Article | Received 16 June 2026; Revised 14 August 2026; Accepted 2 September 2026; Published 21 September 2026
https://doi.org/10.55092/qr20260004

Circuit-model blind quantum computation with key decoupling

Ting Xiang[1],†, Bingwen Feng[2],† and Xiaoqian Zhang[3],*

[1] School of Physics and State Key Laboratory of Optoelectronic Materials and Technologies, Sun Yat-sen University, Guangzhou, China

[2] College of Cyber Security, Jinan University, Guangzhou, China

[3] College of Information Science and Technology, Jinan University, Guangzhou, China

† These authors contributed equally to this work.

* Correspondence author; E-mail: zhangxq64@jnu.edu.cn.

Highlights:

- Key-decoupled circuit-model blind quantum computation is proposed.
- Independent encryption and decryption eliminate gate-by-gate key propagation.
- Target operations are blind within their Pauli equivalence classes.
- Random Pauli sampling enables low-overhead statistical verification.
- Proof-of-principle CMBQC is demonstrated on an IBM superconducting processor.

Abstract: In the noisy intermediate-scale quantum (NISQ) era, quantum computing resources are increasingly provided through cloud platforms, enabling users to perform quantum computations without owning expensive hardware. However, delegating computations to remote servers raises a key challenge: ensuring the privacy of the client's input, algorithm, and output. To address this issue, we propose a circuit-model blind quantum computation (CMBQC) protocol that conceals the target quantum computation while decoupling the encryption and decryption keys. This key-decoupled structure avoids gate-by-gate propagation of decryption information on the client side. We establish blindness of the target operation within its Pauli equivalence class and prove the verifiability of the protocol, where verification is achieved by estimating expectation values of randomly chosen Pauli observables, thereby substantially reducing the verification overhead. We further demonstrate a single-qubit implementation of the CMBQC protocol on an International Business Machines Corporation (IBM) superconducting quantum system. The simple structure and low verification cost of the CMBQC protocol make it a promising framework for secure delegated quantum computation, with potential applications in quantum cloud computing.

Keywords: blind quantum computation; statistical verification; pauli observables; quantum one-time pad; superconducting quantum processor

1. Introduction

The emergence of cloud-accessible quantum processors marks a significant milestone in quantum computing, with universal devices predominantly hosted by a limited number of technologically advanced institutions. In this setting, clients with limited quantum capabilities must delegate nontrivial quantum computations to remote servers. Consequently, preserving the privacy of the client's quantum inputs, the target algorithm, and the computation outcomes becomes essential. Blind quantum computation (BQC) addresses this requirement by enabling secure delegated quantum computation while preserving all meaningful information about the computation from the server. In recent years, BQC has been actively investigated, with its development primarily centered on measurement-based (MBQC) [1–5] and circuit-based (CBQC) [6–8] protocols.

MBQC protocols [1–5] provide rigorous blindness, fully concealing the client's input quantum states, target algorithm, and computation outcomes from the server. However, each computation consumes a large graph state, resulting in significant quantum resource overhead. By contrast, the circuit model aligns naturally with superconducting quantum systems, including International Business Machines Corporation's (IBM's) superconducting quantum processors, the 287-qubit Tianyan platform, the 105-qubit Zuchongzhi 3 processor, and the Benyuan "Wukong" superconducting platform. In this model, computations are specified by parameterized quantum circuits that support reuse and modular composition, rendering circuit-based approaches particularly well suited for scalable BQC.

Andrew M. Childs [6] first proposed a circuit-based secure delegated quantum computation scheme in 2005, which encrypts the client's input quantum states while preserving the correctness of the computation. Anne Broadbent *et al.* [7] subsequently enabled entanglement-assisted CBQC for the non-Clifford T gate with only a single round of interaction. Building on these results, Kevin Fisher *et al.* [8] experimentally demonstrated CBQC. However, CBQC protocols typically leave the executed quantum gates visible to the server while imposing a gate-by-gate coupling between encryption and decryption. As quantum cloud computing advances toward increasingly complex algorithms and deeper circuits, practical security demands extend beyond quantum-state blindness, rendering the privacy of the target quantum computation central to BQC. This gate-by-gate structure causes decryption-related correlations to accumulate with circuit depth, substantially increasing the client's decryption burden and limiting scalability. More specifically, if k_i denotes the key information after the i-th delegated gate G_i, this dependence can be expressed schematically as $k_i = \mathscr{F}_{G_i}(k_{i-1})$. For a circuit $U = G_L \cdots G_1$, the final decryption key therefore takes the recursive form $k_{\mathrm{dec}} = \mathscr{F}_{G_L} \circ \cdots \circ \mathscr{F}_{G_1}(k_{\mathrm{enc}})$, illustrating its dependence on both the initial encryption key and the entire gate sequence. Consequently, a central goal of CBQC is to conceal the quantum algorithm while preserving correctness, without introducing a decryption overhead that scales with circuit depth.

