Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Verifiable blind observable estimation

Bo Yang

Bo.Yang@lip6.fr

Sorbonne University

Elham Kashefi

Sorbonne University

Harold Ollivier

École Normale Supérieure - PSL

Article

Keywords:

Posted Date: March 5th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-8691968/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Verifiable blind observable estimation

Bo Yang,[1,*] Elham Kashefi,[1,2,†] and Harold Ollivier[3,‡]

[1] *LIP6, Sorbonne Université, CNRS, 4 place Jussieu, 75005 Paris, France*
[2] *School of Informatics, University of Edinburgh, 10 Crichton Street, EH8 9AB Edinburgh, United Kingdom*
[3] *QAT, DIENS, École Normale Supérieure, PSL University, CNRS, INRIA, 45 rue d'Ulm, Paris 75005, France*

Cryptographic verification is essential for establishing trust in quantum-computing-as-a-service. However, a fundamental gap exists in the current verification landscape: existing efficient protocols are largely restricted to decision problems where correctness is boosted by classical majority voting. This excludes observable estimation, the statistical task underpinning nearly all near-term quantum advantage applications. For such tasks, current verification techniques face a prohibitive trade-off: either weak security guarantees or massive space overhead that exceeds the capacity of near-term hardware. To resolve this, we introduce the Secure Delegated Observable Estimation (SDOE) ideal resource, the first formal cryptographic framework for trustworthy expectation-value estimation within Abstract Cryptography. We then present the Verifiable Blind Observable Estimation (VBOE) protocol, which efficiently constructs this resource. VBOE circumvents the limitations inherent in prior methodologies by enabling the sequential collection of samples with negligible security error, requiring zero extra qubit overhead. By directly averaging computation rounds in classical post-processing, our protocol provides the only known path to rigorous, composable verification for the most common class of near-term quantum-classical hybrid algorithms. This work bridges foundational cryptographic theory with practical quantum tasks, enabling the certification of quantum utility on current and near-future devices.

I. INTRODUCTION

Quantum computing promises to solve problems intractable for classical machines, a capability known as quantum advantage [1–6]. Recent advances in quantum hardware [7–10] have brought this promise closer to reality, with near-term devices demonstrating potential for advantage in tasks that do not require full fault tolerance [11–14]. However, because these devices are error-prone and often remotely accessed, they raise critical questions: How can a client ensure the privacy of a delegated quantum computation while also verifying its result, especially when the output is classically intractable to compute?

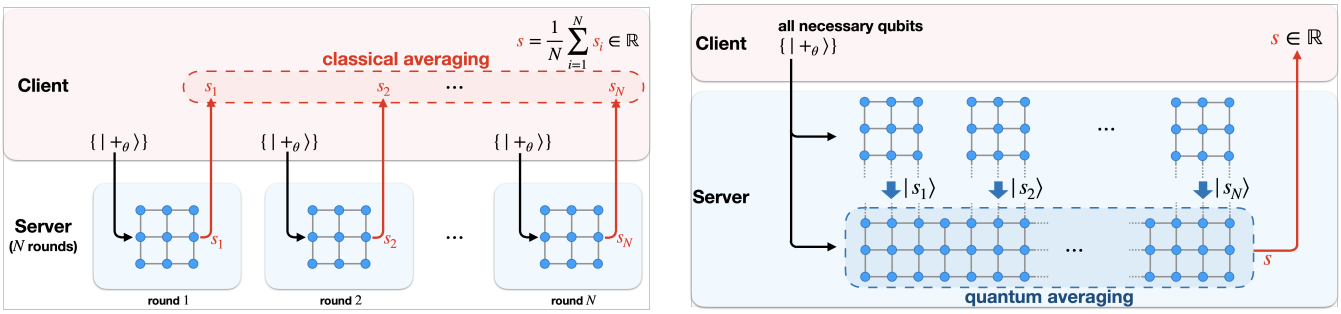
This challenge is particularly acute for algorithms based on expectation value estimation, which underpin many near-term quantum applications such as quantum simulation [15, 16] and machine learning [16–19]. For these applications, establishing a rigorous method to verify the returned estimates is a prerequisite for achieving reliable quantum utility. Yet, even in the case of honest but noisy devices, a fundamental challenge remains: is it possible for a purely classical client to efficiently and robustly verify the outcome of a task delegated to a remote server? While particular schemes have been developed for specific problems [20], no general practical solution currently exists for the broad class of near-term algorithms mentioned above.

