International Journal of Theoretical Physics (2023) 62:17
https://doi.org/10.1007/s10773-023-05288-x

A Verifiable (k, n)-Threshold Quantum Secure Multiparty Summation Protocol

Fulin Li[1,2,3] · Hang Hu[1] · Shixin Zhu[1,2] · Ping Li[1,2]

Received: 17 August 2022 / Accepted: 11 January 2023 / Published online: 21 January 2023

Abstract

Quantum secure multiparty summation plays an important role in quantum cryptography. In the existing quantum secure multiparty summation protocols, the (n, n)-threshold protocol has been given extensive attention. To increase the applicability of quantum secure multiparty summation protocols, a new quantum secure multiparty summation protocol based on Shamir's threshold scheme and d-dimensional GHZ state is proposed in this paper. In the proposed protocol, $i)$ it has a (k, n)-threshold approach; $ii)$ in the result output phase, it can not only detect the existence of deceptive behavior but also determine the specific cheaters; $iii)$ compared with the (n, n)-threshold quantum secure multiparty summation protocols, it needs less computation cost when L satisfies $L > 4$, where L is the length of each participant's secret. In addition, the security analysis shows that our protocol can resist intercept-resend attack, entangle-measure attack, Trojan horse attack, and participant attack.

Keywords Quantum secure multiparty summation · d-dimensional GHZ state · Shamir's threshold scheme · Hash function · Security

Supported by the National Natural Science Foundation of China (No.U21A20428, No.61972126, No.12171134)

✉ Hang Hu
huhang123hfgydx@163.com

Fulin Li
lflsxx66@163.com

Shixin Zhu
zhushixinmath@hfut.edu.cn

Ping Li
lpmath@126.com

[1] School of Mathematics, HeFei University of Technology,
Hefei 230009, Anhui, People's Republic of China

[2] Intelligent Interconnected Systems Laboratory of Anhui Province,
Hefei 230009, Anhui, People's Republic of China

[3] Xinjiang Agricultural University, Urumqi 830052, Xinjiang, People's Republic of China

1 Introduction

In practical applications, the research of secure multiparty computation is of great significance, which was first proposed by Yao [1]. The goal of studying secure multiparty computation is to enable numerous participants to collaboratively compute a particular function without disclosing their respective input information. They all receive accurate output information after the calculation. In the current research, to solve various practical problems, many secure multiparty computation protocols are proposed, respectively. For instance, secure multiparty sorting protocols [2, 3], secure multiparty cloud computation protocols [4, 5], and secure multiparty summation protocols [6–9]. In this paper, we only pay attention to the secure multiparty summation, which is one of the common security problems in current life. Secure multiparty summation protocols allow the existence of multiple participants, and each participant has a private message. Subsequently, these participants jointly compute a summation function under the condition that no personal private message is revealed. Finally, the output information of the summation function is available to each participant.

With the emergence of quantum algorithms [10, 11], quantum secure multiparty computation has attracted much attention [12–16]. At present, the research on quantum secure multiparty summation (QSMS) is still limited. And most of the existing QSMS protocols are (n, n)-threshold protocols. That is to say, in the result output phase, the output information of the summation function can only be obtained when all participants are present. In 2017, Shi et al. [17] designed a special quantum two-party summation protocol, but when one of the two participants is dishonest, the other participant will not get the correct output information. In 2022, Ye et al. [18] proposed a lightweight 2-dimensional three-user secure quantum summation protocol, but if the quantum system is free-space, then the protocol will be less applicable than the protocols of high-dimensional quantum system in some cases. And another consideration in protocols [17, 18] is that the number of participants is limited. Therefore, Liu et al. [19] in 2017, Yang et al. [20] in 2018, Lv et al. [21] in 2019 and Wang et al. [22] in 2021 respectively proposed an (n, n)-threshold QSMS protocol. Compared to previous related protocols, they are more efficient and solve the limitation of the number of participants. However, through analysis, we found that protocols [19–22] have one thing in common that is not well considered, that is, if any participant provides wrong information in the output result phase, the honest participants cannot get the correct output information and cannot find the specific cheaters. In particular, protocols [19–21] still require non-negligible computation cost, as they need to measure more message particles.

In addition, it is also worth noting if one or more participants fail during the result output phase. The applicability of these (n, n)-threshold protocols will be affected. Therefore, to increase the applicability of QSMS protocols. In 2020, Song et al. [23] proposed a (k, n)-threshold quantum secure multiparty computation protocol based on Lagrange unitary operation and Shamir's threshold secret sharing, which has higher computational efficiency than previous similar protocols. However, in the result output phase, protocol [23] can only identify the existence of the deception and cannot find the specific deceivers. The same year, Sutradhar et al. [24] introduced a quantum multiparty protocol that supports (k, n)-threshold summation and contrasts it with comparable protocols, demonstrating that it is more advantageous in terms of both communication and computation. However, the honest participants in protocol [24] may not obtain the correct output information because there is a possibility that some participants provide invalid measurement results in the result output phase.

In this paper, we present a new verifiable (k, n)-threshold QSMS protocol by using Shamir's threshold scheme, d-dimensional GHZ (Greenberger-Horne-Zeilinger) state, and hash function. Compared with the existing QSMS protocols, our protocol has the following properties:

(1) Based on Shamir's threshold scheme and d-dimensional GHZ state, we construct a (k, n)-threshold QSMS protocol. Compared with the (n, n)-threshold QSMS protocols, our protocol is more flexible and needs less computation cost when L meets $L > 4$, where L denotes the length of each participant's secret.

(2) Based on hash function, in the result output phase, this paper can not only achieve verifiability of the deception, but also find the specific cheaters.

(3) This paper can resist intercept-resend attack, entangle-measure attack, Trojan horse attack, forgery attack, collusion attack, and malicious attack by the semi-honest third party TP.

The remaining structure of the protocol is as follows: In Section 2, some available basic knowledge are introduced. In Section 3, a verifiable (k, n)-threshold quantum secure multiparty summation protocol is designed. In Section 4, the correctness analysis and the security analysis are given respectively. In Section 5, we present the performance comparison between our protocol and related protocols. Finally, conclusion of this protocol is given.

2 Preliminaries

2.1 Shamir's Threshold Scheme

In 1979, Shamir [31] proposed a (k, n)-threshold secret sharing scheme based on the Lagrange interpolation formula. The details are as follows:

(1) Preparation phase

Let $GF(d)$ be a finite field (d is a large odd prime number); the shared secret is S; $n(< d)$ participants P_1, P_2, ..., P_n; a secret distributor D; any $k(\leq n)$ of n participants can reconstruct the secret S.

(2) Construction phase

Firstly, D independently chooses $k-1$ elements $\alpha_1, \alpha_2, ..., \alpha_{k-1} \in GF(d)$ and constructs a $(k-1)$-th polynomial as

$$f(x) = S + \alpha_1 x + \alpha_2 x^2 + ... + \alpha_{k-1} x^{k-1} (\text{mod} d), \tag{1}$$

where Eq. 1 satisfies $f(0) = S(\text{mod} d)$ and $\alpha_{k-1} \neq 0$.