Here, we propose a circuit-model blind quantum computation (CMBQC) protocol that eliminates the need for gate-by-gate decryption. In our CMBQC protocol, the encryption and decryption keys, k_{enc} and k_{dec}, are independently generated, and their effects are incorporated directly into the blinded operation V through $D(k_{\mathrm{dec}})VE(k_{\mathrm{enc}}) = U$, or equivalently $V = D^{\dagger}(k_{\mathrm{dec}})UE^{\dagger}(k_{\mathrm{enc}})$. Therefore, no gate-by-gate propagation of the decryption key is required on the client side, substantially reducing the

client's operational burden and facilitating experimental implementation. We also rigorously prove the blindness of the target operation within its Pauli equivalence class and the verifiability of the protocol, with verification requiring only the evaluation of randomly chosen Pauli observables. We further demonstrate the CMBQC protocol experimentally in the single-qubit setting on an IBM superconducting quantum system. The proposed protocol has potential applications in secure quantum cloud computing, benefiting from its simple structure, low verification cost, and favorable scalability.

2. The CMBQC protocol

We now present the proposed CMBQC protocol, as illustrated in Figure 1, which involves two parties, a client and a server. To implement a target quantum computation U on a quantum state $|\psi\rangle$, the client is responsible for state preparation, encryption, and decryption, while the server performs the quantum computation. The CMBQC protocol consists of the following three steps.

(i) Encryption of the input quantum state. For an n-qubit input state $|\psi\rangle$, the client randomly generates encryption and decryption keys $(\mathbf{a},\mathbf{b})$ and $(\mathbf{a}',\mathbf{b}')$, respectively, where $\mathbf{a},\mathbf{b},\mathbf{a}',\mathbf{b}' \in \{0,1\}^n$. The client applies single-qubit Pauli operations determined by the encryption key $(\mathbf{a},\mathbf{b})$, $E(k_{\text{enc}}) = \bigotimes_{i=1}^{n} X_i^{a_i} Z_i^{b_i}$, to encrypt the input state, yielding $E(k_{\text{enc}})|\psi\rangle$, which is information-theoretically hidden and then transmitted to the server for subsequent computation.

(ii) Implementation of BQC. The client derives a blinded quantum computation V from the target quantum algorithm U and the associated keys, revealing to the server only its mathematical description. The server then executes V on the encrypted quantum state using its quantum computational resources and returns the resulting state to the client.

(iii) Decryption of the output quantum state. The client applies the Pauli decryption operator $D(k_{\text{dec}})$ to the state returned by the server, thereby recovering the desired output state $U|\psi\rangle$. Accordingly, the procedure satisfies $D(k_{\text{dec}})VE(k_{\text{enc}})|\psi\rangle = U|\psi\rangle$.

