}^{N} R(S_A) |0_z\rangle_j$.

b. The signing phase

The signer A selects a random numbers r_A, and prepares her message m. She then computes the related parameters, as shown on the left side of Figure 2, and sends them to the verifier to authenticate her identity.

c. The verification phase

After receiving the parameters from the signer, the verifier performs the steps shown on the right side of Figure 2 to complete the verification.

3.3 The quantum blind signature scheme's [108] modification used in QVS for the voter's candidate selection

To simulate the embedded candidate ballot being blindly signed by the election committee, this section replaces the roles of the signer and verifier in the blind signature scheme [108] with those of the voter and the election committee, respectively. In this context, no third party is required to verify the signature. Therefore, we adapt the original scheme [108] by dividing it into three phases: (1) initial phase, (2) blind signature phase, and (3) verification phase, as illustrated in Figure 3.

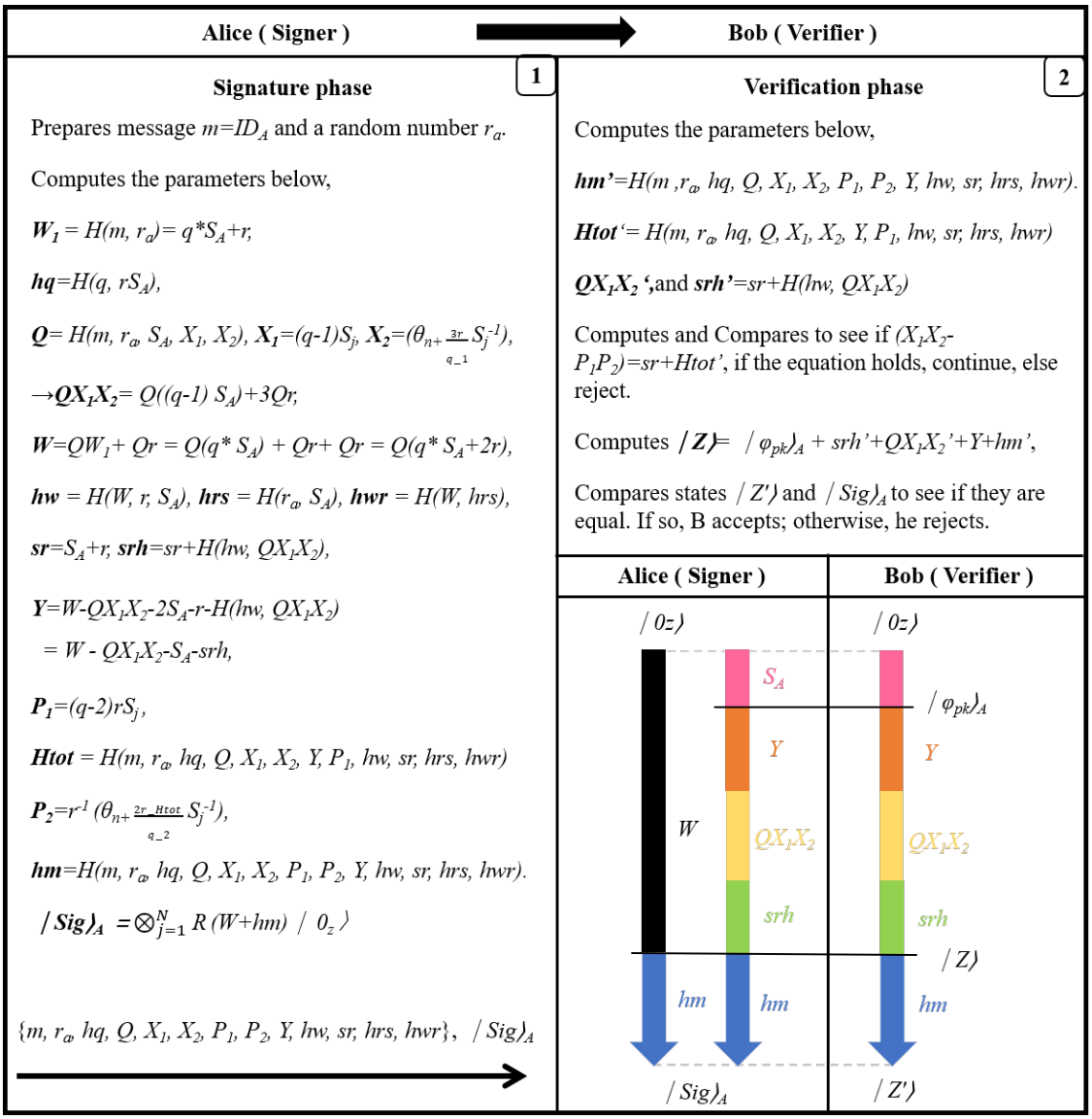

Figure 2 The quantum signature scheme[107] and its schematic diagram shown in the lower right corner

.

a. The verifier B's initial phase

The verifier Bob (B) first selects a random number r_1, prepares a message m, and computes $M_B=r_1+H(m)$, $Sh_B= (M_B, S_B)$, and $SM_B=M_B+sh_B$. He then transmits SM_B and sh_B to the signer Alice for verification and blind signing of M_B.

b. The signer Alice's blind signing phase

After receiving the blind messages SM_B and sh_B transmitted by the verifier B, the signer Alice (A) performs the steps to do the blind signature phase, as shown in Figure 3. A computes $\boldsymbol{W_1}=H(M_B, r_2)=q*S_A+r$, $\boldsymbol{X_1}=(q-2)* M_B *S_A$, $\boldsymbol{X_2}=(1+r(q-2)^{-1} S_A^{-1})$, $\boldsymbol{Q}=H(M_B, S_A, X_1, X_2)$, $\boldsymbol{QX_1 X_2} =QM_A*(q-2)*S_A+r$, $\boldsymbol{W}=(QW_1+2Qr) M_B +S_A=Q(qS_A+3r)M_B+S_A$, $\boldsymbol{Y_A}=W- QX_1 X_2 - S_A$, and generates $\boldsymbol{|Z\rangle_A} = \bigotimes_{j=1}^{N} R(W) |\varphi_{pk}\rangle_B$. After that, she computes $H(Y_A)$, $\boldsymbol{a}= H(Y_A)-Y_A$, $\boldsymbol{P_A}= H(sh_B, H(M_B, S_A, Y_A, a, sh_B), M_B, H(Y_A), a)$, and $\boldsymbol{P}=P_A-QX_1X_2+a+M_B$, and finally generates the blind signature $|BSig\rangle_A =\bigotimes_{j=1}^{N} R^{(j)}(P) |Z\rangle_A$.

Then, he sends all the parameters back to the verifier B.

c. The verifier B's verification phase

As the figure shows, the verifier B calculates $sh_B'=SM_B-M_B$, $\boldsymbol{P_A}'=H(sh_B', H(M_B, S_A, Y_A, a, sh_B), M_B, H(Y_A), a)$ and compares if $H(P_A')= H(P_A)$. If so, he then computes $\boldsymbol{P_B} =H(Y_A)+P_A'+ S_B + M_B$, finally generates $\boldsymbol{|Z'\rangle_B}= \bigotimes_{j=1}^{N} R(P_B) |\varphi_{pk}\rangle_A$, and compares to see whether $|BSig\rangle_A = |Z'\rangle_B$ holds or not. If it holds, B accepts.

4. The Proposed quantum voting system (QVS)

In this section, by referencing the voting environment in Taiwan's voting system (TVS), we adapt both our quantum signature and quantum blind signature schemes to design an online quantum voting system architecture. This system is expected to be applicable globally for online voting in the near future.

In Taiwan, prior to an election, government agencies screen eligible citizens and send them both voting notices and candidate information. On election day, voters must bring their notice documents to the polling station and deposit them into the voting cabinet promptly. At this stage, election officers first verify the legality of the voter's ballot and confirm their identity. Voters cast their ballots in a private booth enclosed by a curtain. After marking their choice, they fold the ballot and place it into the ballot box. During the voting process, voters are prohibited from discussing the election with others or revealing the content of their ballots or any related information. Even the identity verification officer, who is present in the same space, cannot access any information about a voter's ballot. After voting, it is impossible to trace any ballot back to the individual voter, thereby ensuring the anonymity of the entire electoral system.

Our QVS simulates the above scenario, except that it operates online. It consists of two main components: (1) quantum signature and (2) quantum blind signature.

Below, we denote the quantum signature produced by role i as $|Sig\rangle_i$, and the quantum blind signature generated by B as $|BSig\rangle_B$. We first show the relationships among all the roles in Taiwan voting process in Figure 4. Then, using the required signatures $|Sig\rangle_i$, $|BSig\rangle_B$, and all participating roles in the voting system, we present the voting process in QVS in Figure 5. Meanwhile, we describe how QVS operates from Section 4.1 through Section 4.4 in this section. They each correspondingly define the followings in QVS : (1)the registration phase; (2) $|Sig\rangle_A$ is a certification for A's citizenship ownership; (3) $|Sig\rangle_T$ as a legal voting notice; and (4) $|BSig\rangle_B$ as the blind voting information, which is verified by the ballot opener C, with its correctness randomly examined by the government agency T.

4.1 The registration phase in QVS.

During the registration stage, the system generates private keys, public keys and memory random tables for all QVS members, including voter A, election committee B, ballot opener C, and government agency T. The private key $S_j\theta_n$ is known only to the registrant, while the public key, which is publicly accessible, is generated by rotating the angle of $S_j\theta_n$ on the zero state $|0\rangle_z$.

Regarding the random table of A, the pair (r_{Ai}^j, J_{Ai}^j) represents the random values associated with voter Ai during her j^{th} voting instance, known only to Ai and T. Similarly, the pairs (r_{BAi}^j, J_{BAi}^j) and (r_{CAi}^j, J_{CAi}^j) in the memory tables of B and C indicate that staff members B and C hold distinct values for voter A$i's$ jth vote. Additionally, the government agency T maintains versions of these tables corresponding to those owned by A, B, and C within its database. We present the parameters related to their roles in T's memory using Tables 4 through 6, and those in the memories of A, B, and C using Tables 7 through 9, respectively. Furthermore, all voters A share the same random value K_{CT}^j, which is shared with C and T and used during their jth voting instance. Staff members B, C, and T share random values K_{BCT}^j for the system j^{th} voting activity. We assume there are n voters and m voting activities, where $i \in \{1, 2, ..., n\}$, $j \in \{1, 2, ..., m\}$. Moreover, T shares rc^j with C, which is used in the computation of K_V for both blurring and defuzzifying parameters D and r_m; K_{CT}^j.

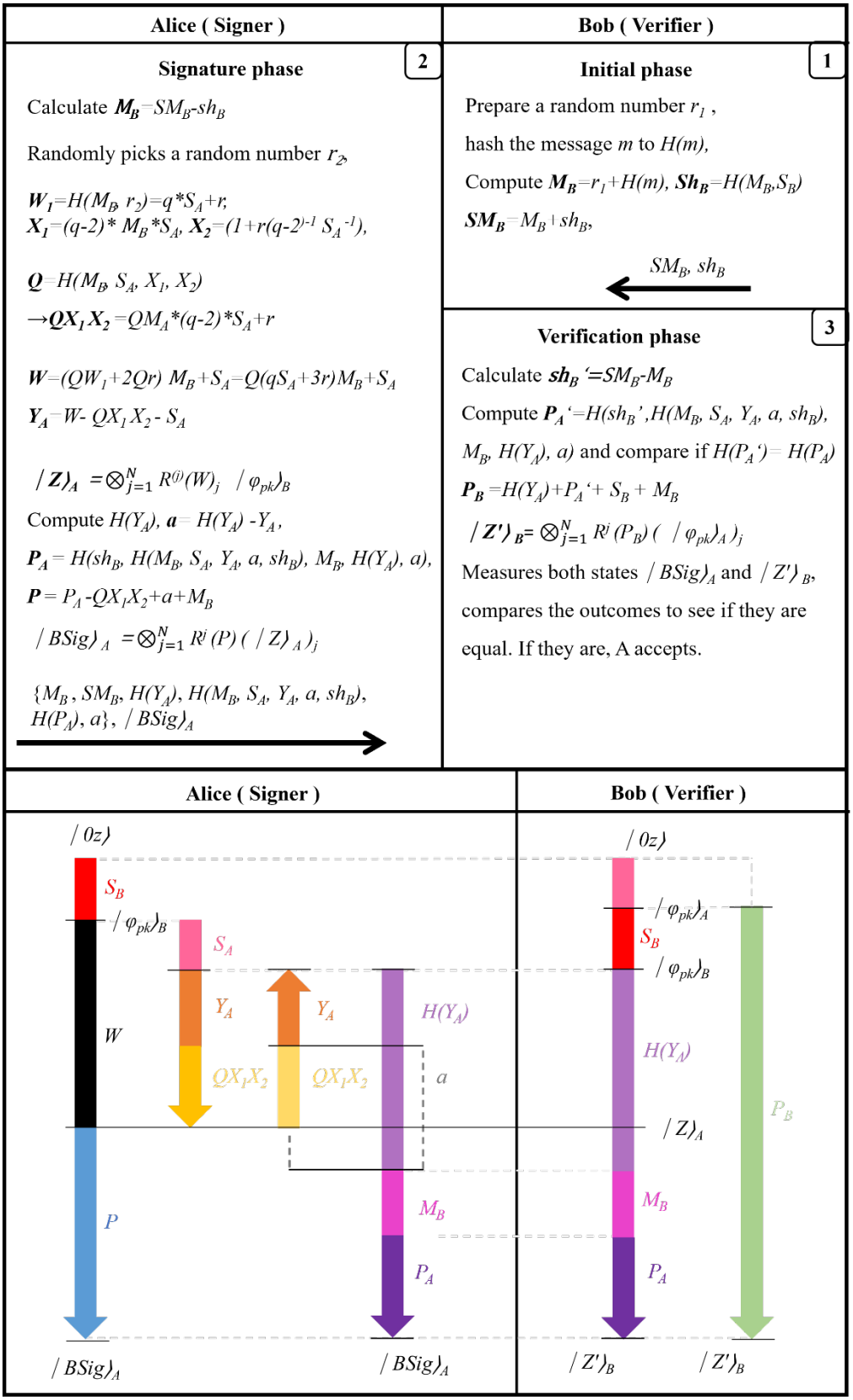

Figure 3 The modified quantum blind signature from scheme[108] and its schematic diagram shown in the lower part

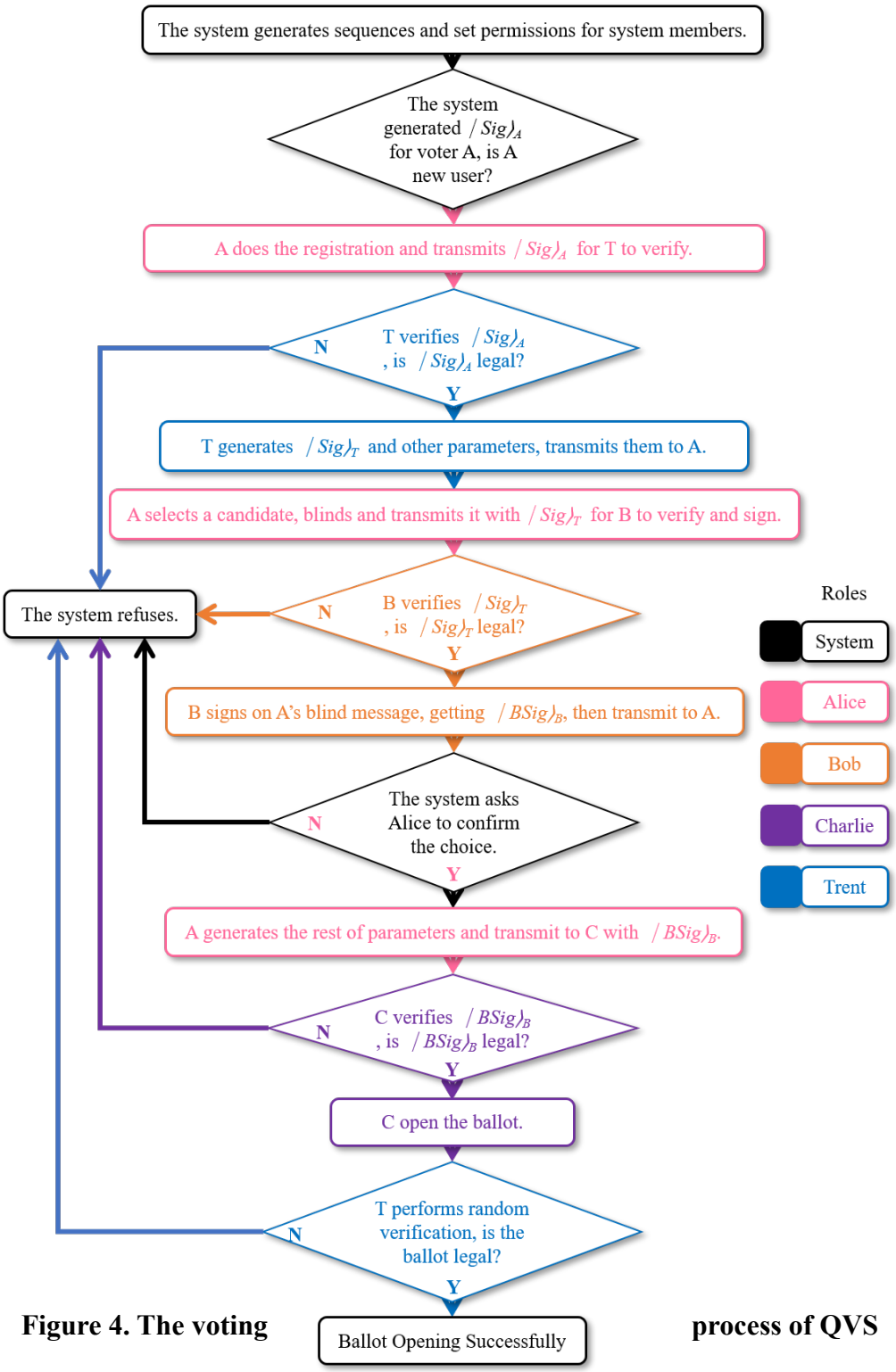

Figure 4. The voting process of QVS

with C and all voters; $VP=H(r_T, VCS_T)$, which is shared with all voters for T to verify voting correctness; and O_{CT}^j with C for the system's j^{th} voting opening, respectively.