Secondly, D chooses n different non-zero elements $x_1, x_2, ..., x_n \in GF(d)$, computes $y_i = f(x_i)$ for $i = 1, 2, ..., n$, and takes (x_i, y_i) as the secret share of the participant P_i.

Finally, D sends (x_i, y_i) to the corresponding participant P_i for $i = 1, 2, ..., n$.

(3) Reconstruction phase

Suppose that there are k participants who want to reconstruct the secret S together, then the following operations are performed:

Firstly, without loss of generality, assuming that the k participants are exactly P_1, P_2, ..., P_k, then the k secret shares (x_1, y_1), (x_2, y_2), ..., (x_k, y_k) can be obtained.

Secondly, the polynomial $f(x)$ can be reconstructed by using the Lagrange interpolation formula:

$$f(x) = \sum_{i=1}^{k} y_i \prod_{j=1, j\neq i}^{k} \frac{x - x_j}{x_i - x_j} (\text{mod} d). \tag{2}$$

Finally, the constant term $f(0) = S(\text{mod } d)$ of Eq. 2 can be obtained, i.e., the shared secret S is reconstructed.

2.2 *d*-dimensional GHZ State

In the d-dimensional Hilbert space, the X-basis and Z-basis are defined as follows [25]:

$$X = \{|J_j\rangle, j = 0, 1, \cdots, d-1\}, Z = \{|j\rangle, j = 0, 1, \cdots, d-1\}, \tag{3}$$

where $|J_j\rangle = \frac{1}{\sqrt{d}} \sum_{j=0}^{d-1} \omega^{jt} |t\rangle$, and $\omega = e^{\frac{2\pi i}{d}}$. Namely, the d-dimensional GHZ state of n-particles in the Z-basis can be expressed as

$$\Psi = \frac{1}{\sqrt{d}} \sum_{j=0}^{d-1} |j\rangle^{\bigotimes n}. \tag{4}$$

If each particle of quantum state Ψ is measured under the X-basis, and we use $v_1, v_2, \cdots, v_n \in \{0, 1, \cdots, d-1\}$ to denote the measurement results $|J_j\rangle (j \in \{0, 1, \cdots, d-1\})$ of the n particles, then we can get

$$v_1 + v_2 + \cdots + v_n = 0(\text{mod} d). \tag{5}$$

2.3 Hash Function

Hash function is a function of many for one, which takes information of different lengths as input and turns it into output of the same length. The process of generating hash value can be expressed as follows:

$$h = H(m), \tag{6}$$

where m is a string of different lengths that needs to be changed, $H(\cdot)$ denotes the hash function, and h represents a hash value of the fixed length. A secure hash function must have several properties as follows:

1) Rapidity: For any given message m, it is easy to calculate $H(m)$, that is, $H(m)$ is computable in polynomial time.
2) Unidirectionality: For any given hash value h, it is computationally impossible to find the message m satisfying $H(m) = h$.
3) Collision resistance: It is computationally infeasible to find two different messages m and m' so that $H(m) = H(m')$.

3 Our Scheme

This section consists of three main phases: (1) Preparation phase; (2) Construction phase; (3) Multiparty summation phase, including cheating identification phase and result output phase.

3.1 Preparation Phase

1) n participants: $P_1, P_2, ..., P_n$.
2) Assume that the unique identity information of the participant P_i is $x_i \in \{0, 1, ..., d-1\}$, where $i = 1, 2, ..., n$.
3) A semi-honest third-party TP is needed. TP will do his best to get the secrets of the participants while implementing the protocol. But he is not allowed to conspire with others [30].
4) Each participant needs to randomly select some decoy particles in the X-basis or Z-basis.
5) TP needs to publish a suitable hash function $H(\cdot)$.
6) The symbol "+" indicates the addition operation of modulo d.

3.2 Construction Phase

In this phase, TP and each participant have to do the following operations:

Step 1: Each participant $P_i (i = 1, 2, ..., n)$ randomly selects a key $C_i = (c_i^1, c_i^2, ..., c_i^n)$, and keeps c_i^i in his hand, where $c_i^1 + c_i^2 + ... + c_i^n = 0, c_i^r \in \{0, 1, ..., d-1\}, r = 1, 2, ..., n$. Then, the participant P_i sends the correct and valid c_i^j to the participant P_j through a secure quantum channel [26, 27], where $j = 1, 2, ..., i-1, i+1, ..., n$. Finally, the participant P_i has $(c_1^i, c_2^i, ..., c_n^i)$, where $i = 1, 2, ..., n$.

Step 2: Each participant $P_i (i = 1, 2, ..., n)$ generates a d-dimensional GHZ state $\psi_i = \frac{1}{\sqrt{d}} \sum_{j=0}^{d-1} |j\rangle^{\otimes 3}$ of 3-particles in the Z-basis, respectively. When each particle of the quantum state $\psi_i (i = 1, 2, ..., n)$ is measured in the X-basis, TP and the participant P_i agree to use $a_i^1, a_i^2, a_i^3 \in \{0, 1, ..., d-1\}$ denote the measurement results $|J_j\rangle$ of the 3 particles, where $j \in \{0, 1, ..., d-1\}$.

Step 3: The participant P_1 measures a particle of the quantum state ψ_1 in the X-basis, and assumes that the measurement result of this particle is $a_1^3 \in \{0, 1, ..., d-1\}$. Subsequently, he sets his secret to be $S_1 = (d - a_1^3) + t_1$, where $t_1 \in \{0, 1, ..., d-1\}$ is chosen by P_1.

Step 4: Similar to Step 2, each participant $P_i (i = 2, 3, ..., n)$ sets his secret to be $S_i = (d - a_i^3) + t_i$, where $t_i \in \{0, 1, ..., d-1\}$ is chosen by P_i.

Step 5: Each participant $P_i (i = 1, 2, ..., n)$ publishes an integer $T_i = t_i + E_i$, where $E_i = c_1^i + c_2^i + ... + c_n^i$.

Step 6: Each participant $P_i (i = 1, 2, ..., n)$ inserts other two particles of the quantum state ψ_i into the decoy particles prepared in advance, so that two particle sequences Q_i^1 and Q_i^2 are constructed. At the same time, P_i records the insertion positions and the initial states of the decoy particles in Q_i^1 and Q_i^2. Subsequently, Q_i^1 and Q_i^2 are sent to TP, where $i = 1, 2, ..., n$.