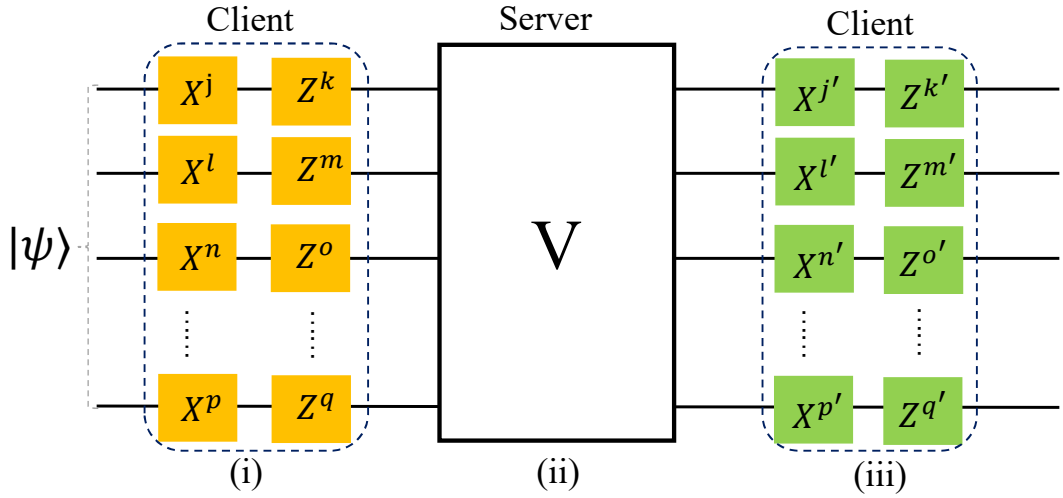

Figure 1. Schematic of the circuit-model universal blind quantum computation protocol. The protocol consists of three parts: the client encrypts the input quantum state using random Pauli operators $X^jZ^k, X^lZ^m, \ldots, X^pZ^q$; the server executes the expected quantum computation V on the encrypted state; and the client decrypts the output state using independently generated Pauli operators $X^{j'}Z^{k'}, X^{l'}Z^{m'}, \ldots, X^{p'}Z^{q'}$.

Following steps (i)–(iii), the protocol proceeds through multiple interaction rounds, only one of which corresponds to the target quantum computation, while the others serve for verification. The inability of the server to distinguish the function of individual rounds precludes selective deviation during the computation round. During verification, the client randomly measures Pauli observables on the returned quantum states and compares the outcomes with the corresponding theoretical values.

The above construction is formulated for a general n-qubit blind quantum computation. In particular, the Pauli-based encryption and decryption, the key-dependent construction of the blinded computation V, and the statistical verification based on Pauli observables are not restricted to the single-qubit setting and naturally extend to multi-qubit systems through tensor-product Pauli operations. The single-qubit implementation presented below therefore serves as a proof-of-principle demonstration of the protocol workflow and verification strategy on currently available quantum hardware. A practical distinction arises in the verification procedure: while the single-qubit key space is sufficiently small to permit exhaustive enumeration, the key space grows exponentially with the number of qubits, so multi-qubit implementations rely on random sampling of keys rather than exhaustive verification.

3. Blindness proof and verifiability analysis

To establish the blindness of the CMBQC protocol and analyze its verifiability, we specify the associated Pauli structure and the verification criteria. Since a single experimental outcome provides insufficient information about the server's behavior, its honesty must be evaluated statistically over multiple interaction rounds. Motivated by this observation, we introduce two verification criteria to assess compliance with the protocol.

Definition 1 (Expectation-Value Check): In a given verification, the client independently and uniformly samples a Pauli observable O from $\{X,Y,Z\}$, measures the returned quantum state to obtain $\langle O\rangle_{\text{exp}}$, and compares it with the theoretical value $\langle O\rangle_{\text{th}}$ for the target computation. The server is accepted as honest and the computation as correct whenever

$$\left|\langle O\rangle_{\text{exp}} - \langle O\rangle_{\text{th}}\right| \leq \varepsilon, \tag{1}$$

with $\varepsilon > 0$ denoting a predefined tolerance threshold.

We next extend Definition 1 to a statistical verification criterion for multiple interaction rounds.

Definition 2 (Acceptance Probability): Let N be the total number of verification rounds and m_{pass} the number of rounds satisfying Definition 1. The acceptance probability is defined as

$$P = \frac{m_{\text{pass}}}{N}. \tag{2}$$

If $P \geq \delta$, where δ is a predefined acceptance threshold, the computation outcome is accepted; otherwise, the server is regarded as dishonest.

Here we provide an algebraic description [9,10] of the Pauli encryption and decryption structure underlying the protocol, which enables a blindness analysis independent of the explicit choice of encryption and decryption operators.

(1) n-qubit Pauli group: Let $\mathscr{P}_1 = \{I,X,Z,XZ\}$ denote the set of single-qubit Pauli operators, and define $\mathscr{P}_n = \{P_1 \otimes P_2 \otimes \cdots \otimes P_n : P_i \in \mathscr{P}_1\}$. Equipped with matrix multiplication, $\mathscr{P}_n$ forms a finite

group, known as the n-qubit Pauli group.

(2) n-qubit Pauli left-right group and group action: We define the n-qubit Pauli left-right group as

$$G_n := \mathscr{P}_n \times \mathscr{P}_n. \tag{3}$$

An element $g \in G_n$ is written as $g = (P_L, P_R)$, where $P_L, P_R \in \mathscr{P}_n$ act on the left and right, respectively. For an arbitrary quantum operation V acting on the n-qubit Hilbert space $\mathscr{H}_n$, we define an action of G_n on V by

$$g \cdot V := P_L V P_R, \tag{4}$$

where P_R and P_L correspond operationally to the Pauli encryption and decryption operations in the protocol.