QVS has two stages: (A) system-off and (B) system-on. Stage (A) is described in Section 4.2, while Stage (B) will be detailed d after Section 4.3.

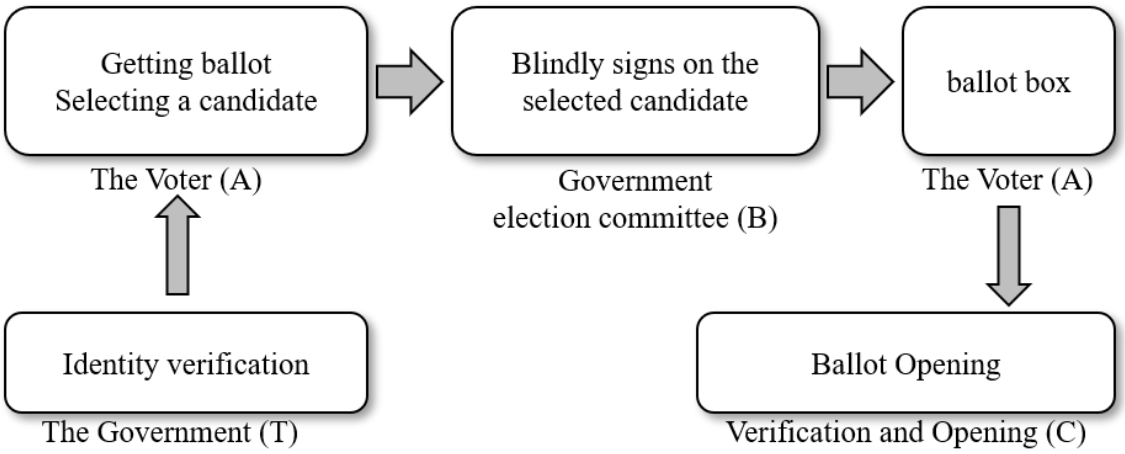

Figure 5 The voting process in Taiwan

Table 4 T's random table for each voter A*i* in her j^{th} times voting.

	V1	V2	V3	…	V*m*
A1	r_{A1}^{1}, J_{A1}^{1}	r_{A1}^{2}, J_{A1}^{2}	r_{A1}^{3}, J_{A1}^{3}	…	r_{A1}^{m}, J_{A1}^{m}
A2	r_{A2}^{1}, J_{A2}^{1}	r_{A2}^{2}, J_{A2}^{2}	r_{A2}^{3}, J_{A2}^{3}	…	r_{A2}^{m}, J_{A2}^{m}
A3	r_{A3}^{1}, J_{A3}^{1}	r_{A3}^{2}, J_{A3}^{2}	r_{A3}^{3}, J_{A3}^{3}	…	r_{A3}^{m}, J_{A3}^{m}
…	…	…	…	…	…
A*n*	r_{An}^{1}, J_{An}^{1}	r_{An}^{2}, J_{An}^{2}	r_{An}^{3}, J_{An}^{3}	…	r_{An}^{m}, J_{An}^{m}

Table 5 T's random table for B with each A*i* in j^{th} voting times

	V1	V2	V3	…	V*m*
A1	r_{BA1}^{1}, J_{BA1}^{1}	r_{BA1}^{2}, J_{BA1}^{2}	r_{BA1}^{3}, J_{BA1}^{3}	...	r_{BA1}^{m}, J_{BA1}^{m}
A2	r_{BA2}^{1}, J_{BA2}^{1}	r_{BA2}^{2}, J_{BA2}^{2}	r_{BA2}^{3}, J_{BA2}^{3}	...	r_{BA2}^{m}, J_{BA2}^{m}

A3	r_{BA3}^{1}, J_{BA3}^{1}	r_{BA3}^{2}, J_{BA3}^{2}	r_{BA3}^{3}, J_{BA3}^{3}	…	r_{BA3}^{m}, J_{BA3}^{m}
…	…	…	…	…	…
A*n*	r_{BAn}^{1}, J_{BAn}^{1}	r_{BAn}^{2}, J_{BAn}^{2}	r_{BAn}^{3}, J_{BAn}^{3}	…	r_{BAn}^{m}, J_{BAn}^{m}

Table 6 T's random table for C with each A*i* in j^{th} voting times

	V1	V2	V3	…	V*m*
A1	r_{CA1}^{1}, J_{CA1}^{1}	r_{CA1}^{2}, J_{CA1}^{2}	r_{CA1}^{3}, J_{CA1}^{3}	…	r_{CA1}^{m}, J_{CA1}^{m}
A2	r_{CA2}^{1}, J_{CA2}^{1}	r_{CA2}^{2}, J_{CA2}^{2}	r_{CA2}^{3}, J_{CA2}^{3}	…	r_{CA2}^{m}, J_{CA2}^{m}
A3	r_{BA3}^{1}, J_{CA3}^{1}	r_{CA3}^{2}, J_{CA3}^{2}	r_{CA3}^{3}, J_{CA3}^{3}	…	r_{CA3}^{m}, J_{CA3}^{m}
…	…	…	…	…	…
A*n*	r_{CAn}^{1}, J_{CAn}^{1}	r_{CAn}^{2}, J_{CAn}^{2}	r_{CAn}^{3}, J_{CAn}^{3}	…	r_{CAn}^{m}, J_{CAn}^{m}

Table 7 A's random table for her j^{th} voting

	V1	V2	V3	…	V*m*
A*n*	r_{An}^{1}, J_{An}^{1}	r_{An}^{2}, J_{An}^{2}	r_{An}^{3}, J_{An}^{3}	…	r_{An}^{m}, J_{An}^{m}

Table 8 B's random table for the j^{th} voting of each A

	V*m*
A1	r_{BA1}^{m}, J_{BA1}^{m}
A2	r_{BA3}^{m}, J_{BA3}^{m}
…	…
A*n*	r_{BAn}^{m}, J_{BAn}^{m}

4.2 $|Sig\rangle_A$ between Alice and Trent for Trent to identify Alice's citizenship

$|Sig\rangle_A$ represents the ownership of citizenship transferred from A to T, indicating that A's legal status has been confirmed by the government agency T.

(A) The QVS System—Off Stage

T (Trent) must first authenticate the identity of A (Alice), who has transmitted her signature $|Sig\rangle_A$ to him. If A is verified as legitimate, T will generate and send a signature $|Sig\rangle_T$ as a voting notice to her. This notice will then be forwarded from A to B, allowing B to verify the validity of A's voting right. This process is illustrated in Figure 6. For the generation of $|Sig\rangle_A$ (Alice's signature verified by Trent), the QVS imports the list of voters from the Household Registration Office during the registration stage. It enables voter A to create $|Sig\rangle_A$ by modifying the scheme described in Section 3.2 as follows.

QVS sets the number j as the number of iterations for this voting process. It assigns the value i to Ai to use the (r_{Ai}^j, J_{Ai}^j) pair during the voting process, and allows T to look up the random table to find the (r_{Ai}^j, J_{Ai}^j) pair when Ai votes.

(1) **A's side**

After checking her own random table and obtaining r_{Ai}^j and J_{Ai}^j, A can successfully generate $Kp_A=H(r_{Ai}^j, (r_{Ai}^j, J_{Ai}^j))$. Then, A prepares $m_A=(ID_A, i)$ and generates the signature $|Sig\rangle_A = \bigotimes_{j=1}^{N} R(W+hm)_j |\varphi_{pk}\rangle_T$ using T's public key, following the steps shown in Figure 6. She then transmits m_{AT} (the signature $|Sig\rangle_A$ and the related parameters) to T for verification.

(2) T's side

After receiving $\boldsymbol{m_{AT}}$ from A, T uses i in m_A to calculate K_A after checking the random table, then follows the steps shown in step 2 of Figure 6 to verify $|Sig\rangle_A$.

We demonstrate the correctness of T's $|Sig\rangle_A$ verification as follows.

Proof

The angle of $|Sig\rangle_A$ from $|0\rangle_Z$ is $R(W+hm)$ $|\varphi_{pk}\rangle_T$, and the angle of $|Z\rangle$ from $|0\rangle_Z$ is $|\varphi_{pk}\rangle_T + S_T + srh' + QX_1X_2 + Y + hm$; therefore, we can see that $W = S_A + srh + QX_1X_2 + Y$. This completes the proof.

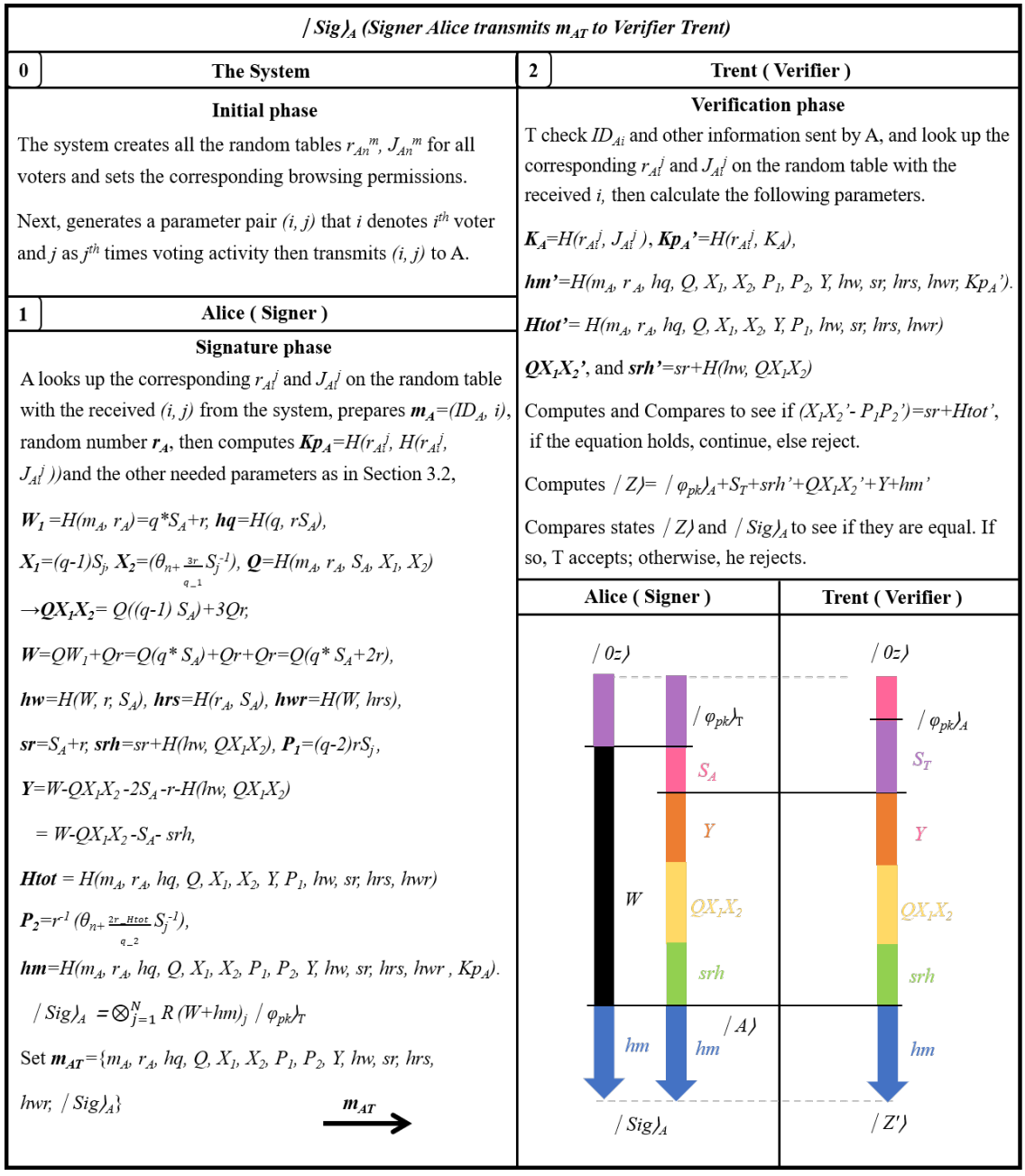

Figure 6. A generates $|Sig\rangle_A$ and T verifies it to ensure that A has the citizenship with T's verification schematic diagram shown in the lower right part

If A is legal, T will generate a signature $/Sig\rangle_T$ as a voting notice for her, which will then be transmitted from A to B for the latter to verify the validity of A's voting right. If $/Sig\rangle_T$ is valid, A is recognized as having the voting right. The system will enter the system-on stage only after all the notices $/Sig\rangle_T$ have been sent to the voters and the election day arrives. T also instructs C to reset the candidates' counts to zero. That is, C sets $Cicnt=0$, $i=1$ to (number of candidates) to count the number of ballots for each candidate when the system-on stage begins. We will discuss $/Sig\rangle_T$ and the relevant steps in Section 4.3 and illustrate them in Figure 7.

4.3 The voting notice $/Sig\rangle_T$ among Trent, Alice and Bob

$/Sig\rangle_T$ represents a voting notice sent from T to A, and subsequently from A to B, allowing B to verify that A's legality has been confirmed by the government agency T.