Step 7: After determining that TP receives $\{Q_i^1, Q_i^2\}$ for $i = 1, 2, ..., n$, the participant P_i announces the insertion positions and the measurement basis of the decoy particles. Subsequently, TP uses the announced insertion positions and measurement basis of the decoy particles to judge whether there is an eavesdropping attack during the transmission of the particle sequences $\{Q_i^1, Q_i^2\}$ for $i = 1, 2, ..., n$. Assuming there is no eavesdropping attack, TP can obtain other two particles of the quantum state ψ_i for $i \in \{1, 2, ..., n\}$; otherwise, TP terminates the protocol immediately and asks the participant $P_i (i \in \{1, 2, ..., n\})$ to resend the particle sequences $\{Q_i^1, Q_i^2\}$.

Step 8: Assuming that the eavesdropping checks all pass, TP will use the X-basis to measure the particles of the quantum state ψ_i for $i = 1, 2, ..., n$, which in turn yields the corresponding measurement results $a_i^1, a_i^2 (\in \{0, 1, ..., d-1\})$ and the pseudo secret $S_i{}^* = a_i^1 + a_i^2 + T_i$, where $T_i = t_i + E_i$ is a public value, $E_i = c_1^i + c_2^i + ... + c_n^i$. Then, TP can obtain the summation $S = \sum\limits_{i=1}^{n} S_i{}^* (\text{mod} d) = \sum\limits_{i=1}^{n} S_i (\text{mod} d)$. Finally, TP constructs a $(k-1)$-th polynomial based on the obtained S and the chosen $b_1, b_2, ..., b_{k-1} \in \{0, 1, ..., d-1\}$:

$$f(x) = S + b_1 x + ... + b_{k-2} x^{k-2} + b_{k-1} x^{k-1}, \tag{7}$$

where the coefficient of the highest order term of Eq. 7 needs to satisfy $b_{k-1} \neq 0$.

Step 9: TP generates (x_i, y_i) by using the unique identity information x_i of the participant P_i and Eq. 7, where $y_i = f(x_i)$, $i = 1, 2, ..., n$. Subsequently, (x_i, y_i) is sent to P_i as his secret share through a secure quantum channel [26, 27]. Meanwhile, TP computes and publishes $(x_i, H(y_i))$, where $H(y_i)$ denotes the hash value of y_i for $i = 1, 2, ..., n$.

3.3 Multiparty Summation Phase

3.3.1 Cheating Identification Phase

Step 1: Assuming that $k (\leq n)$ participants want to get the summation of all the secrets together, each participant P_i needs to provide his secret share $(x_i, y_i{}')$ to other participant P_j through a secure quantum channel [26, 27], where $i, j \in \{1, 2, ..., n\}$, and $i \neq j$. If the summation process can be smoothly carried out, the secret share $(x_i, y_i{}')$ provided by $P_i (i \in 1, 2, ..., n)$ needs to satisfy the following two conditions:

- x_i and $y_i{}'$ are linearly independent.
- $(x_i, H(y_i{}')) = (x_i, H(y_i))$, where $H(y_i{}')$ denotes the hash value of $y_i{}'$ and $(x_i, H(y_i))$ is the public information.

Step 2: From Step 1 above we know that if the participant $P_i (\in \{1, 2, ..., n\})$ is defined as a deceiver, he will be eliminated. Subsequently, the original qualified subsets are updated and the new participant is reselected together to reconstruct the summation $S = \sum\limits_{i=1}^{n} S_i{}^* (\text{mod} d) = \sum\limits_{i=1}^{n} S_i (\text{mod} d)$.

Step 3: Without loss of generality, assuming that the k participants mentioned in the above Step 1 are exactly $P_1, P_2, ..., P_k$, and they are all verified to be honest participants, the summation S can be obtained by executing Section 3.3.2.

3.3.2 Result Output Phase

Step 4: For participant $P_u (u \in \{1, 2, ..., k\})$, after receiving the secret shares sent by other honest participants $P_i (i = 1, 2, ..., k; i \neq u)$ (Step 3 has assumed that $P_1, P_2, ..., P_k$ are all honest participants), he can use the k secret shares (x_t, y_t) owned and the Lagrange interpolation formula to reconstruct the summation $S = \sum\limits_{i=1}^{n} S_i{}^* (\text{mod} d) = \sum\limits_{i=1}^{n} S_i (\text{mod} d)$, where $t = 1, 2, ..., k$. The specific output process is shown below:

$$S = \sum_{i=1}^{n} S_i{}^* = \sum_{i=1}^{n} S_i = f(0) = \sum_{i=1}^{k} y_i \left(\prod_{j=1, j \neq i}^{k} \frac{-x_j}{x_i - x_j} \right) (\text{mod} d). \tag{8}$$

4 Scheme Analysis

4.1 Correctness Analysis

Theorem 1 *If each particle of the Z-basis GHZ state* $\psi_i = \frac{1}{\sqrt{d}}\sum_{j=0}^{d-1}|j\rangle^{\otimes n}$ *is measured in the X-basis, and the measurement results* $|J_j\rangle (j \in \{0, 1, \cdots, d-1\})$ *of the n particle denotes the* $a_i^1, a_i^2, ..., a_i^n \in \{0, 1, ..., d-1\}$*, then these measurements satisfy* $a_i^1 + a_i^2 + ... + a_i^n = 0$ *for* $i = 1, 2, ..., n$.

Proof In Section 2.2, we know that the X-basis and Z-basis are defined as $X = \{\left|J_j\right\rangle, j = 0, 1, ..., d-1\}$ and $Z = \{|j\rangle, j = 0, 1, ..., d-1\}$, respectively, where $|J_j\rangle = \frac{1}{\sqrt{d}}\sum_{j=0}^{d-1}\omega^{jt}|t\rangle$, $\omega = e^{\frac{2\pi i}{d}}$. Therefore, if the quantum state ψ_i is measured in the X-basis and the measurement results are denoted as $a_i^1, a_i^2, ..., a_i^n \in \{0, 1, ..., d-1\}, i = 1, 2, ..., n$, then the quantum state ψ_i can be expressed by using the following equation:

$$\psi_i = \frac{1}{\sqrt{d}}\sum_{j=0}^{d-1}\left(\frac{1}{\sqrt{d}}\sum_{t=0}^{d-1}\omega^{jt}\,|t\rangle\right)^{\otimes n} = \frac{1}{(\sqrt{d})^{n+1}}\sum_{j=0}^{d-1}\omega^{j\sum_{m=1}^{n}a_i^m}\left|a_i^1a_i^2...a_i^n\right\rangle. \tag{9}$$

From Eq. 9, we can discuss the following two cases:

1) If $\sum_{m=1}^{n}a_i^m \neq 0$, then we can see

$$\begin{aligned}\sum_{j=0}^{d-1}\omega^{j\sum_{m=1}^{n}a_i^m} &= \sum_{j=0}^{d-1}(\omega^{\sum_{m=1}^{n}a_i^m})^j = 1 + (\omega^{\sum_{m=1}^{n}a_i^m})^1 + ... + (\omega^{\sum_{m=1}^{n}a_i^m})^{d-1}\\ &= \frac{1-(\omega^{\sum_{m=1}^{n}a_i^m})^d}{1-\omega^{\sum_{m=1}^{n}a_i^m}} = \frac{1-\omega^{d\sum_{m=1}^{n}a_i^m}}{1-\omega^{\sum_{m=1}^{n}a_i^m}} = \frac{1-(e^{\frac{2\pi i}{d}})^{d\sum_{m=1}^{n}a_i^m}}{1-(e^{\frac{2\pi i}{d}})^{\sum_{m=1}^{n}a_i^m}}\\ &= \frac{1-e^{2\pi i\sum_{m=1}^{n}a_i^m}}{1-e^{\frac{2\pi i}{d}\sum_{m=1}^{n}a_i^m}} = \frac{1-(e^{\pi i})^{2\sum_{m=1}^{n}a_i^m}}{1-e^{\frac{2\pi i}{d}\sum_{m=1}^{n}a_i^m}} = \frac{1-(-1)^{2\sum_{m=1}^{n}a_i^m}}{1-e^{\frac{2\pi i}{d}\sum_{m=1}^{n}a_i^m}}\\ &= \frac{1-1}{1-e^{\frac{2\pi i}{d}\sum_{m=1}^{n}a_i^m}} = 0,\end{aligned} \tag{10}$$

where $e^{\pi i} = -1$, and "i" in $e^{\pi i}$ is imaginary unit.

2) If $\sum_{m=1}^{n}a_i^m = 0$, then we can obtain

$$\sum_{j=0}^{d-1}\omega^{j\sum_{m=1}^{n}a_i^m} = \sum_{j=0}^{d-1}\omega^0 = d \tag{11}$$

Therefore, according to Eqs. 9–11, in the X-basis, we can know that the quantum state ψ_i can be represented as:

$$\psi_i = \frac{1}{(\sqrt{d})^{n-1}}\sum_{\substack{a_i^1+a_i^2+...+a_i^n=0,\\ a_i^1,a_i^2,...,a_i^n\in\{0,1,...,d-1\}}}\left|a_i^1a_i^2...a_i^n\right\rangle \tag{12}$$

To sum up, these measurement results $a_i^1, a_i^2, ..., a_i^n \in \{0, 1, ..., d-1\}$ can satisfy $a_i^1 + a_i^2 + ... + a_i^n = 0$ for $i = 1, 2, ..., n$. That is, Theorem 1 is proved.

Remark 1 In our protocol, the GHZ state $\psi_i = \frac{1}{\sqrt{d}} \sum_{j=0}^{d-1} |j\rangle^{\otimes 3}$ generated by P_i is a special case when it is $n = 3$. Namely, according to the proof of Theorem 1, the measurement results $a_i^1, a_i^2, a_i^3 \in \{0, 1, ..., d-1\}$ generated by P_i in this protocol can satisfy $a_i^1 + a_i^2 + a_i^3 = 0$ for $i = 1, 2, ..., n$.

□

Lemma 1 *In our protocol, the summation* $S = \sum_{i=1}^{n} S_i(\mathrm{mod}d) = \sum_{i=1}^{n} S_i{}^*(\mathrm{mod}d)$, *where* $S_i = (d-a_i^3)+t_i$, $S_i{}^* = a_i^1+a_i^2+T_i$, $T_i = t_i+E_i$, $E_i = c_1^i+c_2^i+...+c_n^i$, $c_i^i+c_i^2+...+c_i^n = 0$, $i = 1, 2, ..., n$.

Proof According to Theorem 1 and Remark 1, we can get

$$a_i^1 + a_i^2 + a_i^3 = 0, \tag{13}$$

where a_i^1 and a_i^2 are the measurement results owned by TP, and a_i^3 is the measurement result owned by P_i for $i = 1, 2, ..., n$.

Subsequently, based on Eq. 13, we can obtain

$$a_i^1 + a_i^2 = d - a_i^3. \tag{14}$$

Furthermore, we can get

$$\begin{aligned}\sum_{i=1}^{n} S_i{}^* &= \sum_{i=1}^{n} (a_i^1 + a_i^2 + T_i) = \sum_{i=1}^{n} (a_i^1 + a_i^2 + t_i + E_i) \\ &= \sum_{i=1}^{n} (a_i^1 + a_i^2 + t_i) + \sum_{i=1}^{n} E_i \\ &= \sum_{i=1}^{n} [(d - a_i^3) + t_i] + \sum_{i=1}^{n} \sum_{r=1}^{n} c_r^i \\ &= \sum_{i=1}^{n} S_i + 0 = S.\end{aligned} \tag{15}$$

Lemma 1 is proved. □

Proposition 1 *In the multiparty summation phase, the summation* $S = \sum_{i=1}^{n} S_i{}^*(\mathrm{mod}d) = \sum_{i=1}^{n} S_i(\mathrm{mod}d)$ *can be reconstructed if there are no less than k honest participants correctly executing this protocol.*

Proof In the multiparty summation phase, firstly, without loss of generality, assuming that the participants $P_1, P_2, ..., P_k$ want to reconstruct the summation $S = \sum_{i=1}^{n} S_i{}^*(\mathrm{mod}d) =$

$\sum_{i=1}^{n} S_i (\bmod d)$, then they need to obtain the k secret shares $\{(x_1, y_1'), (x_2, y_2'), ..., (x_k, y_k')\}$ through a secure quantum channel [26, 27]; secondly, for participant $P_i (i = 1, 2, ..., k)$, he can use the public information $(x_t, H(y_t))$ and the hash function $H(\cdot)$ to verify the honesty of other participants P_t for $t \in \{1, 2, ..., k\}$, and $i \neq t$. Assuming that the secret shares provided by $P_1, P_2, ..., P_{i-1}, P_{i+1}, ..., P_k$ are all verified to be correct and valid (see Section 3.3.1), the participant $P_i (i = 1, 2, ..., k)$ can obtain the k secret shares $\{(x_1, y_1), (x_2, y_2), ..., (x_k, y_k)\}$; finally, the participant $P_i (i = 1, 2, ..., k)$ can use the Lagrange interpolation formula and the secret shares $\{(x_1, y_1), (x_2, y_2), ..., (x_k, y_k)\}$ to reconstruct the summation S:

$$S = \sum_{i=1}^{n} S_i{}^* = \sum_{i=1}^{n} S_i = f(0) = \sum_{i=1}^{k} y_i \left(\prod_{j=1, j \neq i}^{k} \frac{-x_j}{x_i - x_j} \right) (\bmod d). \tag{16}$$

Proposition 1 is proved. □

4.2 Security Analysis

4.2.1 Intercept-Resend Attack

In our protocol, assume that the eavesdropper Eve intercepts the particles sent to TP by the participant $P_i (i \in \{1, 2, ..., n\})$, and then sends the forged particles to TP so that he can pass the eavesdropping check. This attack is not feasible. Because each participant $P_i (i = 1, 2, ..., n)$ sends the particle sequences $\{Q_i^1, Q_i^2\}$ to TP, and each particle sequence $Q_i^r (r \in \{1, 2\})$ is composed of one particle in ψ_i and some decoy particles, where each decoy particle is randomly selected in the X-basis or Z-basis. Since the eavesdropper Eve does not know the insertion positions, the initial states, and the measurement basis of the decoy particles, the probability of his attack failing is $P = 1 - (\frac{1}{2} \times \frac{d-1}{d})^q$, where q denotes the number of decoy particles. When q is large enough, $P = 1 - (\frac{d-1}{2d})^q$ converges to 1. Therefore, Eve's intercept-resend attack will be detected with a high probability.