(3) Pauli equivalence classes: For an arbitrary operator $V \in \mathscr{H}_n$, its orbit under the action of G_n is defined as

$$\mathscr{O}(V) := \{g \cdot V \mid g \in G_n\} = \{P_L V P_R \mid P_L, P_R \in \mathscr{P}_n\}, \tag{5}$$

which is called the Pauli equivalence class of V. Equivalently, two operators $V_0, V_1 \in \mathscr{H}_n$ belong to the same Pauli equivalence class if and only if there exists a pair $(P_L, P_R) \in G_n$ such that

$$V_1 = P_L V_0 P_R. \tag{6}$$

(4) Stabilizer: For a given operator $V \in \mathscr{H}_n$, its stabilizer under the action of the Pauli left-right group is defined as

$$\mathrm{Stab}(V) := \{(P_L, P_R) \in G_n \mid P_L V P_R = e^{i\theta} V \text{ for some } \theta \in \mathbb{R}\}. \tag{7}$$

Thus, $\mathrm{Stab}(V)$ contains precisely those elements of G_n that leave V invariant up to a global phase.

3.1. Proof of blindness

The quantum one-time pad ensures that the server learns nothing about the input state, output state, or secret keys. We now present a proof of the blindness of the target quantum algorithm.

Theorem (Uniform Distribution over Pauli Orbits). Let g be sampled uniformly from G_n and define $U := g \cdot V$. Given $V = v$, the random variable U satisfies

$$\Pr[U = u \mid V = v] = \begin{cases} \dfrac{|\mathrm{Stab}(v)|}{|G_n|}, & u \in \mathscr{O}(v), \\ 0, & u \notin \mathscr{O}(v), \end{cases} \tag{8}$$

where $\mathscr{O}(v)$ and $\mathrm{Stab}(v)$ denote the Pauli orbit and stabilizer of v, respectively.

As a result, the conditional distribution $\Pr[U = u \mid V = v]$ is uniform over the orbit $\mathscr{O}(v)$. Therefore, the server cannot distinguish which specific representative within the same Pauli equivalence class corresponds to the target operation, although the Pauli-orbit membership itself may remain distinguishable.

Proof. Fix $V = v$ and let $U := g \cdot v$. For any u, define $K_{v,u} := \{g \in G_n \mid g \cdot v = u\}$. Uniform sampling of g from G_n implies

$$\Pr[U = u \mid V = v] = \frac{|K_{v,u}|}{|G_n|}. \tag{9}$$

We consider the following two cases.

(1) $u \notin \mathcal{O}(v)$. In this case, there exists no element $g \in G_n$ such that $g \cdot v = u$. Hence $K_{v,u} = \varnothing$, and therefore

$$\Pr[U = u \mid V = v] = 0. \tag{10}$$

(2) $u \in \mathcal{O}(v)$. Then there exists an element $g_0 \in G_n$ such that $u = g_0 \cdot v$. For any $g \in G_n$, we have

$$\begin{aligned} g \in K_{v,u} &\iff g \cdot v = u \\ &\iff g \cdot v = g_0 \cdot v \\ &\iff (g_0^{-1} g) \cdot v = v \\ &\iff g_0^{-1} g \in \mathrm{Stab}(v). \end{aligned} \tag{11}$$

Therefore,

$$K_{v,u} = g_0 \,\mathrm{Stab}(v), \tag{12}$$

which is a left coset of the stabilizer $\mathrm{Stab}(v)$ in G_n. According to the properties of cosets, we have

$$|K_{v,u}| = |\mathrm{Stab}(v)|. \tag{13}$$

Substituting into Equation (9) gives

$$\Pr[U = u \mid V = v] = \frac{|\mathrm{Stab}(v)|}{|G_n|}, \tag{14}$$

for all $u \in \mathcal{O}(v)$. This completes the proof of blindness for the target quantum computation.□

It is important to clarify the scope of the blindness established here. The above theorem guarantees indistinguishability among target operations within the same Pauli equivalence class: the server observes a uniformly randomized representative of the corresponding Pauli orbit and therefore cannot determine which specific representative is the target operation. However, the present construction does not guarantee indistinguishability between target operations belonging to different Pauli equivalence classes. Thus, the Pauli-orbit membership may remain accessible to the server, while the specific target operation within that orbit is concealed. In this sense, the security property established by the present protocol is blindness within a Pauli equivalence class.