$/Sig\rangle_T$(Signer Trent transmits m_{TA} to Alice, and Alice sends m_{TB} to Bob)

3 Trent (Signer)

Initial phase

T selects a random number r_T, looks up the corresponding r_{BAi}^j , J_{BAi}^j , r_{CAi}^j, J_{CAi}^j on the random table and calculates

$\boldsymbol{K_B}= H(r_{BAi}^j, J_{BAi}^j)$, $\boldsymbol{Kp_B}=H(r_{BAi}^j, K_B)$,

$\boldsymbol{K_C}= H(r_{CAi}^j, J_{CAi}^j)$, $\boldsymbol{Kp_C}=H(r_{CAi}^j, K_C)+r_{Ai}^j$, $\boldsymbol{Kp_{BC}}=J_{BAi}^j+Kp_C$,

$\boldsymbol{Kp_{CA}}=J_{Ai}^j+Kp_C$, $\boldsymbol{Ex_{CBA}}=J_{Ai}^j+K_B+K_{CT}^j$, $\boldsymbol{VP}=H(r_T, VCS_T)$,

$\boldsymbol{K_V}=H(H(r_T r_{Ai}^j)), H(r_T, r_C^j), VP)$,

$\boldsymbol{K_{VA}}=K_V+H(r_T, r_C^j)$, $\boldsymbol{K_{VC1}}= K_V+H(r_T, r_C^j)+H(r_T, r_{Ai}^j)$

Sets $\boldsymbol{C_iCnt}=0$, $i=1$ to number of candidates

Signature phase

T makes the message $m_T=(ID_A||ID_T, i)$ and computes the relevant parameters as the first part shown in Section 3.2 for T,

$\boldsymbol{W_1}=H(m_T, r_T)=q*S_T+r$, $\boldsymbol{hq}=H(q, rS_T)$,

$\boldsymbol{X_1}=(q-1)S_j$, $\boldsymbol{X_2}=(\theta_{n+\frac{3r}{q-1}} S_j^{-1})$, $\boldsymbol{Q}=H(m_T, r_T, S_T, X_1, X_2)$

$\rightarrow \boldsymbol{QX_1X_2}= Q((q-1) S_T)+3Qr$,

$\boldsymbol{W}=QW_1+Qr=Q(q* S_T)+Qr+Qr=Q(q* S_T+2r)$,

$\boldsymbol{hw}=H(W, r, S_T)$, $\boldsymbol{hrs}=H(r_T, S_T)$, $\boldsymbol{hwr}=H(W, hrs)$,

$\boldsymbol{sr}=S_T+r$, $\boldsymbol{srh}=sr+H(hw, QX_1X_2)$, $\boldsymbol{P_1}=(q-2)rS_j$,

$\boldsymbol{Y}=W-QX_1X_2-2S_T-r-H(hw, QX_1X_2)$
$= W-QX_1X_2-S_T- srh$,

$\boldsymbol{r_{Kp}}=H(r_{Ai}^j, Kp_C)$, $\boldsymbol{m_{T1}}=m_T+r_{Kp}$, $\boldsymbol{r_{T1}}=r_T+r_{Kp}$,

$\boldsymbol{Htot}$ = $H(m_{T1}, r_{T1}, hq, Q, X_1, X_2, Y, P_1, hw, sr, hrs, hwr)$

$\boldsymbol{P_2}=r^{-1} (\theta_{n+\frac{2r \cdot Htot}{q-2}} S_j^{-1})$,

$\boldsymbol{hm}=H(m_T, r_T, hq, Q, X_1, X_2, P_1, P_2, Y, hw, sr, hrs, hwr , K_A , Kp_B)$.

$/Sig\rangle_T = \otimes_{j=1}^{N} R(W+hm) \ /\varphi_{pk}\rangle_B$

Set $\boldsymbol{m_{TA}}=\{m_{T1}, r_{T1}, hq, Q, X_1, X_2, P_1, P_2, Y, hw, sr, hrs,$

$hwr, /Sig\rangle_T, Kp_{BC}, K_{VA}, K_{VC1}, Ex_{CBA}, Kp_{CA}\}$ $\xrightarrow{m_{TA}}$

4 *QVS system-on stage* Alice (Voter)

Prepares a random number r_m, r_0, makes selection of the candidate $C1$, hashes it to $H(C1)$, then calculates $\boldsymbol{Ex_{CB}}=Ex_{CBA}-J_{Ai}^j$, $\boldsymbol{K_B}=Ex_{CB}-K_{CT}^j$, $\boldsymbol{K_V}=K_{VA}-H(r_T, r_{Ai}^j)$,

$\boldsymbol{K_{AB}}=H(r_{Ai}^j, J_{Ai}^j)+K_B$, $\boldsymbol{D}=H(r_m, S_A)$, $\boldsymbol{D^*}=D+K_V$, $\boldsymbol{r_m}*=r_m+K_V$, $\boldsymbol{r_{Kp}}=H(r_{Ai}^j, Kp_C)$,

$\boldsymbol{m_T}=m_{T1}-r_{Kp}$, $\boldsymbol{r_T}=r_{T1}-r_{Kp}$, $\boldsymbol{Kp_C}= Kp_{CA}-J_{Ai}^j (= H(r_{CAi}^j, K_C)+r_{Ai}^j)$, $K_{VC}=K_{VC1}-H(r_T, r_{Ai}^j)$

$\boldsymbol{VP}=H(r_T, VCS_T)$, $\boldsymbol{V_1}=H(Kp_C, D^*, K_{CT}^j, r_0, r_m*, D, r_m, H(C1))$,

$\boldsymbol{V_C}=H(V_1, H(D^*, K_{CT}^j, r_0, r_m*, H(C1), VP))$, $\boldsymbol{V}= H(V_1, V_C)$,

$\boldsymbol{SM_A}=H(C1)+D+r_m+H(r_m)+Ex_{CB}-H(r_0, V)-Kp_C$,

calculate $\boldsymbol{m_{TA}}= m_{TA}-\{K_{VC1}, K_{VA}\}$, set $\boldsymbol{m_{AB}}= \{i, SM_A, K_{AB}, m_{TA}\}$

5 Bob (Verifier) ***Pre-voting*** $\boldsymbol{m_{AB}}$ ↓

Verification phase

Looks up the corresponding r_{BAi}^j and J_{BAi}^j with received i, then calculates

$\boldsymbol{K_B'}=H(r_{BAi}^j, J_{BAi}^j)$, $\boldsymbol{K_A'}=K_{AB}-K_B'(=H(r_{Ai}^j, J_{Ai}^j))$, $\boldsymbol{Kp_B'}=H(r_{BAi}, K_B')$, $\boldsymbol{Htot'}$ = $H(m_{T1}, r_{T1}, hq, Q, X_1, X_2, Y, P_1, hw, sr, hrs, hwr)$, $\boldsymbol{QX_1X_2'}$, $\boldsymbol{srh'}$ $=sr+H(hw, QX_1X_2)$, and $\boldsymbol{hm'}=H(m_T, r_T, hq, Q, X_1, X_2, P_1, P_2, Y, hw, sr, hrs, hwr, K_A', Kp_B')$

Computes and Compares to see if $(X_1X_2'- P_1P_2')=sr+Htot'$, if the equation holds, continue, else rejects.

Computes $/Z\rangle= /\varphi_{pk}\rangle_T+S_B+srh'+QX_1X_2'+Y+hm'$, and compares to $/Sig\rangle_T$ and $/Z\rangle$ see if they are equal. If so, B accepts; otherwise, he rejects.

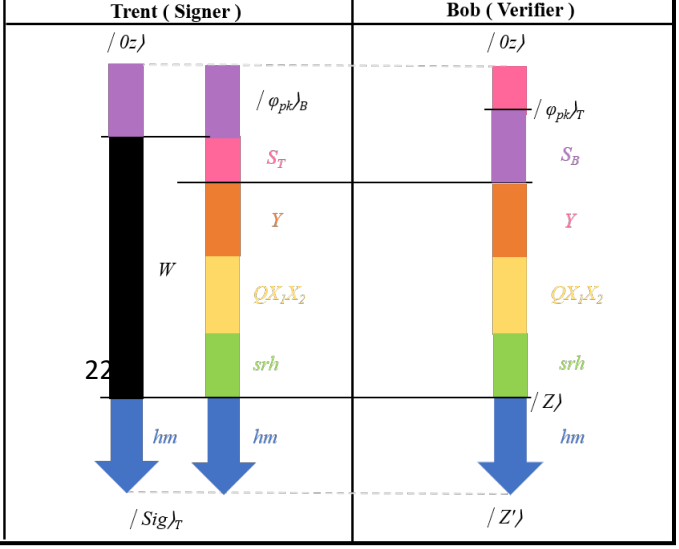

Figure 7. T generates a legal voting notice $/Sig\rangle_T$ for A, and B verifies it when A votes. The verification schematic diagram is shown in the lower right section.

(1) T's side

After T has verified $/Sig\rangle_A$, A*i* will receive T's signature $/Sig\rangle_T$, which is produced by T and used by A*i*, to prove to B that A*i* is a legitimate voter, as described below.

T checks the memory random table after successful verification of $/Sig\rangle_A$. He generates $K_B=H(r_{BAi}^j, J_{BAi}^j)$, $Kp_B=H(r_{BAi}^j, K_B)$, $K_C=H(r_{CAi}^j, J_{CAi}^j)$, and $Kp_C=H(r_{CAi}^j, K_C)+r_{Ai}^j$ after obtaining the corresponding pairs (r_{BAi}^j, J_{BAi}^j) and (r_{CAi}^j, J_{CAi}^j). Additionally, T calculates the following parameters $Ex_{CBA}=J_{Ai}^j+K_B-K_{CT}^j$, $Kp_{BC}=J_{BAi}^j+Kp_C$, $Kp_{CA}=J_{Ai}^j+Kp_C$, $VP=H(r_T, VCS_T)$, $K_V=H(H(r_T, r_{Ai}^j), H(r_T, r_C^j), VP)$, $K_{VA}=K_V+H(r_T, r_{Ai}^j)$, and $K_{VC1}=K_V+H(r_T, r_C^j)+H(r_T, r_{Ai}^j)$. Then, T initializes $CiCnt=0$ for $i=1$ to the number of candidates. In the signature phase, T prepares $m_T=(ID_A||ID_T, i)$, computes $r_{Kp}=H(r_{Ai}^j, Kp_C)$, $m_{T1}=m_T+r_{Kp}$, and $r_{T1}=r_T+r_K$. He generates the signature $/Sig\rangle_T$ following the procedure described in Section 3.3. T then sends $m_{TA}=\{m_{T1}, r_{T1}, hq, Q, X_1, X_2, P_1, P_2, Y, hw, sr, hrs, hwr, /Sig\rangle_T, Kp_{BC}, K_{VA}, K_{VC1}, Ex_{CBA,}, Kp_{CA}\}$ to A, enabling B to verify A's legitimacy when voting, as illustrated in Step 3 of Figure 7. After all voters have been successfully verified and the election day arrives, the system enters the system-on phase. At this point, T has sent $/Sig\rangle_T$ to all legal voters, and C has set $CiCnt=0$, for $i=1$ to number of candidates.

(B) The QVS system-on stage

(2) A's side

After receiving $/Sig\rangle_T$ in m_{TA} from T, A selects two random numbers r_m and r_0, and prepares her candidate choice, $H(C1)$. To construct part of her ballot, she computes the following: $Ex_{CB}=Ex_{CBA}-J_{Ai}^j$, $K_B=Ex_{CB}+K_{CT}^j$, $K_{AB}=H(r_{Ai}^j, J_{Ai}^j)+K_B$, $r_{Kp}=H(r_{Ai}^j, Kp_C)$, $m_T=m_{T1}-r_{Kp}$, $r_T=r_{T1}-r_{Kp}$, $VP=H(r_T, VCS_T)$, $K_V=K_{VA}-H(r_T, r_{Ai}^j)$, $D=H(r_m, S_A)$, $D^*=D+K_V$, $r_m^*=r_m+K_V$, $Kp_C=Kp_{CA}-J_{Ai}^j(=H(r_{CAi}^j, K_C)+r_{Ai}^j)$, $V_1=H(Kp_C, D^*, K_{CT}^j, r_0, r_m^*, D, r_m, H(C1))$, $V_C=H(V_1, H(D^*, K_{CT}^j, r_0, r_m^*, H(C1), VP))$, and $V=H(V_1, V_C)$. Finally, he generates the ballot

$SM_A=H(C1)+D+r_m+H(r_m)+Ex_{CB}-H(r_0, V)-Kp_C$……***equation (1)***

Then, he sets and sends $m_{TA}=m_{TA}-\{K_{VC1}, K_{VA}\}$, $m_{AB}=\{i, SM_A, K_{AB}, m_{TA}\}$ to B, as shown in step 4 of Figure 7.

(3) B's side

After receiving m_{AB} from A, B first looks up J_{BAi}^{j} using r_{BAi}^{j} and calculates $K_B=H(r_{BAi}^{j}, J_{BAi}^{j})$ and $Kp_B=H(r_{BAi}^{j}, K_B)$. Then, B computes $K_A'=K_{AB}-K_B$, allowing him to verify $\lfloor Sig \rfloor_T$, with K_A and his own K_B, as shown in Figure 7.

We prove the correctness of B's $\lfloor Sig \rfloor_T$ verification as follows.

Proof

$\lfloor Sig \rfloor_T = \bigotimes_{j=1}^{N} R(W+hm) \lfloor \varphi_{pk} \rfloor_B$, where $W=Y+QX_1X_2+2S_T+r+H(hw, QX_1X_2)$, and $\lfloor Z \rfloor = \lfloor \varphi_{pk} \rfloor_T +S_B+srh'+QX_1X_2'+Y+hm'$, where $srh'=sr+H(hw, QX_1X_2)$, and $sr=S_T+r$.

Therefore, the correctness of the verification is proven.

4.4 The Blind Voting Information Signature $\lfloor BSig \rfloor_B$ Among Bob, Alice, and Charlie

After B successfully verifies $\lfloor Sig \rfloor_T$ received from A, B, A, C, and T will collaboratively perform the following steps to complete A's voting process, as illustrated in Figure 8. These steps are described below.

(1) B's side ($\lfloor BSig \rfloor_B$ generation)

If $\lfloor Sig \rfloor_T$ from A to B passes B's verification, B computes $Kp_C=(Kp_{BC}-J_{BAi}^{j})=(r_{Ai}^{j}+Kp_{CT})$ and subtracts K_B from SM_A to obtain $M_A=SM_A-K_B$ $(=H(C1)+D+r_m+H(r_m)-K_{CT}^{j}-H(r_0, V)-Kp_C)$, which contains the secret candidate name $H(C1)$ chosen by A. Then, B blindly signs M_A to obtain $\lfloor BSig \rfloor_B$, as shown in step 6 of Figure 8. After that, B sets $m_{BA}=\{H(M_A, S_B, Y_B, a), H(Y_B), H(P_B), \lfloor BSig \rfloor_B\}$ and sends it to A.

(2) A's side (casts the ballot into the box)

After receiving m_{BA}, which includes the blind signature and several parameters from B, A first computes the following: $K_{VC}=K_{VC1}-H(r_T, r_{Ai}^{j})=K_V+H(r_T, r_C^{j})$, $MK_P=M_A+Kp_C$,

$MK=MK_P+H(r_0, V)+K_{CT}^j-K_V$ $(=H(C1)+D+r_m+H(r_m)-K_V)$, $Kcz=H(r_m, H(C1), D, V, r_0)$. These computations correspond to step 7 in Figure 8. Next, A sets m_{AC}= {MK, r_T, r_m*, D*, Kcz, r_0, K_{VC}, V, V_1, V_C, m_{BA}} and transmits it to C for C's ballot opening, as well as for T's storage and random verification.

(3) C's turn

After receiving m_{AC} from A, C first verifies $\langle BSig \rangle_B$ by calculating $K_V=K_{VC}-H(r_T, rc^j)$, $MK_P(=M_A+Kp_C)=MK-H(r_0, V)-K_{CT}^j+K_V$, $P_B'=H(H(M_A, S_B, Y_B, a), H(M_A+K_{PC}), H(Y_B))$. C then compares whether $H(P_B')$ equals $H(P_B)$. If they match, he calculates $P_C=S_C+H(Y_B)+P_B'+MK_P$. Finally, C measures and compares the outcomes of the states Measures and compares the outcomes of the states $\langle Z' \rangle_B (=\bigotimes_{j=1}^{N} R\ (P_C)_j \langle \varphi_{pk} \rangle_B)$ and $\langle BSig \rangle_B$ to check if they are equal, as shown in step 8 of Figure 8.

We prove the correctness of C's $\langle BSig \rangle_B$ verification as follows.

Proof

$\langle BSig \rangle_B = \bigotimes_{j=1}^{N} R(P+W)(=P_B-QX_1X_2+a+M_A+Kp_C+Y_B+QX_1X_2+S_B \langle \varphi_{pk} \rangle_C$, $\langle Z' \rangle_B = \bigotimes_{j=1}^{N} R\ (P_C)(=S_C+H(Y_B)+P_B'+M_A+Kp_C) \langle \varphi_{pk} \rangle_B$.

Therefore, we know that $\langle BSig \rangle_B = \langle Z' \rangle_B$ holds.

If valid, C opens the ballot by calculating $r_m=r_m*-K_V$, $D=D*-K_V$, $H(C1)=MK_P-D-r_m-H(r_m)+K_{CT}^j+H(r_0, V)$, which indicates that the voter selected candidate $C1$. Next, C calculates $Kcz'=H(r_m, H(C1), D, V, r_0))$, and compares whether Kcz' equals Kcz. If so, C increments $C1Cnt$ by one. Then, he calculates $OV=H(O_{CT}^j, r_m*)+H(C1)$, $rr=H(D*, V, r_m*, Kcz\ , H(C1))$, $C1Cnt_C=C1Cnt+rr$. Finally, C sets m_{CT}={$D*$, $rm*$, r_T, $C1Cnt_C$, Kcz, V, V_1, V_C, r_0, OV, MK_P }, and sends it to T for storage and random verification.