4.2.2 Entangle-Measure Attack

In the process of particle transmission, the eavesdropper Eve uses the unitary operation U_E to entangle an auxiliary particle, and then steals the privacy information by measuring this auxiliary particle. We assume that this auxiliary particle is $|E\rangle$. In order to express entanglement-measurement attack more clearly, we analyze the decoy particles selected under different basis as follows:

1) If the decoy particles are randomly selected from the Z-basis, using the unitary operation U_E to act on the decoy particles can obtain:

$$U_E |0\rangle |E\rangle = \beta_{00} |0\rangle |e_{00}\rangle + \beta_{01} |1\rangle |e_{01}\rangle + \cdots + \beta_{0(d-1)} |d-1\rangle \left|e_{0(d-1)}\right\rangle$$

$$U_E |1\rangle |E\rangle = \beta_{10} |0\rangle |e_{10}\rangle + \beta_{11} |1\rangle |e_{11}\rangle + \cdots + \beta_{1(d-1)} |d-1\rangle \left|e_{1(d-1)}\right\rangle$$

$$\vdots$$

$$U_E |d-1\rangle |E\rangle = \beta_{(d-1)0} |0\rangle \left|e_{(d-1)0}\right\rangle + \beta_{(d-1)1} |1\rangle \left|e_{(d-1)1}\right\rangle + \cdots$$
$$+ \beta_{(d-1)(d-1)} |d-1\rangle \left|e_{(d-1)(d-1)}\right\rangle,$$

where these quantum states $|e_{ij}\rangle$ $(i, j \in \{0, 1, \cdots, d-1\})$ are determined by the unitary operation U_E, and

$$\begin{gathered} |\beta_{00}|^2 + |\beta_{01}|^2 + \cdots + \left|\beta_{0(d-1)}\right|^2 = 1 \\ |\beta_{10}|^2 + |\beta_{11}|^2 + \cdots + \left|\beta_{1(d-1)}\right|^2 = 1 \\ \vdots \\ \left|\beta_{(d-1)0}\right|^2 + \left|\beta_{(d-1)1}\right|^2 + \cdots + \left|\beta_{(d-1)(d-1)}\right|^2 = 1. \end{gathered}$$

In addition, in order to prevent eavesdropping attacks, the eavesdropper Eve must make the following provisions:

$$\begin{gathered} \beta_{01} = \beta_{02} = \cdots = \beta_{0(d-1)} = 0 \\ \beta_{10} = \beta_{12} = \cdots = \beta_{1(d-1)} = 0 \\ \vdots \\ \beta_{(d-1)0} = \beta_{(d-1)1} = \cdots = \beta_{(d-1)(d-2)} = 0. \end{gathered}$$

Therefore, we can simplify the above equations as follows:

$$\begin{gathered} U_E\,|0\rangle\,|E\rangle = \beta_0\,|0\rangle\,|e_0\rangle \\ U_E\,|1\rangle\,|E\rangle = \beta_1\,|1\rangle\,|e_1\rangle \\ \vdots \\ U_E\,|d-1\rangle\,|E\rangle = \beta_{d-1}\,|d-1\rangle\,|e_{d-1}\rangle, \end{gathered}$$

where $\beta_0 = \beta_{00}, \beta_1 = \beta_{11}, ..., \beta_{d-1} = \beta_{(d-1)(d-1)}$, and $e_0 = e_{00}, ..., e_{d-1} = e_{(d-1)(d-1)}$.

2) If these decoy particles are randomly selected from the X-basis, using the unitary operation U_E to act on the decoy particles can get:

$$U_E\left|J_j\right\rangle|E\rangle = U_E\left(\frac{1}{\sqrt{d}}\sum_{t=0}^{d-1}\omega^{jt}\,|t\rangle\right)|E\rangle = \frac{1}{\sqrt{d}}\sum_{t=0}^{d-1}\omega^{jt}U_E\,|t\rangle\,|E\rangle = \frac{1}{\sqrt{d}}\sum_{t=0}^{d-1}\omega^{jt}\beta_t\,|t\rangle\,|e_t\rangle\,, \tag{17}$$

where $j \in \{0, 1, \cdots, d-1\}$.

Furthermore, due to $\left|J_j\right\rangle = \frac{1}{\sqrt{d}}\sum\limits_{t=0}^{d-1}\omega^{jt}\,|t\rangle$, then we know $|j\rangle = \frac{1}{\sqrt{d}}\sum\limits_{t=0}^{d-1}\omega^{-jt}\,|J_t\rangle$ and can get

$$\begin{aligned} U_E\left|J_j\right\rangle|E\rangle &= \frac{1}{\sqrt{d}}\sum_{t=0}^{d-1}\omega^{jt}\beta_t\,|e_t\rangle\left(\frac{1}{\sqrt{d}}\sum_{i=0}^{d-1}\omega^{-it}\,|J_i\rangle\right) \\ &= \frac{1}{d}(|J_0\rangle\sum\nolimits_{t=0}^{d-1}\omega^{t(j-0)}\beta_t\,|e_t\rangle + |J_1\rangle\sum\nolimits_{t=0}^{d-1}\omega^{t(j-1)}\beta_t\,|e_t\rangle + ... \\ &\quad + |J_{d-1}\rangle\sum\nolimits_{t=0}^{d-1}\omega^{t(j-(d-1))}\beta_t\,|e_t\rangle) \end{aligned} \tag{18}$$

According to the above analysis, we can know that to avoid eavesdropping inspections, the eavesdropper Eve must set $\sum\limits_{t=0}^{d-1}\omega^{t(j-i)}\beta_t\,|e_t\rangle = 0$ for $i \in \{0, 1, \cdots, d-1\}$, $j \neq i$. There, for $\forall j \in \{0, 1, 2, ..., d-1\}$, $d-1$ equations can be obtained. Then, based on these equations, $\beta_0\,|e_0\rangle = \beta_1\,|e_1\rangle = \cdots = \beta_{d-1}\,|e_{d-1}\rangle$ is easily acquired. In a word, no matter what state the useful particles are in, Eve can only obtain the same information from the auxiliary particles, but cannot obtain the related privacy information. That is, the entanglement-measurement attack stopped.