3.2. Statistical analysis of verifiability

We emphasize that the verification mechanism employed in this work probes output quantum states rather than quantum processes. The mapping from a quantum process to its output state is generally non-injective, so that different processes may yield identical output states for a fixed input. Accordingly, a server may evade detection in a single verification round only if it can reproduce the target output state with sufficiently high precision. In the absence of the private information associated with that round, the probability of achieving this is vanishingly small. To quantify the impact of such deviations, we next consider two attack models: a strong attack and a weak attack.

Definition 3 (Strong attack): Let N be the total number of verification rounds and m_{nopass} the number of rounds that do not satisfy Definition 1. If the server applies a cheating operation V_{cheat} in all N rounds, the detected deviation ratio is defined as

$$\gamma = \frac{m_{\text{nopass}}}{N}, \tag{15}$$

with the corresponding acceptance probability

$$P = 1 - \gamma \ll \delta. \tag{16}$$

If the server engages in cheating in all verification rounds, the acceptance probability P is far below the preset threshold δ, and the client rejects the computation result.

To lower the detection probability, the server may perform a weak attack, cheating in only k of the N verification rounds and behaving honestly otherwise.

Definition 4 (weak attack): Let N be the total number of verification rounds and k the number of cheating rounds, with $k\gamma$ rounds detected as cheating according to Definition 3. Accordingly, the detectable deviation probability is

$$P_{\text{cheat}} = \frac{k\gamma}{N} \approx \frac{k}{N}\gamma. \tag{17}$$

In this way, the server can reduce the detectable deviation probability by lowering the attack fraction k/N, thereby making it more likely to pass the statistical test when the condition $P_{\text{cheat}} < 1-\delta$, as defined in Definition 2, is satisfied. In the weak-attack model considered here, the k cheating rounds are randomly selected from the N verification rounds and are not assumed to occur consecutively. The present statistical model does not include temporal correlations between different verification rounds; therefore, the analysis depends on the number of cheating rounds k, rather than on whether the attacked rounds are consecutive.

The relationships among the above statistical parameters can be summarized as follows. Here, N denotes the total number of verification rounds, and δ specifies the minimum acceptance probability. Under the strong-attack scenario, where all N verification rounds are attacked, $\gamma = m_{\text{nopass}}/N$ characterizes the detectable deviation ratio, giving an acceptance probability $P = 1-\gamma$. For a weak attack restricted to k verification rounds, assuming the same detection rate γ for each attacked round, the expected number of detected rounds is $k\gamma$, and the overall detectable deviation probability becomes $P_{\text{cheat}} = k\gamma/N$. Accordingly, the acceptance condition $P = 1 - P_{\text{cheat}} \geq \delta$ requires $P_{\text{cheat}} \leq 1-\delta$, or equivalently, $k \leq \frac{(1-\delta)N}{\gamma}$. Thus, the detectable deviation ratio γ obtained from the strong-attack scenario determines the maximum number of cheating rounds that can remain compatible with the statistical acceptance criterion.

To make the verification procedure explicit, we present the single-qubit CMBQC protocol, whose quantum circuit is shown in Figure 2. The number of verification rounds is set to $N = 15$, with an acceptance threshold $\delta = 0.9$ and a single-round deviation tolerance $\varepsilon = 0.001$. We determine the detectable deviation ratio γ from the strong-attack scenario using numerical simulations. When the server cheats in all $N = 15$ verification rounds, $m_{\text{nopass}} = 12$ rounds fail the statistical test, yielding $\gamma = 12/15 \approx 0.8$ according to Equation (15). According to Definition 2, the acceptance probability is required to satisfy $P \geq \delta$, which implies that the detectable deviation probability must not exceed $1-\delta$. According to Equation (17), the number of cheating rounds in a weak attack is bounded by $k \leq (1-\delta)N/\gamma \approx 1.88$. Thus, in the single-qubit setting, cheating in at most one round does not affect the statistical acceptance outcome. We therefore test the boundary case $\lceil k \rceil = 2$, where acceptance hinges on whether one of the two cheating rounds remains undetected.