Recently, a landmark experiment [21] proposed a heuristic method where one quantum computer simulates the results of another. While this represents a major step forward, the approach is not mathematically rigorous. If two devices happen to share the same noise profile, they could produce identical results in a regime beyond classical simulability while being equally incorrect. Conversely, if the devices provide different results due to differing noise effects, there is no rigorous way to determine which result is correct.

To resolve this, researchers over the last two decades have established rigorous, scalable verification techniques based on cryptographic obfuscation. These protocols, generally referred to as Verifiable Blind Quantum Computing (VBQC) [22] are formulated as an interactive proof system. A verifier with minimal quantum capabilities, specifically single-qubit preparation, can rigorously prove the correctness of a computation performed by a potentially untrusted and noisy server. By randomly inserting traps while keeping the entire computation process obfuscated, the absence of deviations in the trap computations guarantees the correctness of the computation [23]. Such techniques require the verifier to be quantumly linked to the server, a capability demonstrated in both off-chip and on-chip settings [24–26]. This confirms the practicality of these schemes and their alignment with the modular scalable roadmaps of the major hardware platforms.

Nevertheless, a critical subtlety remains. To obtain exponential confidence in the correctness of the result, one must amplify the security bound. If the goal is the correctness of the entire quantum output, the only known solution is full fault-tolerant computing [23]. Fortunately for decision problems (BQP), that is the most common usage of quantum computers, one can instead blindly in-

* Bo.Yang@lip6.fr
† Elham.Kashefi@lip6.fr
‡ harold.ollivier@ens.fr



(a) Round-wise verification

(b) End-to-end verification

FIG. 1. (a) Schematic illustration of round-wise verification, in which the Server sends all samples directly to the Client, who then computes the empirical mean classically. In this setting, the security of each round is guaranteed to be inverse-polynomial in the number of rounds under conventional security analysis, implying that the entire protocol has at best inverse-polynomial security. (b) Schematic illustration of end-to-end verification, in which the Server returns a final empirical mean obtained via quantum averaging. In this case, exponential security can be achieved within the conventional security analysis; however, quantum averaging incurs substantial space overhead, which would be prohibitive for near-term quantum hardware. Note that in both figures, the black arrows represent quantum communication to send single qubits, and the red arrows represent the classical communication to return the final outcome from the server.

terleave computation rounds and test rounds, containing only traps, and apply a simple post-processing, majority voting, to achieve exponential security [27] (see Protocol 3 in Appendix B). Such a solution is then efficient as the security error is exponentially suppressed as the number of computation rounds and test rounds grows, and is unaffected by the size of the computation.

In the case of the observable estimation algorithms, the desired post-processing is instead the classical computation of the empirical average. However, this simple change breaks the existing verification framework. If the server sends all samples directly to the verifier to classically compute the average as depicted in Fig. 1(a), there is no guarantee that individual samples were correctly computed; unlike decision problems, sampling problems lack simple classical error-detection schemes. This results in a security guarantee that is at best inverse polynomial in the number of rounds. To recover exponential security guarantees, the server should perform the empirical mean quantumly by processing all samples in parallel as depicted in Fig. 1(b). Then, the space complexity of the protocol increases significantly, making the approach impractical for current hardware. This represents a fundamental gap that prevents existing verification schemes from being applicable directly to the most important class of problems targeting near-term quantum utility.

To address these limitations, we first introduce the Secure Delegated Observable Estimation (SDOE) resource. This is a new ideal resource within the abstract cryptography (AC) framework [28] that formalises a trustworthy estimation process. SDOE is specifically designed to handle the statistical nature of near-term tasks, ensuring the client obtains an estimate of an observable's expectation value within a specified bias, ϵ, or detects a deviation and aborts, all while preserving the confidentiality of the observable and the result.