4.2.3 Trojan Horse Attack

In the case that the particles used in this protocol are photons, two types of Trojan horse attacks [25, 29] as described below may exist: (1) Delayed photon attack; (2) Invisible photon attack. To resist these two attacks, we perform the following analysis:

- i) To resist the delayed photon attack, participants randomly select a portion of the received photon signal as the sample signal, and separate each sample signal using the PNS (Photon Number Separator) technique. Subsequently, they arbitrarily choose two signals in the X-basis or the Z-basis for measurement, and they can judge whether the particles need to be resent by the multi-photon rate. If the multi-photon rate is too high, the transmission of particles should be stopped and the particles should be resent.
- ii) To resist the invisible photon attack, participants can add a filter to the device. The filter is added to permit only photons with wavelengths close to the operating particles to enter and to isolate the invisible photons from the attackers.

4.2.4 Participant Attack

Proposition 2 *In our protocol, the insider attacks can be stopped.*

Proof In this paper, we consider the following two kinds of insider attacks: Forgery attack and Collusion attack.

- **Forgery attack:** In the multiparty summation phase, each participant exchanges the secret share (x_i, y_i) through a secure quantum channel [26,27], where $i \in \{1, 2, ..., n\}$. The deception is easily identified if the insider attacker $P_t(t \in \{1, 2, ..., n\})$ sends $(x_t, y_t{}')$ to the honest participant $P_i(i \in \{1, 2, ..., n\}; i \neq t)$, where $(x_t, y_t{}')$ is the forged secret share by P_t. The specific cheating identification is described as follows:

  Step 1: Honest participant P_i hashes the received $y_t{}'$ by using the publicly available hash function $H(\cdot)$ to obtain $H(y_t{}')$.

  Step 2: If Equation $(x_t, H(y_t{}')) = (x_t, H(y_t))$ does not hold, the deception of the participant P_t is identified; otherwise, is honest.
- **Collusion attack:** We analyze the following two collusion scenarios:

  I) The secret $S_i(i = 1, 2, ..., n)$ of each participant is safe under the collusion attack of $n-2$ secret holders. Without losing generality, we assume that the $n-2$ participants happen to be $P_1, P_2, ..., P_{n-2}$. The specific analysis is as follows:

  - 1) On the one hand, the participants $P_1, P_2, ..., P_{n-2}$ have no less than k secret shares through collusion. That is, the summation $S = \sum_{i=1}^{n} S_i(\text{mod}d)$ can be reconstructed by using the Lagrange interpolation formula. On the other hand, the participants $P_1, P_2, ..., P_{n-2}$ conspire to own secrets $S_1, S_2, ..., S_{n-2}$, and then by combining the reconstructed S, they can get $S_{n-1} + S_n$. That is, they can't decrypt S_{n-1} and S_n.
  - 2) On the one hand, TP is a semi-honest third party, so he will not conspire with $P_1, P_2, ..., P_{n-2}$ to make them get $a_i^1 + a_i^2, i = n-1, n$. That is, S_{n-1} and S_n are safe. On the other hand, according to the analysis in Section 4.2.1, we can know that the intercept-resend attack can be prevented, that is, S_{n-1} and S_n are safe.

II) For the security of the summation S, we consider the worst case, that is, there are $k-1$ insider participants who launch a collusion attack. Without losing generality, we assume that the $k-1$ participants happen to be $P_1, P_2, ..., P_{k-1}$. The specific analysis is as follows:

- 1) The third-party TP needed in this protocol is semi-honest, so he will not reveal $a_i^1 + a_i^2$ of each participant voluntarily, where $i = 1, 2, ..., n$. That is to say, i) the participants $P_1, P_2, ..., P_{k-1}$ can't get the secrets $S_k, S_{k+1}, ..., S_{i-1}, S_i, S_{i+1}, ..., S_n$ and the summation $S = \sum_{i=1}^{n} S_i (\text{mod } d)$, where $S_i = a_i^1 + a_i^2 + t_i$ for $i = k, k+1, ..., n$, $t_i = T_i - E_i$, and T_i is a public value; ii) the participants $P_1, P_2, ..., P_{k-1}$ can't get the pseudo secrets $S_k^*, S_{k+1}^*, ..., S_{i-1}^*, S_i^*, S_{i+1}^*, ..., S_n^*$ and the summation $S = \sum_{i=1}^{n} S_i{}^* (\text{mod} d) = \sum_{i=1}^{n} S_i (\text{mod} d)$, where $S_i^* = a_i^1 + a_i^2 + T_i$ for $i = k, k+1, ..., n$.
- 2) The participants $P_1, P_2, ..., P_{k-1}$ want to get the k-th secret share (x_t, y_t) by breaking the public information $(x_t, H(y_t))$, and then use the secret shares $(x_1, y_1), (x_2, y_2), ..., (x_{k-1}, y_{k-1}), (x_t, y_t)$ and the Lagrange interpolation formula to reconstruct the summation $S = \sum_{i=1}^{n} S_i = f(0) = \sum_{i=1}^{k} y_i \left(\prod_{j=1, j \neq i}^{k} \frac{-x_j}{x_i - x_j} \right) (\text{mod } d)$, where $t \in \{k, k+1, ..., n\}$. This is not feasible and the detailed analysis can be found in Section 5 of protocol [28].
- 3) According to the analysis in Section 4.2.1, we can know that the intercept-resend attack doesn't work. Therefore, $a_i^1 + a_i^2$ of each participant is safe, where $i = 1, 2, ..., n$. That is to say, S_i, S_i^*, and $S = \sum_{i=1}^{n} S_i{}^* (\text{mod} d) = \sum_{i=1}^{n} S_i (\text{ mod } d)$ cannot be obtained by $P_1, P_2, ..., P_{k-1}$. The specific analysis is similar to that Item "II),1)" of Collusion attack.

Proposition 2 is proved.

Remark 2 This protocol can't resist the collusion attack from $n-1$ secret holders, because they can easily decrypt the last secret from the summation $S = \sum_{i=1}^{n} S_i (\text{mod} d)$.

□

Proposition 3 *In our protocol, the external attacks can be resisted.*

Proof If the external attackers want to obtain the secret $S_i (i = 1, 2, ..., n)$ of each participant or the summation $S = \sum_{i=1}^{n} S_i{}^* (\text{mod} d) = \sum_{i=1}^{n} S_i (\text{mod} d)$, they can only try to get them in the following three ways:

- 1) The external attackers want to obtain the secret shares (x_i, y_i) of the participants through the public information $(x_i, H(y_i))$ for $i = 1, 2, ..., n$. Obviously, this is not feasible. The specific analysis can be found in Section 5 of protocol [28].