(4) T's turn (Storing and random verification)

After receiving m_{CT} from C, T first opens the ballot by computing $H(C1)=OV- H(O_{CT}^j, r_m*)$, $VP'=H(r_T, VCS_T)$, then computes $V_C'=H(V_1, H(D*, K_{CT}^j, r_0, r_m*, H(C1), VP'))$, and $V'=H(V_1, V_C')$. T then compares V' with V to determine if they are equal. If they match, T verifies whether $C1Cnt$ has been increased by 1 by computing $rr=H(D*, V,$

r_m*, *Kcz* , *H(C1))*, *C1Cnt=C1Cnt*$_C$*-rr*. If *C1Cnt* had increased by exactly 1, T confirms that the count of C1 is correct, as shown in step 10 of Figure 8.

5. Security analyses

In our security analysis, we categorized our discussion into three types of signatures: two regular signatures $/Sig\rangle_A$ and $/Sig\rangle_T$, and one blind signature $/BSig\rangle_B$. The first regular signature $/Sig\rangle_A$ is generated using the voter's private key and is verified by the government agency T to confirm the voter's citizenship eligibility. The second regular signature $/Sig\rangle_T$,, issued by agency T, similar to a voting notification, certifies the voter's legal right to participate in the election. Finally, the blind signature $/BSig\rangle_B$, created by the election committee B, contains blinded candidate information and verification data for ballot counting. This blind signature ensures that the election committee cannot gain additional information about the ballot, and the ballot opener cannot identify the voter.

There are three cases of security issues associated with each of the three signature types: (a) forgery of a signature, where the attacker fabricates all parameters to create a counterfeit signature; (b) replacement of a legitimate signature, where the attacker intercepts and uses authentic signatures exchanged between users in an attempt to impersonate a legitimate voter; and (c) inversion attack, where the attacker tries to reverse the intercepted parameters to obtain the private keys of the parties involved in the signature. Additionally, we consider one more security concern of $/BSig\rangle_B$: (d) the possibility of a dishonest ballot opener.

We will present the three attack cases on $/Sig\rangle_A$ in Section 5.1, the attacks on $/Sig\rangle_T$ in Section 5.2, and the four attack cases on $/BSig\rangle_B$ in Section 5.3, respectively. Finally, in Section 5.4, we provide a comparison of Xu et al.'s voting system, other related works, and our QVS.

5.1 Attacks on $/Sig\rangle_A$

In this section, we define the three attack cases for $/Sig\rangle_A$. The scenario is illustrated in Figure 6.

(a) A forgery attack in which the attacker forges the voters' signatures $/Sig\rangle_{AE}$

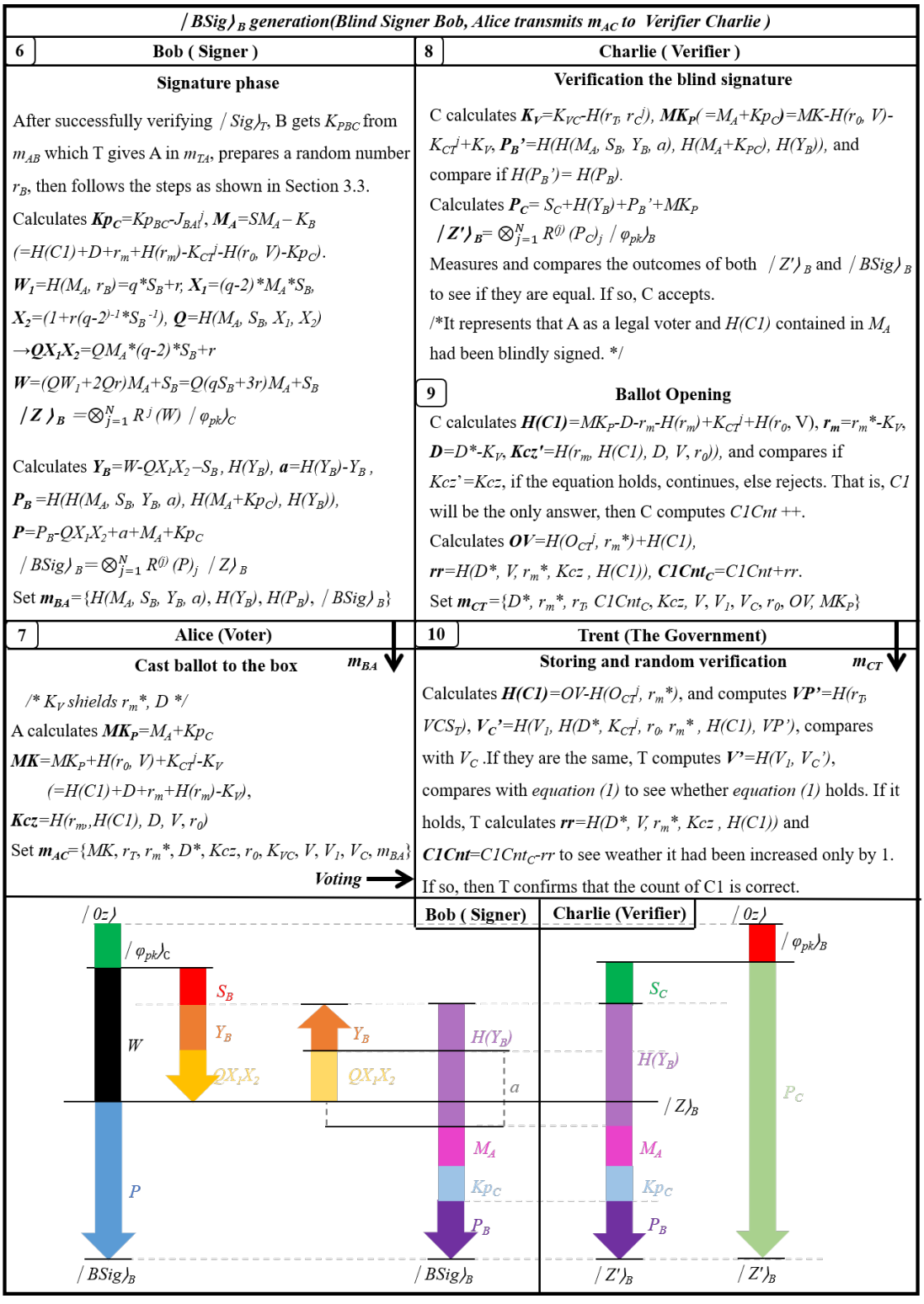

Figure 8. When voting, B generates $/ BSig\rangle_B$, and C verifies it and transfers m_{CT} to T for storage and random verification. The schematic diagram of C's verification process is shown in the lower part.

(1) **On the attacker's side (E)**

To forge a signature, E must use forged parameters, such as $m_E=(ID_{AE}, i_E)$, a random number r_{aE}, the number of voting times j_E, the verification sequences r_{AiE}^j, J_{AiE}^j, and the private key S_{AE}, and use these to perform the calculation of $|Sig\rangle_{AE}$..

E uses the fake verification sequences r_{AiE}^j, J_{AiE}^j to compute $Kp_{AE}=H(r_{AiE}^j,H(r_{AiE}^j, J_{AiE}^j))$. Then, E uses other fabricated parameters to compute the intermediate values W_{1E}, hq_E, X_{1E}, X_{2E}, Q_E, $(QX_1X_2)_E$, W_E, hw_E, hrs_E, hwr_E, sr_E, srh_E,Y_E, $Htot_E$, P_{2E}, and Kp_{AE}. (For brevity, below we use $(QX_1X_2)_E$ to denote $Q_EX_{1E}X_{2E}$.) Using these parameters, the attacker computes $hm_E=H(m_E, r_{aE}, hq_E, Q_E, X_{1E}, X_{2E}, P_{1E}, P_{2E}, Y_E, hw_E, sr_E, hrs_E, hwr_E, Kp_{AE})$ and creates the fake signature: $|Sig\rangle_{AE} = \bigotimes_{j=1}^{N} R^{(j)}(W_E+hm_E)_j \, |\varphi_{pk}\rangle_T$. Finally, E sets $m_{ATE}=\{ m_E, r_{aE}, hq_E, Q_E, X_{1E}, X_{2E}, hw_E, hrs_E, hwr_E, sr_E, Y_E, P_{1E}, P_{2E}, |Sig\rangle_{AE} \}$ and transmits it to T.

(2) **On the verifier T's side**

T first uses the received i_E contained in m_E to look up the corresponding $r_{AiE}^{j'}$ and $J_{AiE}^{j'}$. Then, T calculates $Kp_{AE}'= H(r_{AiE}^{j'},(r_{AiE}^{j'}, J_{AiE}^{j'}))$. Next, T computes $(QX_1X_2)_E$, $srh_E=sr_E+H(hw_E, (QX_1X_2)_E)$, $Htot_E$, and the angle parameter $hm_E'=H(m_E, r_{aE}, hq_E, Q_E, X_{1E}, X_{2E}, P_{1E}, P_{2E}, Y_E, hw_E, sr_E, hrs_E, hwr_E , Kp_{AE})$ for the verification process. Then, T proceeds by computing and comparing to verify whether $(X_{1E}X_{2E}- P_{1E}P_{2E})=sr+Htot$ hold. If this condition is satisfied, T generates the state $|Z_E\rangle= |\varphi_{pk}\rangle_{AE} +S_T+srh_E+(QX_1X_2)_E +Y_E+hm_E'$, measures it, and compares the measurement results to check if $|Z_E\rangle$ and $|Sig\rangle_{AE}$ are equal.

Apparently, $r_{AiE}^{j'}$ and $J_{AiE}^{j'}$, which T looks up in the random table, differ from E's forged values, causing both values hm (computed by attacker E and verifier T) to be different. This discrepancy allows T to detect that the measurement outcome of both states $|Z_E\rangle$ and $|Sig\rangle_{AE}$ do not match. Consequently, E's forgery attempt fails.

(b). An impersonation attack occurs when an attacker intercepts a legitimate voter's signature $|Sig\rangle_A$ and replaces it with their own identity to cast a vote.

(1) On the attacker's side (E)

E intercepts voter A's data $m_A=(ID_A, i)$ and attempts to replace the voter's identity i_A with i_E to cast a vote. That is, $m_{AE}=(ID_{AE}, i_E)$. Since E is not the actual voter, he cannot access A's private parameters. Therefore, by following the process, if E defines r_{aE}, r_{AiE}^{j}, J_{AiE}^{j}, S_{AE} and computes W_{1E}, hq_E, X_{1E}, X_{2E}, Q_E, $(QX_1X_2)_E$, W_E, hw_E, hrs_E, hwr_E, sr_E, srh_E, Y_E, $Htot_E$, P_{2E}, Kp_{AE}, then E calculates $hm_E=H(m, r_{aE}, hq_E, Q_E, X_{1E}, X_{2E}, P_{1E}, P_{2E}, Y_E, hw_E, sr_E, hrs_E, hwr_E, Kp_{AE})$, and uses values to generate the fake signature $\not| Sig\rangle_{AE} = \bigotimes_{j=1}^{N} R^{(j)}(W_E+hm_E)_j \not| \varphi_{pk}\rangle_T$. Finally, E sets $m_{ATE}=\{m, r_{aE}, hq_E, Q_E, X_{1E}, X_{2E}, P_{1E}, P_{2E}, Y_E, hw_E, sr_E, hrs_E, hwr_E, \not| Sig\rangle_{AE}\}$ and transmits it to T.

(2) **On verifier T's side**

Similar to the case (2) On the verifier T's side (in Section 5.1.Case (a)), T first uses the received i_E contained in m_{AE} to look up the corresponding r_{AiE}^{j}' and J_{AiE}^{j}' , then computes $Kp_{AE}'= H(r_{AiE}^{j}',(r_{AiE}^{j}', J_{AiE}^{j}'))$. Next, he computes $(QX_1X_2)_E$, srh_E, $Htot_E$, and the rotation angle $hm_E'=H(m_{AE}, r_{aE}, hq_E, Q_E, X_{1E}, X_{2E}, P_{1E}, P_{2E}, Y_E, hw_E, sr_E, hrs_E, hwr_E, Kp_{AE})$. He then verifies whether $(X_{1E}X_{2E}- P_{1E}P_{2E})=sr+Htot$ holds. Finally, T generates the state $\not| Z_E\rangle= \not| \varphi_{pk}\rangle_A+S_T+srh_E+(QX_1X_2)_E +Y+hm_E'$. Then, T measures and compares both outcomes with the measurement result of the state $\not| Sig\rangle_{AE}$.

As in the previous case, because r_{AiE}^{j}' and J_{AiE}^{j}' that T looks up in the random table differ from E's fabricated values, the quantum state rotation angles hm_E computed by E and T will not be the same. That is, $\not| \varphi_{pk}\rangle_T+S_{AE}+srh_E+Q_EX_{1E}X_{2E}+Y_E+hm_E$ does not equal to $\not| \varphi_{pk}\rangle_A +S_T+srh_E+(QX_1X_2)_E+Y+hm_E'$. This discrepancy causes the measurement outcome of state $\not| Z_E\rangle$ to differ from that of the other state $\not| Sig\rangle_{AE}$, so E's attack fails.

(C) A signature inversion attack occurs when an attacker intercepts a legitimate voters' signature $\not| Sig\rangle_A$, and attempts to obtain the $\not| \not| \varphi_{pk}\rangle_T+S_A\rangle$ by reversing the quantum signature.

(1) On attacker E's side

To obtain a valid value $\not| \varphi_{pk}\rangle_T+S_A$ for exploitation, E intercepts a message m_{AT} from genuine voter A, capturing parameters including m_A, r_A, hq, Q, X_1, X_2, P_1, P_2, Y, hw, sr, hrs, hwr, and $\not| Sig\rangle_A$ as shown in Figure 6. Since E is not the real voter A, he cannot retrieve K_A. Instead, he forges parameters r_{AiE}^{j} and J_{AiE}^{j} to calculate $Kp_{AE}=H(r_{AiE}^{j},(r_{AiE}^{j},$

J_{AiE}^{j})). Then, he computes QX_1X_2, $srh=sr+H(hw, QX_1X_2)$, and the signature rotation angle $hm_E=H(m_A, r_A, hq, Q, X_1, X_2, P_1, P_2, Y, hw, sr, hrs, hwr , Kp_{AE})$.

After completing the calculations, E attempts to reverse the angle $Y+QX_1X_2+ hm_E+srh$ on $|Sig\rangle_A$ to obtain the quantum state $|\varphi_{pk}\rangle_T+S_A\rangle_E$. Then, following steps similar to Case (a).(1), the attacker uses their random number r_{aE} and private key S_{AE} to compute W_{1E}, hq_E, X_{1E}, X_{2E}, Q_E,W_E, hw_E, hrs_E, hwr_E, sr_E, srh_E,Y_E, $Htot_E$, P_{2E}, Kp_{AE}, $Q_EX_{1E}X_{2E}$. They set $m_{AE}=(ID_{AE}, i_E)$ and calculate the angle parameter $hm_{AE}=H(m_{AE}, r_{AE}, hq_E, Q_E, X_{1E}, X_{2E}, P_{1E}, P_{2E}, Y_E, hw_E, sr_E, hrs_E, hwr_E, K_{AE})$.

After these computations, E generates a forged signature on the stolen quantum state, with the state ($|Sig\rangle_{AE}$) angle defined as $(|\ |\varphi_{pk}\rangle_T+S_A\rangle_E+(QX_1X_2)_E+srh_E+hm_{AE})$. Finally, E sets $m_{ATE}=\{m_{AE}, r_{AE}, hq_E, Q_E, X_{1E}, X_{2E}, P_{1E}, P_{2E}, Y_E, hw_E, sr_E, hrs_E, hwr_E, |Sig\rangle_{AE}\}$and transmits it to T.

(2) During verifier T's turn

T first uses the received i_E in m_{AE} to look up the corresponding $r_{AiE}^{j'}$ and $J_{AiE}^{j'}$, then computes $Kp_{AE}'=H(r_{AiE}^{j'},(r_{AiE}^{j'}, J_{AiE}^{j'}))$. Next, T computes $(QX_1X_2)_E$, srh_E, hm_{AE}', and $Htot_E$ based on m_{ATE}. For verification, T calculates $srh_E=sr_E+H(hw_E, (QX_1X_2)_E)$ and checks if $(X_{1E}X_{2E}- P_{1E}P_{2E})=sr_E+Htot_E$. If not, T rejects. Otherwise, T generates the state $|Z_E\rangle = |\varphi_{pk}\rangle_A +S_T+srh_E+(QX_1X_2)_E+Y+hm_E'$, measures it, and compares the outcomes with the received quantum state $|Sig\rangle_{AE}$.