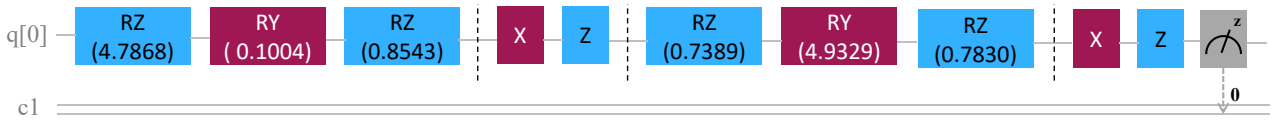

Figure 2. The single-qubit CMBQC protocol implemented on the IBM superconducting platform consists of four parts: state preparation, encryption, target quantum computation, and decryption.

In the single-qubit CMBQC setting, the key space contains only 16 elements, allowing for exhaustive verification over the entire key set, which represents the most stringent verification configuration. By contrast, for multi-qubit systems with $n \geq 2$, the key space grows exponentially as 4^{2n} (e.g., 256 for $n = 2$), rendering exhaustive verification impractical. In this regime, verification is instead carried out by randomly sampling a sufficient number of keys from the full set (see Tables S1 and S2 in Supplemental Material, Section 2).

4. Experiment and results

As a proof-of-principle implementation of the general n-qubit CMBQC framework described above, we experimentally demonstrate its single-qubit instance on IBM's 133-qubit superconducting quantum processor Torino using the parameter settings specified above. The target gates include $U_1^{(1)}$, $U_2^{(1)}$, H, T, and S, and only the results for $U_1^{(1)}$ are presented here, with the remaining data shown in Figure S1 of Section 1 of the Supplemental Material. The additional experiments for $U_2^{(1)}$, H, S, and T exhibit the same qualitative verification behavior as the representative $U_1^{(1)}$ example: honest verification rounds remain consistent with the theoretical predictions within experimental uncertainty, while sufficiently large deviations introduced by cheating operations are successfully identified. Depending on the specific cheating operations, the detected deviations lead to the corresponding statistical acceptance or rejection outcomes, as detailed in Supplemental Material, Section 1.

As shown in Figure 3, the $U_1^{(1)}$-gate BQC experiment comprises 16 interaction rounds, with the 5th round performing the target computation and the remaining 15 rounds serving as verification. Since the target computation round is omitted from the verification plot, the displayed round numbers retain the original interaction-round indices and therefore skip round 5. In this example, rounds 1 and 8 are randomly selected as deviation rounds. For each verification round, the Pauli observable is independently and uniformly sampled from $\{X, Y, Z\}$, without applying any optimization criterion. The realized sequence used in the experiment is explicitly reported below for reproducibility. The Pauli observables used in the 15 verification rounds are, in order, $Y, Y, Z, Z, X, Z, Z, X, X, Z, Y, X, X, X, Y$. As shown in Figure 3, the corresponding theoretical expectation values are -0.6142, -0.7556, -0.2198, -0.2154, 0.7117, 0.2198, 0.2154, 0.6646, -0.7592, 0.3148, -0.6921, 0.7117, -0.6646, -0.7117, 0.7556. The experimentally measured expectation values are -0.5565 ± 0.0469, -0.7206 ± 0.0382, -0.1906 ± 0.0410, -0.1614 ± 0.0245, 0.7292 ± 0.0295, 0.2459 ± 0.0231, 0.4901 ± 0.0219, 0.6677 ± 0.0203, -0.7150 ± 0.0259, 0.3382 ± 0.0298, -0.6912 ± 0.0386, 0.7255 ± 0.0319, -0.6646 ± 0.0388, -0.7155 ± 0.0113, 0.7261 ± 0.0302. For each Pauli-observable measurement, $N_s = 10^4$ shots are used to estimate the corresponding expectation value. Since the measurement outcomes of a Pauli observable are ± 1, the standard error satisfies

$$\sigma_{\langle O\rangle} = \sqrt{\frac{1-\langle O\rangle^2}{N_s}} \leq \frac{1}{\sqrt{N_s}} = 0.01, \tag{18}$$

which is one order of magnitude smaller than the experimental verification tolerance $\varepsilon_{\text{exp}} = 0.1$. Thus, the chosen shot number ensures that finite-sampling fluctuations do not dominate the verification outcome.