Table 1 Comparison of basic properties

	Shi and Zhang [17]	Liu et al. [19]	Yang and Ye [20]	Lv et al. [21]	Song et al. [23]	Sutradhar and Om [24]	Ours
Threshold	$(2, 2)$	(n, n)	(n, n)	(n, n)	(k, n)	(k, n)	(k, n)
Dimension of space	–	2	d	d^u	d	d	d
Verifiability	No	No	No	No	No	No	Yes
Collusion attack	—	—	Yes	Yes	Yes	Yes	Yes
Trojan horse attack	—	Yes	—	—	—	—	Yes

- 2) In the construction phase, the external attackers want to obtain the measurement results a_i^1, a_i^2 by intercepting the particle sequences $\{Q_i^1, Q_i^2\}$ for $i \in \{1, 2, ..., n\}$, and then get $S_i{}^* = a_i^1 + a_i^2 + T_i$ and $S_i = a_i^1 + a_i^2 + t_i$, where $t_i = T_i - E_i$, T_i is a public value, $E_i = c_1^i + c_2^i + ... + c_n^i$. Obviously, this is not feasible either. The reason can be known in the analysis of Section 4.2.1 and Item "II),1)" of Collusion attack.
- 3) In the multiparty summation phase, each participant exchanges the secret share though a secure quantum channel [26, 27]. That is, the external attackers cannot intercept the secret share of each participant through this way. Therefore, the summation S will not be stolen.

Proposition 3 is proved. □

Proposition 4 *In our protocol, the malicious attack by the semi-honest third party TP can be prevented.*

Proof TP can obtain $a_i^1 + a_i^2$ by measuring the received particles, where $i = 1, 2, ..., n$. If he wants to get the secret $S_i = a_i^1 + a_i^2 + t_i$ of each participant, he also needs to know E_i, where $T_i = E_i + t_i$ is a public value. Obviously, this is not feasible because he can't know E_i.

Proposition 4 is proved. □

5 Performance Analysis

In this paper, a new verifiable (k, n)-threshold quantum secure multiparty summation protocol is proposed. Compared with the existing protocols, the results are as follows:

Under the condition that the quantum environment is a free space, our protocol and protocols [20, 21, 23, 24] are more adaptable in this respect than the 2-dimensional quantum secure multiparty summation [19].

Protocols [17, 19–21] respectively construct a (n, n)-threshold quantum secure multiparty summation protocol. That is, if participants of these protocols want to get the output information, they must require all participants to be online. Therefore, compared with protocols [17, 19–21], our protocol and protocols [23, 24] are more flexible.

Although protocols [20, 21, 23, 24] considers both external attack and collusion attack, that is, neither external attackers nor internal attackers can obtain the input information of the corresponding participants. However, in the result output phase, if there are malicious internal participants, they may provide wrong information or measurement results to other honest participants, so that they can get correct output information, while other honest participants can't. In addition, the analysis shows that protocol [23] can determine the existence

Table 2 Comparison of computation costs

	Shi and Zhang [17]	Liu et al. [19]	Yang and Ye [20]	Lv et al. [21]	Song et al. [23]	Sutradhar and Om [24]	Ours
QFT	—	—	n	—	—	k	—
QFT^{-1}	—	—	—	—	—	—	—
Measure operation	—	nL	nL	nL	1	k	$3n$
Unitary operation	—	nL	nL	nL	$n+1$	k	—
Hash operation	—	—	—	—	—	—	n
Classical information operation	—	—	—	—	$n+k$	$3n$	$4n$

QFT: The quantum Fourier transform; QFT^{-1}: The quantum inverse Fourier transform; n: The number of participants; k: The threshold value, and $k \leq n$; L: The length of each participant's secret

of cheating behavior by judging whether the relevant equation is true or not, but can't find out the specific cheaters. Therefore, our protocol has more advantages in this respect.

According to the analysis of Section 4.2.3, we know that if the particles used in protocols are photons, there may be Trojan horse attack. Therefore, our protocol is more secure than protocols [20, 21] which fail to consider Trojan horse attack.

To analyze the efficiency of different protocols, we compare the computation and communication costs between our protocol and protocols [19–21, 23, 24]. The computation cost can be considered based on the following six parts: QFT, QFT^{-1}, measure operation, unitary operation, hash operation, and classical information operation. And the communication cost can be considered based on message particles, decoy particles, and classic information. In protocol [19], nL measure operation + nL unitary operation and nL message particles + $q(n-1)$ decoy particles are the total computation cost and the total communication cost, respectively. In protocol [20], n QFT + nL measure operation + nL unitary operation and nL message particles + $q(n-1)$ decoy particles are the total computation cost and the total communication cost, respectively. In protocol [21], nL measure operation + nL unitary operation and L message particles + qn decoy particles are the total computation cost and the total communication cost, respectively. In protocol [23], 1 measure operation + $(n+1)$ unitary operation + $(n+k)$ classical information operation and 1 message particles + n classical information are the total computation cost and the total communication cost, respectively. In protocol [24], k QFT + k measure operation + k unitary operation +

Table 3 Comparison of communication costs

	Shi and Zhang [17]	Liu et al. [19]	Yang and Ye [20]	Lv et al. [21]	Song et al. [23]	Sutradhar and Om [24]	Ours
Number of message particles	—	nL	nL	L	1	$k-1$	$3n$
Number of decoy particles	—	$q(n-1)$	$q(n-1)$	qn	—	—	$2qn$
Number of classic information	—	—	—	—	n	$n(n-1)$	$n(n-1)$

q: The number of decoy particles

$3n$ classical information operation and $(k-1)$ message particles + $n(n-1)$ classical information are the total computation cost and the total communication cost, respectively. In our protocol, $3n$ measure operation + n hash operation $4n$ classical information operation and $3n$ message particles + $2qn$ decoy particles + $n(n-1)$ classical information are the total computation cost and the total communication cost, respectively. In short, on the one hand, compared with the (n,n)-threshold protocols [19–21], our protocol requires less computation cost when L satisfies $L > 4$. But compared with the (k,n)-threshold protocol [23, 24], we need more computation cost. The reason why is that this paper uses more message particles and classic information to achieve secure multiparty summation, and also uses the hash function to achieve verifiability; on the other hand, we can find that there is no advantage in communication cost between our protocol and protocols [19–21, 23, 24]. This is because our protocol uses decoy particles to prevent eavesdropping attacks during particle transmission, and also uses classical key information to ensure that the secret of each participant is not known by others.

To understand the performance of related protocols more clearly, we can see Tables 1, 2 and 3.

6 Conclusion

Based on Shamir's threshold scheme and d-dimensional GHZ state, we proposed a new (k,n)-threshold quantum secure multiparty summation protocol. Based on the one-to-one correspondence of hash values, in the result output phase, the honesty of each participant can be verified, and dishonest participants can be eliminated. In addition, compared with the (n,n)-threshold protocols [19–21], our protocol needs less computation cost when L satisfies $L > 4$, where L is the length of each participant's secret. Further, the security analysis shows that this paper can resist a series of typical attacks.

This paper put forward a new quantum secure multiparty summation protocol. It has a (k,n)-threshold approach and is verifiable, but it has no clear advantage in computation and communication costs. We hope to propose a more efficient verifiable (k,n)-threshold quantum secure multiparty summation protocol in the future.