As in previous examples, because $r_{AiE}^{j'}$ and $J_{AiE}^{j'}$ that T retrieves from his own random table differ from E's fabricated values, the hm_E computed by E does not equal hm_E' calculated by T. This discrepancy causes the measurement outcomes of $|Z_E\rangle$ and $|Sig\rangle_{AE}$ to mismatch. Moreover, since K_{AE} was forged by E, the value obtained by reversing the angle is incorrect for A's private key combined with T's public key. Thus, attacker E's attempt fails.

5.2 Attacks on $|Sig\rangle_T$

The government signature $|Sig\rangle_T$ (see Figure 7 for its role in the voting system) acts as a voting notice. Assuming it is vulnerable to forgery, counterfeiting, and reversal attacks, an attacker might impersonate a voter to trick the government agency T into signing parameters. Below, we describe three attack scenarios and explain why they fail.

(a). Forging a government signature $|Sig\rangle_{TE}$

(1) **On Attacker E's side**

Attacker E generates fake parameters to forge the signature $|Sig\rangle_T$ between voter A and government T. E calculates $m_{TE}=(ID_{AE}||ID_{TE}, i_E)$, forges random number r_{TE} and private key S_{TE} and prepares fake pairs for other system roles generated by the government, such as (r_{AiE}^j, J_{AiE}^j), $(r_{BAiE}{}^{j\prime}, J_{BAiE}^{j\prime})$, and (r_{CAiE}^j, J_{CAiE}^j). E then computes $K_{AE}=H(r_{AiE}^j, J_{AiE}^j)$, $K_{BE}=H(r_{BAiE}^{j\prime}, J_{BAiE}^{j\prime})$, and $K_{CE}=H(r_{CAiE}^j, J_{CAiE}^j)$, $Kp_{BE}=H(r_{BAiE}^j, K_{BE})$, $Kp_{CE}=H(r_{CiE}^j, K_{CE})$, $Ex_{CBE}=K_{BE}+K_{CT}^j$, $Kp_{BCE}=r_{BiE}^j+J_{BiE}^j+Kp_{CE}$, $Kp_{CAE}=J_{AiE}^j+Kp_{CE}$, $VP_E=H(r_{TE}, VCS_{TE})$, $K_{VE}=H(H(r_{TE}, r_{AiE}^j), H(r_{TE}, r_{CE}^j), VP_E)$, $K_{VAE}=K_{VE}+H(r_{TE}, r_{AiE}^j)$, $K_{VCE}=K_{VE}+H(r_{TE}, r_{CE}^j)$. Since E impersonates voter A to government T, he could try to simplify several calculations. For example, he may simplify calculations, computing $Ex_{CBE}=K_{BE}-K_{CTE}^j$ instead of $Ex_{CBAE}= J_{AiE}^j+K_{BE}-K_{CTE}^j$.

Secondly, E calculates S_{AE}, W_{1E}, hq_E, X_{1E}, X_{2E}, Q_E, $(QX_1X_2)_E$, W_E, hw_E, hrs_E, hwr_E, sr_E, srh_E, Y_E, P_{1E}, $Htot_E$, P_{2E} , and $hm_E=H(m_E, r_E, hq_E, Q_E, X_{1E}, X_{2E}, P_{1E}, P_{2E}, Y_E, hw_E, sr_E, hrs_E, hwr_E, K_{AE}, Kp_{BE})$, and generates $|Sig\rangle_{TE}=\bigotimes_{j=1}^{N} R^{(j)}(W_E+hm_E)_j\ |\varphi_{pk}\rangle_B$. That is, $|Sig\rangle_{TE}=\ |\varphi_{pk}\rangle_B+S_{TE}+srh_E+(QX_1X_2)_E+Y_E+(hm_E)_j$.

Even if E is an insider, without knowledge of J_{BAi}^j, E cannot derive A's Kp_C from Kp_{BC} in m_{TA}. To impersonate voter A, E must declare the blinding parameter r_{mE}, compute $D_E=H(r_{mE}, S_{AE})$, select candidate option $H(C2)$, add K_{AE}, K_{BE} to form $K_{ABE}=K_{AE}+K_{BE}$, then generates the fake $V_E=H(V_{1E}, V_{CE})$ and produce a fake ballot.

$SM_{AE}=H(C2)+D_E+r_{mE}+H(r_{mE})-H(r_{0E}, V_E)+Ex_{CBE}-Kp_{CE}$ ***.....…equation(2)***

That is, $H(C2)=SM_{AE}-K_{BE}-D_E-r_{mE}-H(r_{mE})+K_{CT}^j+H(r_{0E}, V_E)+Kp_{CE}$.

E sets m_{ABE}={m_{TE}, r_{TE}, hq_E, Q_E, X_{1E}, X_{2E}, P_{1E}, P_{2E}, Y_E, hw_E, sr_E, hrs_E, hwr_E, K_{ABE}, $|Sig\rangle_{TE}$, SM_{AE}, Kp_{BCE}, K_{VAE}, K_{VCE}, Ex_{CBE}, VP_E, Kp_{CAE}} and transmits it to B for verification.

(2) **On verifier B's side**

Election committee staff B first look up $r_{BAiE}{}^{j}{}'$ and $J_{BAiE}{}^{j}{}'$ in the random table using i_E from m_{TE}. If found, based on the received K_{ABE} in m_{ABE}, B calculates $K_B'=H(r_{BAi}{}^{j}{}', J_{BAi}{}^{j}{}')$, $Kp_B'=H(r_{BAi}{}^{j}{}', K_B')$, and $K_{AE}'=K_{ABE}$ - K_B'. Next, B computes parameters like $(QX_1X_2)_E$, $srh_E=sr_E+H(hw_E, (QX_1X_2)_E)$, $Htot_E$ (step 5, Figure 7), and the rotation angle $hm_E'=H(m_E, r_E, hq_E, Q_E, X_{1E}, X_{2E}, P_{1E}, P_{2E}, Y_E, hw_E, sr_E, hrs_E, hwr_E, K_{ABE}, Kp_{BE}')$.

According to the verification steps, B first checks whether $(X_{1E}X_{2E}- P_{1E}P_{2E})= sr_E +Htot_E$ holds. If so, B performs a rotation on $|\varphi_{pk}\rangle_T$ to form state $|Z_E\rangle$, then measures the state $|Z_E\rangle=(|\varphi_{pk}\rangle_T+S_B+srh_E+(QX_1X_2)_E)+Y+hm_E')$ and compares it with the measurement outcome of state $|Sig\rangle_{TE}$. However, since B uses the real $r_{BAiE}{}^{j}$ and $J_{BAiE}{}^{j}$ of voter A_i's i_E, these values will not match the fabricated $r_{BAiE}{}^{j}{}'$ and $J_{BAiE}{}^{j}{}'$ created by E. Consequently, the key K_A' computed by B differs from K_{ABE} declared by E in hm_E, making the angle hm_E' calculated by B unequal to the forged angle hm_E by E. This discrepancy causes the measurement outcomes of $|Z_E\rangle$ and $|Sig\rangle_{TE}$ to differ, causing E's forgery to fail. Moreover, E lacks knowledge of Kp_C and K_V, which vary among voters to generate V1 and V, and does not know K_B to create SM_A, which B uses to generate $|BSig\rangle_B$ to pass C's verification. Since SM_{AE}-K_B computed by B differs from SM_{AE}-K_{BE} calculated by E, the value $M_A(SM_{AE}$-$K_B)+Kp_C(Kp_{CA}$-$J_{Ai}{}^{j})$ computed by B also differs from E's. Therefore, it cannot pass C's $|BSig\rangle_B$ verification because $H(M_A+Kp_C)$ in P_B in B's $|BSig\rangle_B$ does not equal $H(M_{AE}+Kp_{CE})$ (where $M_{AE}+Kp_{CE}=MK_E$-$H(r_{0E}, V_E)$-$K_{CT}{}^{j}+K_{VE})$, as shown in step 7 of Figure 8. That is, E.s attempt fails.

(b). Intercepting a legitimate government signature $|Sig\rangle_T$ and transmitting it to Election Committee B with a fake ballot

(1) **On Attacker E's side**

In this case, attacker E intercepts the message m_{TA} sent from T to A, which includes the legitimate $|Sig\rangle_T$, Ex_{CBA}, and other parameters, or the message m_{AB} from A to B, which also contains m_{TA} that includes $|Sig\rangle_T$. Whether E intercepts the message from T to A

or from A to B (as shown in steps 3 and 4 of Figure 7), E must use a fake J_{AiE}^{j} to compute Ex_{CBAE} and forge other parameters to create a ballot, since E lacks the random tables. Thus, in both scenarios (from m_{TA} or m_{AB}), E retains the legitimate $\langle Sig \rangle_T$, but tries to fabricate a ballot.

For example, even if E is an insider knowing K_{CT}^{j}, E can fake a ballot by declaring blinding parameter r_{mE} and private key S_{AE}, changing candidate option to $H(C2)$, faking K_{BE}, r_{0E}, V_E, and Kp_{CE} , then calculating $D_E=H(r_{mE}, S_{AE})$ and $K_{ABE}=K_{AE}+K_{BE}$. The resulting fake ballot becomes SM_{AE}, as shown in equation (2), as shown in ***equation (2)***.

E sets $m_{ABE}=\{i, SM_{AE}, K_{ABE}, m_{TA}\}$ and sends it to B for verification, as shown in step 4 of Figure 7.

(2) **On verifier B's side**

After receiving m_{ABE}, election committee B retrieves r_{BAi}^{j} and J_{BAi}^{j} from the random table using i from m_{ABE} to calculate $K_B=H(r_{BAi}^{j}', J_{BAi}^{j}')$, $Kp_B=H(r_{BAi}^{j}, K_B)$, and $K_{AE}'=K_{ABE}-K_B$. B then calculates $hm'=H(m_T, r_T, hq, Q, X_1, X_2, P_1, P_2, Y, hw, sr, hrs, hwr, K_{AE}', Kp_B)$ along with other verification parameters. Next, B verifies whether $(X_1X_2-P_1P_2)=sr+Htot$ holds. Finally, B computes the verification angle $S_B+srh+(QX_1X_2)_E+Y+hm'$ to set state $\langle Z_E \rangle = \langle \varphi_{pk} \rangle_T+S_B+srh+(QX_1X_2)_E+Y+hm'$. B measures both states $\langle Z_E \rangle$ and $\langle Sig \rangle_T$ and compares the outcomes to confirm equality.

From the above, the K_{AE} forged by attacker E does not match the correct $K_A(=H(r_{Ai}^{j}, J_{Ai}^{j}))$ generated by government T for the legitimate voter A. Additionally, the correct pair $(r_{BAi}^{j}, J_{BAi}^{j})$ retrieved by B differs from the forged pair $(r_{BAiE}^{j}', J_{BAiE}^{j}')$ created by E. Therefore, the verification fails due to the incorrect value of E's K_{AE} and Kp_{BE} in hm_E.

Thus, $\langle Sig \rangle_T = \langle \varphi_{pk} \rangle_B +S_T+srh+QX_1X_2+Y+hm$ does not equal $\langle \varphi_{pk} \rangle_B + S_T+srh+QX_1X_2+Y+hm'$. Moreover, due to the fact that $SM_A=H(C1)+D+r_m+H(r_m)+K_B-K_{CT}^{j}-H(r_0, V)-Kp_C$, even if E holds the value K_{AB} in m_{AB} and Ex_{CBA} in m_{TA} (steps 3 and 4, Figure 7), he cannot alter the candidate's name $H(C1)= SM_A-K_B-D-r_m-H(r_m)+K_{CT}^{j}+H(r_0, V)+Kp_C$, where $V=H(V_1, V_C)$, without knowing K_B, D and Kp_C. This is because D is obscured by K_V, K_B is owned by B, and Kp_C is A's secret. Thus, E's attack fails.

(C). Intercepts legitimate $|Sig\rangle_T$ and attempts to obtain the state $|\ |\varphi_{pk}\rangle_B+S_T\rangle$ by reversing $|Sig\rangle_T$.

(1) **Attacker E's turn.**

Assuming attacker E intercepts a legitimate $|Sig\rangle_T$ and m_{AB} to obtain m_T, r_T, hq, Q, X_1, X_2, P_1, P_2, Y, hw, sr, hrs, hwr, i, SM_A, and K_{AB}, but lacks Kp_B and Kp_C , E must generate fake values (r_{BAiE}^{j}, J_{BAiE}^{j}) to compute $K_{BE}=H(r_{BAiE}^{j}, J_{BAiE}^{j})$, $K_{CE}= H(r_{CAiE}^{j}, J_{CAiE}^{j})$, $Kp_{BE}=H(r_{BAiE}^{j}, K_{BE})$, and $Kp_{CE}=H(r_{CAiE}^{j}, K_{CE})$. Then E calculates $K_{AE}=K_{AB}–K_{BE}$, QX_1X_2, $srh=sr+H(hw, QX_1X_2)$, and $hm_E=H(m_T, r_T, hq, Q, X_1, X_2, P_1, P_2, Y, hw, sr, hrs, hwr , K_{AE}, Kp_{BE})$. Finally, E performs a reverse calculation on $|Sig\rangle_T$ using the angle formed by summing Y, QX_1X_2, hm_E, and srh to obtain $|\ |\varphi_{pk}\rangle_B+S_T\rangle_E$ $(=|Sig\rangle_T\text{-}Y\text{-}QX_1X_2\text{-}hm_E\text{-}srh)$.

Next, E replaces the intercepted signature parameters (related to the intercepted signature $|Sig\rangle_T$) with forged ones using the following steps. First, E calculates the necessary parameters to impersonate T's signing, including $m_E=(ID_{AE}||ID_{TE}, i_E)$, W_{1E}, hq_E, X_{1E}, X_{2E}, Q_E, $(QX_1X_2)_E$, W_E, hw_E, hrs_E, hwr_E, sr_E, srh_E, Y_E, P_{1E}, $Htot_E$, and P_{2E}. Then, E computes the quantum signature rotation angle: $hm_E'=H(m_E, r_E, hq_E, Q_E, X_{1E}, X_{2E}, P_{1E}, P_{2E}, Y_E, hw_E, sr_E, hrs_E, hwr_E, K_{AE}, Kp_{BE},)$. This generates a forged signature $|Sig\rangle_{TE}=(|\varphi_{pk}\rangle_B+S_T)_E+Y_E+(QX_1X_2)_E+srh_E+hm_E$. Finally, E sets $m_{TAE}=\{m_E, r_E, hq_E, Q_E, X_{1E}, X_{2E}, P_{1E}, P_{2E}, Y_E, hw_E, sr_E, hrs_E, hwr_E, |Sig\rangle_{TE}, Kp_{BCE}, K_{VA}, K_{VC}, Ex_{CBA}, Kp_{CA}\}$

As in previous cases, attacker E impersonates both voter A and T. E defines parameters such as the blinding parameter r_{mE}, private key S_{AE}, r_{0E}, V_E, $D_E=H(r_{mE}, S_{AE})$, candidate option $C2$, $Ex_{CBE}=K_{BE}\text{-}K_{CTE}^{j}$, and $Kp_{BCE}=J_{BAiE}^{j}+Kp_{CE}$, then generates the forged ballot SM_{AE} (equation 2). Finally, E sets $m_{TBE}=\{i_E, SM_{AE}, K_{ABE}, m_{TAE}\}$ and transmits it to B.

(2) **Verifier B's side**

After receiving m_{ABE} from E, election committee B uses the included i_E to retrieve (r_{BAi}^{j}', J_{BAi}^{j}') and calculates $K_B'= H(r_{BAi}^{j}', J_{BAi}^{j}')$, $Kp_B'=H(r_{BAi}^{j}', K_B')$, $K_{AE}'=K_{ABE}$ - K_B', $Htot_E$, $(QX_1X_2)_E$, $srh_E=sr_E+H(hw_E, (QX_1X_2)_E)$, and the angle $hm_E'=H(m_E, r_E, hq_E, Q_E, X_{1E}, X_{2E}, P_{1E}, P_{2E}, Y_E, hw_E, sr_E, hrs_E, hwr_E, K_{AE}', Kp_{BE}')$ in the quantum signature. B then verifies whether $(X_{1E}X_{2E}\text{-}\ P_{1E}P_{2E})= sr_E +Htot_E$. If valid, B generates and measures the

verification state $|Z_E\rangle = |\varphi_{pk}\rangle_T + S_B + srh_E + (QX_1X_2)_E + Y_E + hm_E'$ and compares it with the measurement result of state $|Sig\rangle_{TE}$.