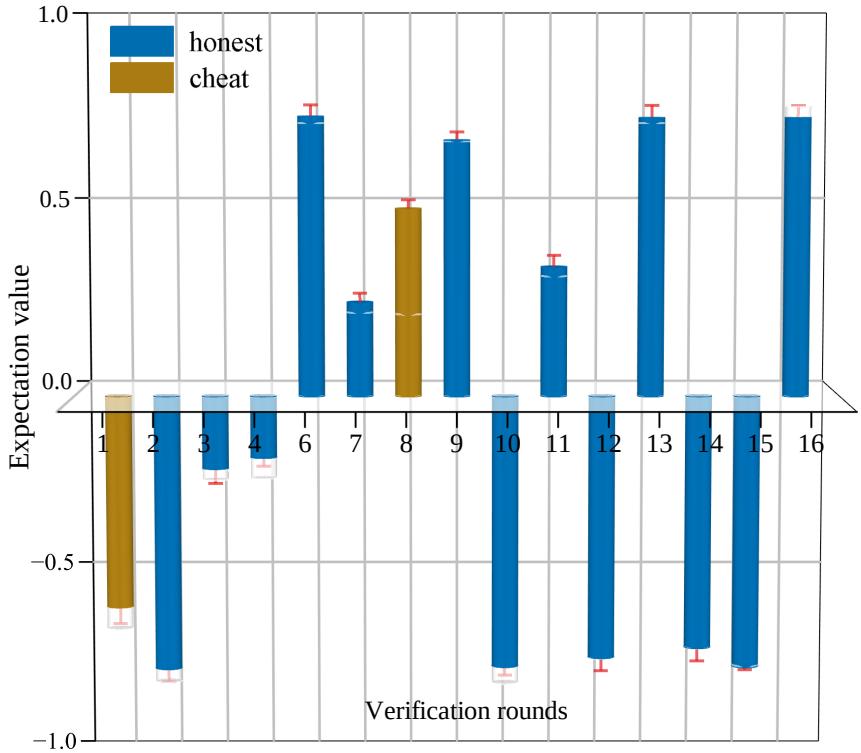

Figure 3. Experimental verification of the CMBQC protocol for a single-qubit gate $U_1^{(1)}$ implemented on the IBM superconducting platform. The protocol consists of 16 interaction rounds, with the 5th round designated as the target computation round and therefore omitted from the verification plot, while the remaining 15 rounds are used for verification. The round labels retain the original interaction-round numbering; hence, the verification-round sequence skips round 5. Blue bars indicate honest execution, while brown bars indicate cheating execution.

Here, $V_{\text{cheat}}^{(r)}$ denotes the cheating operation implemented in the r-th interaction round, whereas V_{cheat} without a superscript denotes the generic cheating operation introduced in the theoretical analysis. It is constructed from the target gate $U_1^{(1)}$ and the corresponding encryption and decryption keys according to

$$D\left(k_{\text{dec}}^{(r)}\right) V_{\text{ideal}}^{(r)} E\left(k_{\text{enc}}^{(r)}\right) = U_1^{(1)}. \tag{19}$$

The superscript r accounts for the round-dependent keys. The cheating operations $V_{\text{cheat}}^{(1)}$ and $V_{\text{cheat}}^{(8)}$ are deliberately chosen to yield output states whose fidelities with the corresponding ideal outputs under $V_{\text{ideal}}^{(1)}$ and $V_{\text{ideal}}^{(8)}$, respectively, exceed 0.9999. Nevertheless, their gate fidelities with respect to the corresponding ideal blinded operations are 0.6570 and 0.5120, respectively.

As discussed above, the theoretical tolerance threshold is set to $\varepsilon = 0.001$. In practice, hardware imperfections and finite sampling lead to deviations in the honest verification rounds of up to approximately 0.05, rendering this theoretical threshold overly stringent for the experimental implementation. We therefore

set $\varepsilon_{\text{exp}} = 0.1$, approximately twice the largest deviation observed in the honest verification data, to provide a conservative margin against false identification of normal experimental fluctuations as cheating. This value is not adopted from a default IBM setting. Moreover, with $N_s = 10^4$ shots, the finite-sampling standard error of a Pauli expectation value is bounded by $1/\sqrt{N_s} = 0.01$, indicating that the experimental tolerance mainly accounts for hardware-related imperfections rather than finite-shot fluctuations. With this threshold, the deviation in round 8 is 0.2747 (theoretical value 0.4498, exceeding the threshold and thus being detected, whereas the deviation in round 1 is 0.0576 (theoretical value 0.0384), which remains below the threshold and is not flagged. Overall, only one detectable deviation is observed among the 15 verification rounds, yielding $P_{\text{cheat}} \approx 0.053$ and hence $1 - P_{\text{cheat}} > 0.9$, so the protocol passes the statistical test.