Acknowledgements The authors would like to thank the National Natural Science Foundation of China (No.U21A20428, No.61972126, No.12171134) for supporting this research.

Author Contributions Conceptualization, L.F.L., H.H., Z.S.X. and L.P.; formal analysis, L.F.L., H.H., Z.S.X. and L.P.; investigation, L.F.L. and H.H.; methodology, L.F.L. and H.H.; validation, L.F.L., H.H., Z.S.X. and L.P.; writing-original draft, L.F.L. and H.H.; writing-review & editing, L.F.L. and H.H.

Funding This work was supported by the National Natural Science Foundation of China (No.U21A20428, No.61972126, No.12171134).

Data Availability All data generated or analysed during this study are included in this published article.

Declarations

Consent for Publication All authors have read and agreed to the published version of the manuscript.

Competing interests The authors declare that there is no conflict of interest regarding the publication of this manuscript.

References

1. Yao, A.C.: Protocols for secure computations. In: 23rd IEEE Symposium on Foundations of Computer Science, pp. 160-164 (1982)
2. Hamada, K., Kikuchi, R., Ikarashi, D., Chida, K., Takahashi, K.: Practically efficient multi-party sorting protocols from comparison sort algorithms. In: Information Security and cryptology-ICISC 2012. 15th International Conference, pp. 202–216 (2013)
3. Laud, P., Pettai, M.: Secure multiparty sorting protocols with covert privacy. Secure IT systems NordSec 2016 **10014**, 216–231 (2016)
4. Maheshwari, N., Kiyawat, K.: Structural framing of protocol for secure multiparty cloud computation. In: 2011 Fifth Asia Modelling Symposium IEEE, pp. 187–192 (2011)
5. Lopez-Alt, A., Tromer, E., Vaikuntanathan, V.: On-the-fly Multiparty Computation on the Cloud via Multikey Fully Homomorphic Encryption. In: Proceedings of the 2012 ACM Symposium on Theory of Computing, pp. 1219–1234 (2012)
6. Jung, T., Li, X.Y., Wan, M.: Collusion-tolerable privacy-preserving sum and product calculation without secure channel. IEEE Trans. Dependable Secure Comput. **12**(1), 45–57 (2015)
7. Mehnaz, S., Bellala, G., Bertino, E.: A secure sum protocol and its application to privacy-preserving multiparty analytics. In: Proceedings of the 22nd ACM on Symposium on Access Control Models and Technologies, pp. 219–230. ACM (2017)
8. Ashouri, T.M., Baraani, D.A.: Cryptographic collusion-resistant protocols for secure sum. Int. J. Electron. Secur. Digit. Forensic. **9**(1), 19–34 (2017)
9. Kantarcioglu, M., Clifton, C.: Privacy-preserving distributed mining of association rules on horizontally partitioned data. IEEE Trans. Knowl. Data Eng. **16**(9), 1026–1037 (2004)
10. Shor, P.W.: Algorithms for quantum computation: discrete logarithms and factoring. In: Proceedings of the 35th Annual Symposium of Foundation of Computer Science, pp. 124–134 (1994)
11. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Proceedings of the Twenty-Eighth Annual ACM Symposium Computing, pp. 212–219. ACM (1996)
12. Crépeau, C., Gottesman, D., Smith, A.: Secure multi-party quantum computation. In: Proceedings of the 34th Annual ACM Symposium on Theory of Computing, pp. 643–652 (2002)
13. Chen, X.B., Xu, G., Yang, Y.X., Wen, Q.Y.: An efficient protocol for the secure multi-party quantum summation. Int. J. Theor. Phys. **49**(11), 2793–2804 (2010)
14. Wang, Q.L., Sun, H.X., Huang, W.: Multi-party quantum private comparison protocol with n-level entangled states. Quant. Inf. Process. **13**(11), 2375–2389 (2014)
15. Liu, B., Zhang, M.W., Shi, R.H.: Quantum secure multi-party private set intersection cardinality. Int. J. Theor. Phys. **59**(7), 1992–2007 (2020)
16. Dou, Z., Chen, X.B., Xu, G., Liu, W., Yang Y.X., Yang, Y.: An attempt at universal quantum secure multi-party computation with graph state. Phys. Scr. **95**(5), 055106 (2020)
17. Shi, R.H., Zhang, S.: Quantum solution to a class of two-party private summation problems. Quantum Inf. Process. **16**(9), 225 (2017)
18. Ye, T.Y., Xu, T.J.: A lightweight three-user secure quantum summation protocol without a third party based on single-particle states. Quant. Inf. Process. **21**(9), 309 (2022)
19. Liu, W., Wang, Y.B., Fan, W.Q.: An novel protocol for the quantum secure multi-party summation based on two-particle bell states. Int. J. Theor. Phys. **56**(9), 2783–2791 (2017)
20. Yang, H.Y., Ye, T.Y.: Secure multi-party quantum summation based on quantum Fourier transform. Quant. Inf. Process. **17**(6), 129 (2018)
21. Lv, S.X., Jiao, X.F., Zhou, P.: Multiparty quantum computation for summation and multiplication with mutually unbiased bases. Int. J. Theor. Phys. **58**, 2872–2882 (2019)
22. Wang, Y.L., Hu, P.C., Xu, Q.L.: Quantum secure multi-party summation based on entanglement swapping. Quant. Inf. Process. **20**(10), 319 (2021)
23. Song, X., Gou, R., Wen, A.: Secure multiparty quantum computation based on Lagrange unitary operator. Sci. Rep. **10**, 7921 (2020)
24. Sutradhar, K., Om, H.: A generalized quantum protocol for secure multiparty summation. IEEE Trans. Circuits Syst. II Express Briefs **67**(12), 2978–2982 (2020)
25. Qin, H., Dai, Y.: Dynamic quantum secret sharing by using d-dimensional GHZ state. Quantum Inf. Process. **16**(3), 64 (2017)
26. Cai, Q.Y., Li, W.B.: Deterministic secure communication without using entanglement. Chin. Phys. Lett. **21**, 601–603 (2004)
27. Deng, F.G., Long, G.L.: Secure direct communication with a quantum one-time pad. Phys. Rev. A. **69**(2004), 98–106 (2016)

28. Wang, P., Zhang, R., Sun, Z.W.: Practical quantum key agreement protocol based on BB84. Quant. Inf. Comput. **22**(3-4), 241–250 (2022)
29. Li, Y.B., Qin, S.J., Yuan, Z., Huang, W., Sun, Y.: Quantum private comparison against decoherence noise. Quant. Inf. Process. **12**(6), 2191–2205 (2013)
30. Yang, Y.G., Xia, J., Jia, X., Zhang, H.: Comment on quantum private comparison protocols with a semi-honest third party. Quantum Inf. Process. **12**(2), 877–885 (2013)
31. Shamir, A.: How to share a secret. Commun. ACM **22**(11), 612–613 (1979)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.