As in previous cases, since B's retrieved values r_{BAi}^{j}', J_{BAi}^{j}' do not match the attacker E's fabricated values (r_{BAiE}^{j}, J_{BAiE}^{j}), B's calculated K_{AE}' will differ from the forged K_{AE}. Thus, the hm_E' verified by B differs from the hm_E computed and sent by E. In other words, the measurement outcomes of both $|Z_E\rangle$ and $|Sig\rangle_{TE}$ differ, so the angle hm_E used by E to reverse on $|Sig\rangle_T$ will produce a state different from the actual state $|\ |\varphi_{pk}\rangle_B + S_T\rangle$.

That is, $|Sig\rangle_{TE} = (\ |\varphi_{pkB} + S_T\rangle)_E + Y_E + (QX_1X_2)_E + srh_E + hm_E$ is not equal to $|\varphi_{pk}\rangle_T + S_B + Y_E + (QX_1X_2)_E + srh_E + hm_E'$. This causes E's signature forgery to fail..

In conclusion of Sections 5.1 and 5.2, if attacker E attempts to replace any parameters in $|Sig\rangle_A$ or $|Sig\rangle_T$ (e.g. m_A, r_A in m_{AT} or m_T, r_T in m_{TA}) and transmit them between A and T, as shown in Figures 6 and 7, T will calculate *hm*, *Htot* and related parameters to verify if X_1X_2-P_1P_2 equals *srh*+*Htot*. T will detect inequality and reject the message because all parameters are hashed into *hm* and *Htot*. Thus, E's attack will fail regardless of the altered parameter. For instance, if E substitute *sr* with *sr'*, T computes *Htot'=H(m, r_A, hq, Q, X_1, X_2, Y, P_1, hw, sr', hrs, hwr)* and finds *sr'*+*Htot'* does not equal to X_1X_2-P_1P_2. Since $|\varphi_{pk}\rangle_A$ is formed by rotating A's private key (secret degree S_A) from the zero degree of the quantum state, E cannot derive S_A from the quantum state $|\varphi_{pk}\rangle_A$, which T uses as a basis to form $|Z\rangle$ to examine parameters (step 2, Figure 6).

5.3 Attacks on Blind Signatures $|BSig\rangle_B$

The blind signature includes government agency B's commitment to the voter's ballot without revealing its content. As before, we examine three attack scenarios: (a) forging, (b) intercepting and altering the embedded candidate name, and (c) reversing the quantum blind signature. Additionally, we address a fourth security concern about $|BSig\rangle_B$: (d) a dishonest ballot opener. This scenario can be referenced to Figure 8.

(a). Forging a blind signature $|BSig\rangle_{BE}$

(1) **Attacker E's side**

Attacker E attempts to vote using a forged ballot by fabricating all required parameters, including the blinding random number r_{mE}, private key S_{AE}, selected candidate $H(C2)$, random pairs $(r_{AiE}^{j}, J_{AiE}^{j})$ and $(r_{CAiE}^{j}, J_{CAiE}^{j})$, as well as r_{0E}, r_{TE}, r_{CE}^{j}, and K_{CTE}^{j}. E then computes $D_E=H(r_{mE}, S_{AE})$, $K_{CE}= H(r_{CAiE}^{j}, J_{CAiE}^{j})$, $Kp_{CE}=H(r_{CAiE}^{j}, K_{CE})$, $VP_E=H(r_{TE}, VCS_T)$, $V_{1E}=H(Kp_{CE}, D_E^*, K_{CTE}^{j}, r_{0E}, r_{mE}^*, D_E, r_{mE}, H(C1))$, $V_{CE}=H(V_{1E}, H(D_E^*, K_{CTE}^{j}, r_{0E}, r_{mE}^*, H(C1), VP_E))$, and $V_E= H(V_{1E}, V_{CE})$, $K_{VE}=H(H(r_{TE}, r_{AiE}^{j}), H(r_{TE}, r_{CE}^{j}), VP_E)$, and $K_{VCE}=K_{VE}+H(r_{TE}, r_{CE}^{j})$.

Finally, E generates a fake ballot: $M_{AE}=H(C2)+D_E+r_{mE}+H(r_{mE})-K_{CTE}^{j}-H(r_{0E}, V_E)-Kp_{CE}$.

After forging the ballot, E generates a fake blind signature by counterfeiting parameters such as the random number r_{BE}, private key S_{BE}, and i. E then computes W_{1E}, Q_E, X_{1E}, X_{2E}, $(QX_1X_2)_E$, W_E, Y_{BE}, $H(Y_{BE})$, $a_E=H(Y_{BE})-Y_{BE}$, $P_{BE}=H(H(M_{AE}, S_{BE}, Y_{BE}, a_E), H(M_{AE}+K_{PCE}), H(Y_{BE}), a_E)$, and $P_E=P_{BE}-(QX_1X_2)_E+M_{AE}+K_{PCE}$. Finally, E generates the forged signature $|BSig\rangle_{BE}=\bigotimes_{j=1}^{N} R^j (P_E)_j \; |Z\rangle_{BE} (=Y_{BE}+S_{BE}+S_C)$.

Before voting, attacker E impersonating voter A must compute verification parameters: $MK_{PE}=M_{AE}+K_{PCE}$, $MK_E=MK_{PE}+H(r_{CE}, V_E)+K_{CTE}^{j}-K_{VE}$, $r_{mE}^*=r_{mE}+K_{VE}$, $D_E^*=D_E+K_{VE}$, and $K_{CZE}=H(r_{mE}, D_E, V_E, r_{0E}, H(C2))$. He then sets $m_{BAE}=\{H(M_{AE}, S_{BE}, Y_{BE}, a_E), H(Y_{BE}), H(P_{BE}), |BSig\rangle_{BE}\}$, and $m_{ACE}=\{MK_E, r_{TE}, r_{mE}^*, D_E^*, K_{CZE}, r_{0E}, V_E, V_{1E}, V_{CE}, K_{VCE}, m_{BAE}\}$, which he transmits to ballot opener C for verification (step 7, Figure 8).

(2) **Verifier C's side**

The ballot opener C first computes $K_V'=K_{VCE}-H(r_{TE}, r_C^{j})$, $MK_P'(=M_{AE}+Kp_{CE})=MK_E-H(r_{0E}, V_E)-K_{CT}^{j}+K_V'$, and $P_{BE}'=H(H(M_A, S_B, Y_B, a), H(MK_P'), H(Y_{BE}))$. It then checks whether $H(P_{BE}')=H(P_{BE})$ holds. However, without knowledge of S_B and S_C, C must rely on $|\varphi_{pk}\rangle_B$ or $|\varphi_{pk}\rangle_C$ for verification of $|BSig\rangle_B$ as shown in Section 4.4.(3). Since Y_B is unrevealed and S_B is B's secret, the degree of $|Z'\rangle_B$ for C's verification depends on $S_C+S_{BE}+H(Y_{BE})+P_{BE}'+M_{AE}+Kp_{CE}= S_C+S_{BE}+H(Y_{BE})+H(H(M_{AE}, S_{BE}, Y_{BE}, a_E), H(M_{AE}+Kp_{CE}), H(Y_{BE}))+M_{AE}+Kp_{CE}$.

Assuming E impersonates B to use $/\varphi_{pk}\rangle_C$ as a basis, the left side of equation (2) includes SB and PB, which E cannot handle.

When E uses $(M_{AE}+Kp_{CE})(=MK_E-H(r_{0E}, V_E)-K_{CTE}^j+K_{VE})$ to be signed by B, the one-way property of the hash function and E's lack of knowledge of r_C^j and K_V, make it computationally infeasible for E to find S_B and M_A+Kp_C that satisfy *equation (2)*. Thus, E's attack fails.

(b). Intercept legitimate $/BSig\rangle_B$ and attempt to alter only the candidate's name.

(1) Attacker E's side

Attacker E may intercept the message m_{BA} sent from B to A, including legitimate parameters $/BSig\rangle_B$ such as, $H(M_A, S_B, Y_B, a)$, $H(Y_B)$, and $H(P_B)$, as shown in step 6 of Figure 8. E can also intercept parameters in m_{AC} (containing MK, r_T, r_m*, D*, K_{CZ}, r_0, K_{VC}, V_1, V_C) when A transmits m_{AC} to C as shown in step 7 of Figure 8. Both cases are discussed below.

In this interception scenario, E intercepts m_{BA} and replaces the selected candidate name $H(C1)$ with $H(C2)$ on the ballot. To do this, E fabricates necessary ballot parameters, including private key S_{AE}, blind numbers r_{mE}, r_{TE}, r_{0E}, K_{CTE}^j, i_E, r_{CE}^j, (r_{AiE}^j, J_{AiE}^j), and (r_{CAiE}^j, J_{CAiE}^j). E calculates $K_{CE}=H(r_{CAiE}^j, J_{CAiE}^j)$, $Kp_{CE}=H(r_{CAiE}^j, K_{CE})$, $D_E=H(r_{mE}, S_{AE})$, $VP_E=H(r_{TE}, VCS_T)$, $V_{1E}=H(Kp_{CE}, D_E$*, K_{CT}^j, r_{0E}, r_{mE}*, D_E, r_{mE}, $H(C1))$, $V_{CE}=H(V_{1E}$, $H(D_E$*, K_{CT}^j, r_{0E}, r_{mE}*, $H(C1)$, $VP_E))$, $V_E=H(V_{1E}, V_{CE})$, $K_{VE}=H(H(r_{TE}, r_{AiE}^j), H(r_{TE}, r_{CE}^j)$, $VP_E)$, and $K_{VCE}=K_{VE}+H(r_{TE}, r_{CE}^j)$. Finally, E generates $M_{AE}=H(C2)+D_E+r_{mE}+H(r_{mE})-K_{CTE}^j-H(r_{0E}, V_E)-Kp_{CE}$, $MK_{PE}=M_{AE}+Kp_{CE}$, and $MK_E = MK_{PE}+H(r_{0E}, V_E)+K_{CTE}^j-K_{VE}$. After computing verification parameters r_{mE}* $=r_{mE}+K_{VE}$, D_E* $=D_E+K_{VE}$ and $K_{CZE}=H(r_{mE}, H(C2), D_E, V_E, r_{0E})$ for the fake ballot, E sets $m_{BAE}=\{H(M_A, S_B, Y_B, a), H(Y_B), H(P_B), /BSig\rangle_{BE}\}$ and $m_{ACE}=\{MK_E, r_{TE}, r_{mE}$*, D_E*, K_{CZE}, r_{0E}, K_{VCE}, V_E, V_{1E}, V_{CE}, $m_{BAE}\}$and transmitted them to C.

To intercept m_{AC} (containing MK, r_T, r_m*, D*, K_{CZ}, r_0, K_{VC}, V_1, V_C) from A, unlike intercepting m_{BA} from B, E fabricates S_{AE}, K_{CTE}^j, r_{mE}, r_{CE}^j, (r_{CAiE}^j, J_{CAiE}^j), computes $D_E=H(r_{mE}, S_{AE})$, $K_{CE}=H(r_{CAiE}^j, J_{CAiE}^j)$, $Kp_{CE}=H(r_{CAiE}^j, K_{CE})$, and $K_{VE}= K_{VC}-r_{CE}^j$. E then calculates $V=H(V_1, V_C)$ and ballot $M_{AE}=H(C2)+D_E+r_{mE}+H(r_{mE})-K_{CTE}^j-H(r_0, V)-Kp_{CE}$. Next, $MK_{PE}=M_{AE}+Kp_{CE}$, $MK_E=MK_{PE}+H(r_0, V)+K_{CTE}^j-K_{VE}$. Finally, E sets

m_{BAE}={ $H(M_A, S_B, Y_B, a), H(Y_B), H(P_B), |BSig\rangle_{BE}$} and m_{ACE} ={MK_E, r_{TE}, r_{mE}*, D_E*, K_{CZE}, r_0, K_{VC}, V, V_1, V_C, m_{BAE}} and transmits it to C (step 7, Figure 8).

(2) Verifier C's side

As before, the ballot opener C calculates $K_V'=K_{VC}-H(r_{TE}, r_C^j)$, $MK_P'=MK_E-H(r_{0E}, V_E)-K_{CT}^j+K_V'$, and $P_B'=H(H(M_A, S_B, Y_B, a), H(MK_P'), H(Y_B))$, then compares $H(P_B')$ with the received $H(P_B)$in m_{BA}. These steps show that r_C^j' and thus K_V computed by C do not match the forged K_{VE} created by E. Moreover, the real ballot parameter M_A in SM_A sent by voter A to B is part of the blind signature rotation angle (P_C on $|\varphi_{pk}\rangle_B$), where $P_C=S_C+H(Y_B)+P_B'+MK_P$ and $P_B=P+QX_1X_2-a-M_A-Kp_C$. Therefore, E cannot alter M_A as in step 8 of Figure 8. If E changes M_A to M_{AE}, the value $H(P_B')(=H(H(M_A, S_B, Y_B, a), H(M_A+Kp_C), H(Y_B)))$ calculated by C would not match $H(P_B)$ from the original voter, since K_V and MK_P are known only to C ($MK_P=MK-H(r_0, V)-K_{CT}^j+K_V$) . Without MK_P, E cannot forge a valid P_B (to be calculated in $H(P_B))$ for C's verification, causing verification to fail and rejection by C..

(C). Intercept legitimate $|BSig\rangle_B$ and attempt to obtain the state $|\ |\varphi_{pk}\rangle_C+S_B\rangle$ by reversing the blind signature.

(1) **Attacker E's side**

E intercepts the legitimate $|BSig\rangle_B$ and all parameters in m_{AC} from A to C (step7, Figure 8). Without knowing K_V, E forges r_{CE}^j and calculates $K_{VE}= H(r_T, r_{CE}^j)$ and $MK_{PE}=MK-H(r_0, V)-K_{CTE}^j+K_{VE}$. Then, E calculates $P_{BE}=H(H(M_A, S_B, Y_B, a), H(MK_{PE}), H(Y_B))$ and reverses the angle $P_{BE}+MK_{PE}+H(Y_B)$ on $|BSig\rangle_B$ to get state $|\ |\varphi_{pk}\rangle_C+S_B\rangle_E$ (i.e., reverse the state $|BSig\rangle_B$ by angle $P_{BE}+MK_{PE}+H(Y_B)$).

Next, E discards the intercepted parameters in m_{AC} and replaces them with forged values: random number r_{BE}, private key S_{BE}, MK_{AE}. E then calculates W_{1E}, Q_E, X_{1E}, X_{2E}, $(QX_1X_2)_E$, W_E, Y_{BE}, $H(Y_{BE})$, $a_E=H(Y_{BE})-Y_{BE}$, $P_{BE}=H(H(M_{AE}, S_{BE}, Y_{BE}, a_E), H(M_{AE}+ K_{PCE})$, $H(Y_{BE})$, and $P_{BE}=P_{BE}-(QX_1X_2)_E+a_E+M_{AE}+K_{PCE}$. Finally, E generates a fake blind signature $|BSig\rangle_{BE}$ by adding the reversed quantum state $|\ |\varphi_{pk}\rangle_C+S_B\rangle_E$ with these angle parameters, forming state $|BSig\rangle_{BE}= (|\varphi_{pkc}+S_B\rangle)_E+H(Y_{BE})+MK_{PE}+P_{BE}$.

Finally, in step 4 of Figure 7, attacker E forges parameters including blind number r_{mE}, private key S_{AE}, and candidate option $H(C2)$. E calculates $D_E=H(r_{mE}, S_{AE})$, $r_{mE}*=r_{mE}+K_{VE}$, $D_E*=D_E+K_{VE}$, and $MK_{AE}= MK_{PE}+H(r_0, V)+K_{CT}^j-K_V$. After that, E sets $m_{BAE}=\{H(M_{AE}, S_{BE}, Y_{BE}, a_{BE}), H(Y_{BE}), H(P_{BE}), \lfloor BSig \rfloor_{BE}\}$ and $m_{ACE} = \{MK_{AE}, r_T, r_{mE}*, D_E*, K_{CZE}, r_0, K_{VC}, V, V_1, V_C, m_{BAE}\}$, as shown in step 7 of Figure 8, and sends m_{ACE} to C.

(2) **Verifier C's side**

The ballot opener C calculates $K_V=H(r_T, r_C^j)$, $MK_{PE}'=MK_{AE}-H(r_0, V)-K_{CT}^j+K_V$, and $P_{BE}'=H(H(M_{AE}, S_{BE}, Y_{BE}, a_E), H(M_{AE}+ Kp_C'), H(Y_{BE}))$, then compares $H(P_{BE}')$ to $H(P_{BE})$. Since the r_C^j used by C in computing K_V does not match E's forged r_{CE}^j, the P_{BE}' value, which includes MKP, differs from E's P_{BE}. Consequently, verification fails, and C rejects.