5. Summary

In summary, we propose and experimentally demonstrate a CMBQC protocol that ensures the blindness of the target quantum algorithm, protecting both the quantum states and the computation process. A key feature of our protocol is the decoupling of encryption and decryption keys, which reduces the client-side quantum-operation burden and relaxes hardware-specific constraints. In conventional circuit-model BQC protocols with gate-by-gate coupling between encryption and decryption, the client must track or update decryption-related information as the delegated circuit proceeds, and the corresponding processing burden therefore increases with the circuit depth. In our CMBQC protocol, by contrast, the client-side quantum operations are confined to a single n-qubit Pauli-encryption layer at the input and a single n-qubit Pauli-decryption layer at the output, with no intermediate client-side quantum operations required during the server-side computation. Although the proof-of-principle experiment presented here is performed on an IBM superconducting quantum processor, the CMBQC protocol is formulated at the circuit level and can in principle be implemented on other circuit-based quantum computing platforms that support the required state preparation, Pauli operations, quantum-circuit execution, and output measurements. We provide proofs of the blindness and verifiability of the proposed protocol, in which verification is achieved with low overhead via statistical sampling of Pauli observables. Experiments on current quantum hardware further confirm the practicality of the approach and its capability to detect intermittent deviations. The experimentally demonstrated CMBQC protocol therefore has potential applications in privacy-preserving delegated computation for quantum cloud and networked quantum environments.

Data availability statement

The experimental data supporting the findings of this study are presented in Figure 3 of the main text and Figure S1 of the Supplementary Information. Figure S1 provides additional experimental results of the CMBQC protocol for different single-qubit target gates.

Declaration of generative AI and AI-assisted technologies

During the preparation of this manuscript, the authors used generative AI tools only to improve language and readability. Specifically, the authors used ChatGPT for language polishing only. The authors take full responsibility for the content of the manuscript.

Acknowledgments

Funding: This work was supported by the Natural Science Foundation of Guangdong Province of China (Grant No. 2023A1515011556), and the Fundamental Research Funds for the Central Universities. Bingwen Feng acknowledges support from the National Natural Science Foundation of China (Grant No. 62472199, U23B2023), Natural Science Foundation of Guangdong Province, China (Grant No. 2025A1515011601), and the Fundamental Research Funds for the Central Universities.

Authors' contribution

Conceptualization, X.Z.; methodology, X.Z.; formal analysis, T.X., B.F. and X.Z.; investigation, T.X., B.F. and X.Z.; validation, T.X., B.F. and X.Z.; writing—original draft preparation, T.X., B.F. and X.Z.; writing—review and editing, T.X., B.F. and X.Z.; supervision, X.Z.; project administration, X.Z.; funding acquisition, X.Z. All authors have read and agreed to the published version of the manuscript.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] Broadbent A, Fitzsimons J, Kashefi E. Universal blind quantum computation. In *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science*, Atlanta, USA, October 25–27, 2009, pp. 517–526.

[2] Morimae T, Fujii K. Secure entanglement distillation for double-server blind quantum computation. *Phys. Rev. Lett.* 2013, 111(2):020502.

[3] Barz S, Kashefi E, Broadbent A, Fitzsimons J, Zeilinger A, *et al.* Demonstration of blind quantum computin. *Science* 2012, 335(6066):303–308.

[4] Morimae T, Fujii K. Blind topological measurement-based quantum computation. *Nat. Commun.* 2012, 3(1):1036.

[5] Li Q, Chan W, Wu C, Wen Z. Triple-server blind quantum computation using entanglement swapping. *Phys. Rev. A* 2014, 89(4):040302.

[6] Childs AM. Secure assisted quantum computation. *Quantum Inf. Comput.* 2005, 5(6):456–466.

[7] Broadbent A. Delegating private quantum computation. *Can. J. Phys.* 2015, 93(9):941–946.

[8] Fisher KAG, Broadbent A, Shalm LK, Yan Z, Lavoie J, *et al.* Quantum computing on encrypted data. *Nat. Commun.* 2014, 5(1):3074.

[9] Rotman JJ. *Advanced Modern Algebra, Part 1*, 3rd ed. Providence: American Mathematical Society, 2015.

[10] Rotman JJ. *Advanced Modern Algebra, Part 2*, 3rd ed. Providence: American Mathematical Society, 2017.