(d). The ballot opener is dishonest and alters the candidate's name on the ballot.

(1) **Ballot opener C's side**

Since C has the correct K_V from K_{VC}, he computes $r_m=r_m*-K_V$, where r_m* is sent from A to C in m_{AC}. If C tries to change candidate name from *C1* to *C2* , the only parameter he can alter is D $(=H(r_m, S_A))$. However, he must keep $MK_P(=SM_A-K_B+Kp_C)=H(C1)+D+r_m+H(r_m)-K_{CT}^j-H(r_0, V)$ unchanged; otherwise, T will detect the alteration during quality random inspection (described below in Section 5.3.(d).(2)). However, D is computed as $D=MK_P-H(C1)-r_m-H(r_m)+K_{CT}^j+H(r_0, V)$. That is, $H(C1)= MK_P-r_m-H(r_m)-D+K_{CT}^j+H(r_0, V)$, as shown in ***equation (1)*** in Section 4.3 (2). Since $D=MK_P-H(C1)-r_m-H(r_m)+K_{CT}^j+H(r_0, V)$, changing H(C1) to H(C2) while maintaining MK_P constant is impossible. Although C knows M_A, K_V, r_m, and D, if he replaces $H(C1)$ with $H(C2)$, and computes $OV=H(C2)+H(O_{CT}^j, r_{mE}*)$, he must compute $V_{CE}=H(V_1, H(D*, K_{CT}^j, r_0, r_{mE}*, H(C2), VP))$ and send $m_{CT}=\{D*, r_{mE}*, C_1Cunt, K_{CZ}, V, V_1, V_{CE}, r_0, OV, MK_P\}$ to T. But VP contains VCS_T, known only to T and voters, so C cannot find VP_E to compute V_{CE} for satisfying T's random verification. In other words, C cannot alter the candidate name without detection, and his attempt fails. We now describe T's random inspection below.

(2) **Verifier T's turn to conduct quality random inspections.**

During the election, government T can randomly verify a ballot to check if its associated MK_P is accepted. T uses the message m_{CT} from C and consults his table to find O_{CT}^j and VCS_T. Then, T computes $H(C1)=OV\text{-}H(O_{CT}^j, r_m^*)$, followed by $VP'=H(r_T, VCS_T)$ and $V_C'=H(V_1, H(D^*, K_{CT}^j, r_C, r_m^*, H(C1), VP'))$ to confirm that $H(C1)$ corresponds to the ballot opened by C, not another candidate's name (e.g., $H(C2)$). If C tries to substitute C1 with C2, T will detect it because VCS_T is known only to T and embedded by the voter. Thus, C cannot compute $VP_E=H(r_T, VCS_T)$ to generate a valid V_C and can only keep V_C unchanged. Furthermore, T computes $V'=H(V_1, V_C')$ and verifies if $V'=V$. If so, T checks that $C_1Cnt(=C_1Cnt_C\text{-}rr)$ has increased only by one. Therefore, if C mistakes $C1$ for $C2$, T will identify the error. Even if all the shared parameters except $J_{A_l}^j$ (the voter's secret for deducing Kp_C) are revealed, an adversary E cannot succeed. Although, C might compute $OV_E=H(C2)+H(O_{CT}^j, r_{mE}^*)$, T will verify the correctness of V_C and ***equation (2)***. Since $V_C=H(V_1, H(D^*, K_{CT}^j, r_C, r_m^*, H(C1), VP))$ and $V=H(V_1, V_C')$, we see that V_C contains T's secret VCS_T shared with all voters, and V contains $V_1(=H(Kp_C, D^*, K_{CT}^j, r_0, r_m^*, D, r_m, H(C1)))$ which hashes Kp_C, r_m^*, r_m, D^*, and D—both K_V (shared among T, C, and voters) and Kp_C (the voter's secret) are essential. These must satisfy ***equation (1):*** $H(C1)=MK_P\text{-}D\text{-}r_m\text{-}H(r_m)\text{-}H(r_0, V)=MK_P\text{-}D\text{-}r_m\text{-}H(r_m)\text{-}H(r_0, H(V_1, V_C))$. Without Kp_C in $V_1(=H(Kp_C, D^*, K_{CT}^j, r_0, r_m^*, D, r_m, H(C1)))$, it is computationally infeasible for C to find Kp_C due to the one-way property of hash functions. Moreover, our QVS is untraceable before and after the $\lfloor BSig \rfloor_B$ generation, ensuring message unlinkability and enhancing voting security. To simulate a real voting scenario, government agencies B and T can discard all parameters they produced after voting day, preventing ballots from being traced to specific voters

Thus, our QVS is a truly secure, pseudonym-free, and anonymous voting system because it does not reuse any pseudonym once $\lfloor BSig \rfloor_B$ has been generated. Therefore, even T cannot identify the candidate chosen by any specific voter.

In conclusion of Section 5.3, a ballot contains $\lfloor BSig \rfloor_B$ and several parameters. If E transmits the ballot exactly as stolen from voter A without any changes, it is useless because the ballot will be verified and revealed as the original candidate chosen by A.

5.4 Comparison with Xu et al's scheme and other related work in quantum voting

In this section, we first compare our voting system with the scheme proposed by Xu et al.[94], focusing on properties such as blindness, anonymity, undeniability, untraceability, and unforgeability.

In Xu et al.'s system, a public table enables voters to verify that the candidate they voted for is indeed their chosen candidate through pseudonyms. However, this design compromises the strict definition of anonymity. Additionally, the initial shared key allows election committee B to forge votes by impersonating any voter. Voter A cannot deny that a vote is not their own because the overall computations and parameters can be legitimately produced by them according to the protocol. Therefore, Xu's system cannot prevent risks related to unforgeability and undeniability. In contrast, our scheme does not require such a public table. Election committee B can verify the voter's identity only through the pseudonym stored in the pre-shared random table and cannot access the content of the vote (blindness). The ballot opener knows the voter's selection but cannot identify the voter (anonymity). Furthermore, any voting concerns are addressed by government authorities B, C, and T. Thus, our scheme ensures unforgeability, undeniability, and anonymity, as described in Sections 5.1 through 5.3. We have effectively resolved these issues.

Apart from [94], it is evident that our scheme overcomes the limitations identified in previous literature. Our approach employs secret random tables shared among all roles, functioning similarly to the one-time secret used in Chaum's classical ballot protocol [131], thereby ensuring unconditional security. Based on this and the security analyses presented in Section 5, our method clearly possesses the following properties: (1) privacy (anonymity), (2) security (prevention of double voting), (3) flexibility (the scheme requires no adaptation when there are three or more candidates or voters), and it also prevents (4) collusion among parties, (5) illegal operations, and (6) cheating by authorities. Additionally, it does not require (7) a secure channel or (8) quantum key

distribution (QKD). For instance, if voter *A* attempts to vote twice, *B* will detect this because *Ai* must send *i* to **B** in message m_{AB}, as shown in step 4 of Figure 7. Based on the limitations and advantages explicitly stated in each related work, we have compiled the following comparison table.

Table 9 Comparison of ours and voting system in literature

properties / schemes		(1)	(2)	(3)	preventing			needn't	
					(4)	(5)	(6)	(7)	(8)
ours		O	O	O	O	O	O	O	O
[94]			X		O	O	X	O	O
[113]		O	O		X	X	X		
[114]		O		X					
[115]		X	O						
[116]		O						X	O
[117]		O	O	X	X	X	X	X	O
[118]	A	O	O	X				O	O
	B	O	X		O			O	O
	C	X	O					O	O
[119]		X						X	X

O: possesses the property

X: doesn't possess the property

" ": uncertainty

6 Conclusion

In this article, we propose a quantum voting scheme. After cryptanalysis, we confirmed that our scheme not only resists forgery attacks but also achieves the essential functions

of undeniability, anonymity, and untraceability both before and after $|BSig\rangle_B$'s generation which are the dimensions of necessary attributes required in any voting system. In other words, our method offers an alternative approach for implementing a voting system in democratic countries. As shown in Table 9 above, our scheme outperforms those in the literature, except that it requires a trusted third party. Therefore, our quantum voting scheme is practical and can be easily applied worldwide.

Reference

[1] KATZ, Jonathan, et al. Handbook of applied cryptography. CRC press, 1996.

[2] S. Saeednia, "An identity-based society-oriented signature scheme with anonymous signers," Information processing Letters, vol. 83, no. 6, pp. 295–299, 2002.

[3] C. L. Hsu, T. S. Wu, and T. C. Wu, "Group-oriented signature scheme with distinguished signing authorities," Future Generation Computer Systems, vol. 20, no. 5, pp. 865–873, 2004.

[4] C. Y. Lin, T. C. Wu, F. Zhang, and J. J. Hwang, "New identity-based society-oriented signature schemes from pairings on elliptic curves," Applied Mathematics and Computation, vol.160, no. 1, pp. 245–260, 2005.

[5] Z. Shao, "Certificate-based verifiably encrypted signatures from pairings," Information Sciences, vol. 178, no. 10, pp.2360–2373, 2008.

[6] J. Zhang and J. Mao, "A novel ID-based designated verifier signature scheme," Information Sciences, vol. 178, no. 3, pp.766–773, 2008.

[7] Y. F. Chung, Z. Y. Wu, and T. S. Chen, "Ring signature scheme for ECC-based anonymous sign crypton," Computer Standards and Interfaces, vol. 31, no. 4, pp. 669–674, 2009.

[8] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signature: delegation of the power to sign messages," IEICE—Transactions on Fundamentals of Electronics, vol. E79-A, no. 9, pp. 1338–1354, 1996.

[9] R. Lu, Z. Cao, and Y. Zhou, “Proxy blind multi-signature scheme without a secure channel,” Applied Mathematics and Computation, vol. 164, no. 1, pp. 179–187, 2005.

[10] H. F. Huang and C. C. Chang, “A novel efficient （t, n）threshold proxy signature scheme,” Information Sciences, vol. 176, no. 10,pp. 1338–1349, 2006.

[11] B. Kang, C. Boyd, and E. Dawson, “Identity-based strongde signated verifier signature schemes: attacks and new construction,” Computers and Electrical Engineering, vol. 35, no. 1,pp. 49–53, 2009.

[12] K. L. Wu, J. Zou, X. H. Wei, and F. Y. Liu, “Proxy group signature: a new anonymous proxy signature scheme,” in Proceedings of the 7th International Conference on Machine Learning and Cybernetics （ICMLC’08）, pp. 1369–1373, Kunming, China, July 2008.

[13] Z. Shao, “Improvement of identity-based proxy Mult signature scheme,” The Journal of Systems and Software, vol. 82,no. 5, pp. 794–800, 2009.

[14] Z. H. Liu, Y. P. Hu, X. S. Zhang, and H. Ma, “Secure proxy signature scheme with fast revocation in the standard model, “Journal of China Universities of Posts and Telecommunications, Vol. 16, no. 4, pp. 116–124, 2009.

[15] Y. Yu, C. Xu, X. Huang, and Y. Mu, “An efficient anonymous proxy signature scheme with provable security,” Computer Standards and Interfaces, vol. 31, no. 2, pp. 348–353, 2009.

[16] F. Cao and Z. Cao, “A secure identity-based proxy multi signature scheme,” Information Sciences, vol. 179, no. 3, pp.292–302, 2009.

[17] A. Yang and P. Peng, “A modified anonymous proxy signature with a trusted party,” in Proceedings of the 1st International Workshop on Education Technology and Computer Science（ETCS’09）, pp. 233–236,Wuhan, China, March 2009.

[18] J. H. Hu and J. Zhang, “Cryptanalysis and improvement of a threshold proxy signature scheme,” Computer Standards and Interfaces, vol. 31, no. 1, pp. 169–173, 2009.

[19] Y. Yu, C. X. Xu, X. S. Zhang, and Y. J. Liao, “Designated verifier proxy signature scheme without random oracles,” Computers and Mathematics with Applications, vol. 57, no. 8, pp. 1352–1364, 2009.

[20] J. H. Zhang, C. L. Liu, and Y. I. Yang, “An efficient secure proxy verifiably encrypted signature scheme,” Journal of Network and Computer Applications, vol. 33, no. 1, pp. 29–34, 2010.

[21] B. D. Wei, F. G. Zhang, and X. F. Chen, “ID-based ring proxy signatures,” in Proceedings of the IEEE International Symposium on Information Theory （ISIT’07）, pp. 1031–1035, Nice, France, June 2007.

[22] T. S. Wu and H. Y. Lin, “Efficient self-certified proxy CAE scheme and its variants,” The Journal of Systems and Software, Vol. 82, no. 6, pp. 974–980, 2009.

[23] S. Lal and V. Verma, “Identity based Bi-designated verifier proxy signature schemes,” Cryptography E-print Archive Report 394, 2008.

[24] S. Lal and V. Verma, “Identity based strong designated verifier proxy signature schemes,” Cryptography E-print Archive Report 394, 2006.

[25] C. Y. Yang, S. F. Tzeng, and M. S. Hwang, “On the efficiency of non-repudiable threshold proxy signature scheme with known signers,” The Journal of Systems and Software, vol. 73, no. 3, pp.507–514, 2004.

[26] H. Xiong, J. Hu, Z. Chen, and F. Li, “On the security of an identity based multi-proxy signature scheme,” Computers and Electrical Engineering, vol. 37, no. 2, pp. 129–135, 2011.

[27] Y. Sun, C. Xu, Y. Yu, and Y. Mu, “Strongly unforgeable proxy signature scheme secure in the standard model,” The Journal of Systems and Software, vol. 84, no. 9, pp. 1471–1479, 2011.

[28] Y. Sun, C. Xu, Y. Yu, and B. Yang, “Improvement of a proxy multi-signature scheme without random oracles,” Computer Communications, vol. 34, no. 3, pp. 257–263, 2011.

[29] Z. Liu, Y. Hu, X. Zhang, and H. Ma, “Provably secure multi-proxy signature scheme with revocation in the standard model,” Computer Communications, vol. 34, no. 3, pp. 494–501, 2011.

[30] H. Bao, Z. Cao, and S. Wang, “Improvement on Tzenget al.’s non repudiable threshold multi-proxy multi-signature scheme with shared verification,” Applied Mathematics and Computation, vol. 169, no. 2, pp. 1419–1430, 2005.

[31] J. G. Li and Z. F. Cao, “Improvement of a threshold proxy signature scheme,” Computer Research and Development, vol. 39,no. 11, pp. 1513–1518, 2002.

[32] Y. Yu, Y. Mu, W. Susilo, Y. Sun, and Y. Ji, “Provably secure proxy signature scheme from factorization, “Mathematical and Computer Modelling, vol. 55, no. 3-4, pp. 1160–1168, 2012.

[33] K. Shum and V. K. Wei, “A strong proxy signature scheme with proxy signer privacy protection,” in Proceedings of the11th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises （WETICE’02）, pp.55–56, Pittsburgh, Pa, USA, 2002.

[34] N. Y. Lee and M. F. Lee, “The security of a strong proxy signature scheme with proxy signer privacy protection,” Applied Mathematics and Computation, vol. 161, no. 3, pp. 807–812,2005.

[35] Chou, Jue-Sam."A novel anonymous proxy signature scheme." Advances in Multimedia 2012 （2012）: 13.

[36] C.Dwork,M.Naor,A.Sahai,“Concurrent zero-knowledge. ”Proceedings of 30th ACMSTOC’98, 1998, pp. 409–418.[37] Y. Aumann, M. Rabin, “Efficient deniable authentication of long messages. ”Int. Conf. on Theoretical Computer Science in Honor of Professor Manuel Blum’s 60th birthday, http://www.cs.cityu.edu.hk/dept/video.html. April 20–24, 1998.

[38] Mario Di Raimondo, Rosario Gennaro and Hugo Krawczyk, “Deniable 17 Authentication and Key Exchange, ”ACM CCS’06, October, 2006, Alexandria, Virginia, USA.

[39] C. Boyd, W. Mao, K. Paterson, “Deniable authenticated key establishment for Internetprotocols.”11th International Workshop on Security Protocols, Cambridge （UK）, April 2003.

[40] C. Boyd&W. Mao,“Key agreement using statically key ed authentication.” Applied Cryptology and Network Security（ACNS’04）, LNCS 3089, pp.248-262.

[41] Z.Shao,“Efficient deniable authentication protocol based on generalized ElGamal signature scheme. ”Computer Standards & Interfaces 26 （5）, 2004, pp.449–454.

[42] R. Lu,Z.Cao,“A new deniable authentication protocol from bilinear pairings.” Applied Mathematics and Computation 168 （2）, 2005, pp.954–961.

[43] R.Lu,Z.Cao,“Non-interactive deniable authentication protocol based on factoring. ”Computer Standards & Interfaces 27 （4）, 2005, pp.401–405.

[44] TianjieCao, Dongdai Lina and RuiXue,“ An efficient ID-based deniable authentication protocol from pairings, ”Proceedings of the 19th International Conference on Advanced Information Networking and Applications （AINA’05）, IEEE, 2005.

[45]Wei-Bin Lee, Chia-Chun Wu and Woei-Jiunn Tsaur,“A novel deniable authentication protocol using generalized ElGamal signature scheme,” Information Science, 2006.

[46] Rongxing Lu, Zhenfu Cao, “Erratumto“ Non-interactive deniable authentication protocol based on factoring”[Computer Standards & Interfaces27（2005） 401–405].”Computer Standards & Interfaces 29, pp.275, February 2007

[47] Chun-Ta Li, Min-Shiang Hwang and Chi-Yu Liu, “An electronic voting protocol with deniable authentication for mobile ad hoc networks. ”Computer Communication 31（10）, pp.2534-2540, June 2008.

[48] Bin Wang and ZhaoXia Song, “Anon-interactive deniable authentication scheme based on designated verifier proofs. ”Information Sciences 179（6）, pp.858-865, March 2009.

[49] Taek-Young Youn, Changhoon Lee and Young-Ho Park, “An efficient non-interactive deniable authentication scheme based on trapdoor commitment schemes.” Computer Communications, In Press, Corrected Proof, March 2010.

[50] Lein Harn and Jian Ren, “Design of Fully Deniable Authentication Service for E-mail Applications. ”IEEE Communications Letters 12（3）, pp.219-221, March 2008.

[51]Chen, Yalin, Jue-Sam Chou, and Chi-Fong Lin. "A Novel Non-interactive Deniable Authentication Protocol with Designated Verifier on elliptic curve cryptosystem." IACR Cryptology ePrint Archive 2010（2010）: 549.

[52] F. Kerschbaum, N. Oertel, and L. W. F. Chaves, “Privacy preserving computation of benchmarks on item-level data using RFID.” in Proceedings of the 3rd ACM Conference on Wireless Network Security （WiSec ‘10）, pp. 105–110, March2010.

[53] M. O. Rabin, “How to exchange secrets with oblivious transfer.”Tech. Rep. TR-81, Aiken Computation Lab, Harvard University, Cambridge, Mass, USA, 1981.

[54] S.Even, O. Goldreich, and A. Lempel, “A randomized protocol for signing contracts.”Communications of the ACM, vol. 28,no. 6, pp. 637–647, 1985.

[55] G. Brassard, C. Crepeau, and J.-M. Robert, “All-or-nothing disclosure of secrets.” in Proceedings of the International Conference on Advances in Cryptology （CRYPTO ‘86）, vol. 263of Lecture Notes in Computer Science, pp. 234–238, 1986.

[56] Chou, Jue-Sam, and Yi-Shiung Yeh."Mental poker game based on a bit commitment scheme through network." Computer Networks 38.2 （2002）: 247-255.[57] M. Bellare and S. Micali, “Non-interactive oblivious transfer and application,” in Proceedings of the International Conference on Advances in Cryptology （CRYPTO ‘89）, vol. 435 of Lecture Notes in Computer Science, pp. 547–557, 1989.

[58] M. Naor and B. Pinkas, “Oblivious transfer with adaptive queries,” in Proceedings of the International Conference on Advances in Cryptology （CRYPTO ‘99）, Lecture Notes in Computer Science, pp. 573–590, 1999.

[59] M. Naor, B. Pinkas, and R. Sumner, “Privacy preserving auctions and mechanism design,” in Proceedings of the 1st ACM Conference on Electronic Commerce, 1999.

[60] M. Naor and B. Pinkas, “Distributed oblivious transfer,” in Proceedings of the International Conference on Advances in Cryptology（CRYPTO ‘00）, vol. 1976 of Lecture Notes in Computer Science, 2000.

[61] M. Naor and B. Pinkast, “Oblivious transfer and polynomial evaluation.” in Proceedings of the 31st Annual ACM Symposium on Theory of Computing （FCRC ‘99）, pp. 245–254, May 1999.

[62] M. Naor and B. Pinkas, “Efficient oblivious transfer protocols.”in Proceedings of the 12th annual ACM-SIAM symposium on Discret Mathematics（SODA ‘01）, pp. 448–457, 2001.

[63] H. Ghodosi, “On insecurity of Naor-Pinkas’ distributed oblivious transfer,” Information Processing Letters, vol. 104, no.5, pp. 179–182, 2007.

[64] Y. Mu, J. Zhang, and V. Varadharajan, “m out of n oblivious transfer,” in Proceedings of the 7th Australasian Conference on Information Security and Privacy （ACISP ‘02）, vol. 2384 of Lecture Notes in Computer Science, pp. 395–405, 2002.

[65] H. Ghodosi and R. Zaare-Nahandi, “Comments on the ‘m out of n oblivious transfer. ”Information Processing Letters, vol. 97,no. 4, pp. 153–155, 2006.

[66] W. Ogata and K. Kurosawa, “Oblivious keyword search. ”Journal of Complexity, vol. 20, no. 2-3, pp. 356–371, 2004.

[67] C. K. Chu and W. G. Tzeng, “Efficient k-out-of-n oblivious transfer schemes with adaptive and non-adaptive queries.” In Proceedings of the 8th International Workshop on Theory and Practice in Public Key Cryptography （PKC ‘05）, pp. 172–183,January 2005.

[68] J. Zhang and Y. Wang, “Two provably secure k-out-of-n oblivious transfer schemes,” Applied Mathematics and Computation, Vol. 169, no. 2, pp. 1211–1220, 2005.

[69] H. F. Huang and C. C. Chang, “A new design for efficient tout-n oblivious transfer scheme.” in Proceedings of the 19thInternational Conference on Advanced Information Networking and Applications （AINA ‘05）, pp. 28–30, March 2005.

[70] A. Parakh, “Oblivious transfer using elliptic curves.” in Proceedings of the 15th International Conference on Computing（CIC ‘06）, pp. 323–328, November 2006.

[71] S. Kimand G. Lee, “Secure verifiable non-interactive oblivious transfer protocol using RSA and Bit commitment on distributed environment. “Future Generation Computer Systems, vol.25, no. 3, pp. 352–357, 2009.

[72] Y. F. Chang and W. C. Shiao, “The essential design principles of verifiable non-interactive OT protocols.” in Proceedings of the 8th International Conference on Intelligent Systems Design and Applications （ISDA ‘08）, pp. 241–245, November 2008.

[73] L. M. Kohnfelder, “On the signature reblocking problem in public-key cryptography.” Communications of the ACM, vol.21, no. 2, p. 179, 1978.

[74] S. Halevi and Y. T. Kalai, “Smooth projective hashing and two-message oblivious transfer.” Cryptology ePrint Archive2007/118, 2007.

[75] J. Camenisch, G. Neven, and A. Shelat, “Simulatable adaptive oblivious transfer.” in Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, vol. 4515 of Lecture Notes in Computer Science, pp.573–590, 2007.

[76] M. Green and S. Hohenberger, “Blind identity-based encryption and simulatable oblivious transfer. ”Cryptology ePrint Archive 2007/235, 2007.

[77] J. Qin, H. W. Zhao, and M. Q. Wang, “Non-interactive oblivious transfer protocols.” in Proceedings of the International Forum on Information Technology and Applications （IFITA ‘09）,pp. 120–124, May 2009.

[78] C. C. Chang and J. S. Lee, “Robust t-out-of-n oblivious transfer mechanism based on CRT. ”Journal of Network and Computer Applications, vol. 32, no. 1, pp. 226–235, 2009.

[79] X. Ma, L. Xu, and F. Zhang, “Oblivious transfer with timed release receiver’s privacy.” Journal of Systems and Software, vol.84, no. 3, pp. 460–464, 2011.

[80]Chou, Jue-Sam."A novel k-out-of-n oblivious transfer protocol from bilinear pairing." Advances in Multimedia 2012 （2012）: 3.

[81] Shi, Wei-Min, et al. "A scheme on converting quantum signature with public verifiability into quantum designated verifier signature." Optik 164 (2018): 753-759.

[82] Shi, Wei-Min, et al. "A non-interactive quantum deniable authentication protocol based on asymmetric quantum cryptography." Optik-International Journal for Light and Electron Optics 127.20 (2016): 8693-8697.

[83] Shi, Wei-Min, et al. "A restricted quantum deniable authentication protocol applied in electronic voting system." Optik-International Journal for Light and Electron Optics 142 (2017): 9-12.

[84] Shi, Wei-Min, et al. "A scheme on converting quantum signature with public verifiability into quantum designated verifier signature." Optik 164 (2018): 753-759.

[85] Wen, Xiaojun, et al. "A weak blind signature scheme based on quantum cryptography." Optics Communications 282.4 (2009): 666-669.

[86] Shi, Wei-Min, et al. "An efficient quantum deniable authentication protocol without a trusted center." Optik-International Journal for Light and Electron Optics 127.16 (2016): 6484-6489.

[87] Lee, Hwayean, et al. "Arbitrated quantum signature scheme with message recovery." Physics Letters A 321.5-6 (2004): 295-300.

[88] Wang, Jian, et al. "Comment on: “Arbitrated quantum signature scheme with message recovery”[Phys. Lett. A 321 (2004) 295]." Physics Letters A 347.4-6 (2005): 262-263.

[89] Shemin, P. A., and K. S. Vipinkumar. "E–payment system using visual and quantum cryptography." Procedia Technology 24 (2016): 1623-1628.

[90] Hwang, Tzonelih, et al. "New arbitrated quantum signature of classical messages against collective amplitude damping noise." Optics communications 284.12 (2011): 3144-3148.

[91] Yang, Yu-Guang, and Qiao-Yan Wen. "Arbitrated quantum signature of classical messages against collective amplitude damping noise." Optics Communications 283.16 (2010): 3198-3201.

[92] Qi, Su, et al. "Quantum blind signature based on two-state vector formalism." Optics Communications 283.21 (2010): 4408-4410.

[93] Qiu, Lirong, Feng Cai, and Guixian Xu. "Quantum digital signature for the access control of sensitive data in the big data era." Future Generation Computer Systems (2018).

[94] Xu, Rui, et al. "Quantum group blind signature scheme without entanglement." Optics Communications 284.14 (2011): 3654-3658.

[95] Tang, Bing, and De-Jun Li. "Quantum signature of discrete breathers in a nonlinear Klein–Gordon lattice with nearest and next-nearest neighbor interactions." Communications in Nonlinear Science and Numerical Simulation 34 (2016): 77-85.

[96] Shi, Wei-Min, Yu-Guang Yang, and Yi-Hua Zhou. "Quantum signature-masked authentication schemes." Optik-International Journal for Light and Electron Optics 126.23 (2015): 3544-3548.

[97] Possignolo, Rafael Trapani, Cintia Borges Margi, and Paulo SLM Barreto. "Quantum-assisted QD-CFS signatures." Journal of Computer and System Sciences 81.2 (2015): 458-467.

[98] Yi, Haibo, and Zhe Nie. "Side-channel security analysis of UOV signature for cloud-based Internet of Things." Future Generation Computer Systems (2018).

[99] Moro, Giorgio J., Giulia Dall'Osto, and Barbara Fresch. "Signatures of Anderson localization and delocalized random quantum states." Chemical Physics (2018).

[100] Aumasson, Jean-Philippe. "The impact of quantum computing on cryptography." Computer Fraud & Security 2017.6 (2017): 8-11.

[101] Stadelmann, Kathrin, et al. "The SERS signature of PbS quantum dot oxidation." Vibrational Spectroscopy 91 (2017): 157-162.

[102] Jimenez, Kevin, and Jose Reslen. "Thermodynamic signatures of an underlying quantum phase transition: A grand canonical approach." Physics Letters A 380.34 (2016): 2603-2607.

[105] A. Kaushik, A.K. Das, D. Jena, “A novel approach for simple quantum digital signature based on asymmetric quantum cryptography. ”Int. J. Appl. Innov.Eng. Manage.（IJAIEM）6（June（6））（2013）

[106]Shi, W. M., Wang, Y. M., Zhou, Y. H., & Yang, Y. G.（2018）. Cryptanalysis on quantum digital signature based on asymmetric quantum cryptography. Optik-International Journal for Light and Electron Optics, 154, 258-260.

[107] Yalin Chen and Jue-Sam Chou and Fang-Qi Zhou. A publicly verifiable quantum signature scheme based on asymmetric quantum cryptography

[108] Yalin Chen and Jue-Sam Chou and Liang-Chun Wang and Yu-Yuan Chou. A publicly verifiable quantum blind signature scheme without entanglement based on asymmetric cryptography.

[109] ABC News (Australia)-《Taiwan mid-terms: Electorate turns away from ruling Democratic Progressive Party》

[110] Bloomberg-《Taiwan's President Handed Stinging Defeat in Regional Elections》

[111] Liberty Times Net, “Vote-c Liberty Times Net, “Vote-counting while voting causes controversy in Taiwan's 2018 local elections,”[Online].

[112] Central Election Commission (Taiwan), “Election mishaps: Simultaneous vote-counting and voting during the 2018 local elections,” [Online].

[113] Hillery Mark, et al. "Towards quantum-based privacy and voting." Physics Letters A 349.1-4 (2006): 75-81.

[114] Dolev Shahar, Itamar Pitowsky, and Boaz Tamir. "A quantum secret ballot." arXiv preprint quant-ph/0602087 (2006).

[115] Horoshko, Dmitri, and Sergei Kilin. "Quantum anonymous voting with anonymity check." Physics Letters A 375.8 (2011): 1172-1175.

[116] Shi, Wei-Min, et al. "A restricted quantum deniable authentication protocol applied in electronic voting system." Optik 142 (2017): 9-12.

[117] Xue, Peng, and Xin Zhang. "A simple quantum voting scheme with multi-qubit entanglement." Scientific reports 7.1 (2017): 7586.

[118] Vaccaro, Joan Alfina, Joseph Spring, and Anthony Chefles. "Quantum protocols for anonymous voting and surveying." Physical Review A—Atomic, Molecular, and Optical Physics 75.1 (2007): 012333.

[119] Zheng Mengce, et al. "A practical quantum designated verifier signature scheme for E-voting applications." Quantum Information Processing 20 (2021): 1-22.

[120] Finogina Tamara, and Javier Herranz. "On remote electronic voting with both coercion resistance and cast-as-intended verifiability." Journal of Information Security and Applications 76 (2023): 103554.

[121] Gupta Sweta, et al. "End to end secure e-voting using blockchain & quantum key distribution." Materials Today: Proceedings 80 (2023): 3363-3370.

[122] Emami Ashkan, et al. "A scalable decentralized privacy-preserving e-voting system based on zero-knowledge off-chain computations." Journal of Information Security and Applications 79 (2023): 103645.

[123] Ye Jun, et al. "An Electronic Voting Scheme with Privacy Protection." Procedia Computer Science 243 (2024): 1248-1256.

[124] Aurangzeb Muhammad, et al. "Enhancing cybersecurity in smart grids: Deep black box adversarial attacks and quantum voting ensemble models for blockchain privacy-preserving storage." Energy Reports 11 (2024): 2493-2515.

[125] Li Quanrun, et al. "An efficient quantum-resistant undeniable signature protocol for the E-voting system." Journal of information security and applications 81 (2024): 103714.

[126] Wu Chunhui, et al. "An efficient rejection-free threshold ring signature from lattices and its application in receipt-free cloud e-voting." Computer Standards & Interfaces 94 (2025): 104008.

[127] Sarier Neyire Deniz. "Efficient, usable and Coercion-Resistant Blockchain-Based E-Voting." Journal of Information Security and Applications 92 (2025): 104074.

[128] Wu Chunhui, et al. "An efficient rejection-free threshold ring signature from lattices and its application in receipt-free cloud e-voting." Computer Standards & Interfaces 94 (2025): 104008.

[129] Piétri Yoann. "Quantum cryptography." Imperial College London (2020).

[130] Wang, Feihu, et al. "Quantum blind signature protocol based on single qubit rotation." Optics Communications 583 (2025): 131629.

[131] D. Chaum, in Advances in Cryptology: Euro Crypt '88 proceedings, LNCS, edited by C. G. Gunther, (Springer-Verlag, Berlin, 1998), Vol,403, pp. 177