We then present the Verifiable Blind Observable Estimation (VBOE) protocol, which efficiently constructs the SDOE resource. VBOE resolves the aforementioned gap by enabling a security error that is negligible in the number of rounds without requiring any extra qubit overhead. Unlike the problematic trade-offs of previous approaches, VBOE allows for the sequential collection of samples, avoiding the space blowup of parallel computation, while using a novel verification analysis to ensure the integrity of the empirical average. Our main result is a rigorous proof that VBOE constructs SDOE with composable security, providing the first efficient framework for verifying observable estimation tasks on untrusted devices.

As a consequence, our protocol directly addresses the growing demand for verifiable quantum advantage in the NISQ era. Its absence of overhead provides a path to polynomial-time verification with exponentially small soundness error for tasks performed on current state-of-the-art quantum hardware [29]. This bridges the gap between foundational cryptographic theory and the heuristic methods recently used in milestone experimental demonstrations of quantum advantage [21, 30].

II. VERIFYING OBSERVABLE ESTIMATION

We define the Secure Delegated Observable Estimation (SDOE) Resource 1 and show that it can be constructed by the proposed Verifiable Blind Observable Estimation (VBOE) Protocol 1 to negligible error within the AC framework.

A. Observable estimation problems

A generic observable estimation problem consists of computing $\mathrm{Tr}[\rho O]$ for some state ρ and observable O up to an additive error $\epsilon > 0$. Without loss of generality, we can always assume that O is a coarse-grained measurement of n-qubits in the $|\pm\rangle$ basis for n sufficiently large. This is because it suffices to absorb the basis change for the eigenspace of O into ρ. As a result, an observable estimation problem is completely specified by $\mathsf{C} \in \mathfrak{C}$, a computation that produces ρ from some fiducial state, say $|+\rangle^{\otimes m}$ for some $m \geq n$. A further simplification that we will adopt in the remainder of this paper is to assume that O is indeed a binary observable, e.g. $X = |+\rangle\langle+| - |-\rangle\langle-|$ with eigenvalues -1 and 1, so that the observable estimation consists of producing a single qubit state ρ and measuring it in the $|\pm\rangle$ basis. We will argue in Section III that our protocol and result can be straightforwardly extended to bounded non-binary observables.

A naive estimation procedure works by sampling outcomes $y_i = 2Y_i - 1 \in \{-1, 1\}$ where $Y_i \leftarrow\!\$\ \mathcal{B}(p)$ and $p = \mathrm{Tr}[\rho|+\rangle\langle+|] = \dfrac{1 + \mathrm{Tr}[\rho O]}{2}$ is the probability of obtaining 1 in the measurement of ρ according to observable X, while $\mathcal{B}(p)$ is the Bernoulli distribution; and then by computing the empirical average of N_c samples.

$$\mu = \frac{1}{N_c} \sum_{i=1}^{N_c} y_i. \tag{1}$$

Using Hoeffding's bound (Lemma 1 in Appendix A), one can assess the performance of such a procedure:

$$\Pr[|\mu - \mathrm{Tr}[\rho O]| \geq \epsilon] \leq 2 \exp\left(-\frac{\epsilon^2}{2} N_c\right). \tag{2}$$

It states that for fixed ϵ, the probability of the estimator being further away than ϵ from the true value $\mathrm{Tr}[\rho O]$ is negligible in N_c, the number of collected samples. This motivates the following definition:

Definition 1 ((ϵ, δ)-Observable Estimation)**.** Given an observable O, a reference state ρ, a protocol (ϵ, δ)-estimates $\mathrm{Tr}[\rho O]$ if the protocol outputs an estimate o that satisfies

$$\Pr[|o - \mathrm{Tr}[\rho O]| \geq \epsilon] \leq \delta. \tag{3}$$

Above, $\epsilon > 0$ is the allowed bias and $\delta > 0$ is an upper bound on the failure probability for obtaining an estimate within the allowed bias.

B. Secure delegated observable estimation (SDOE)

To formalise security for observable estimation problems, we define an ideal resource that has perfect blindness and always returns an estimate of $\mathrm{Tr}[\rho O]$ within bias ϵ in the form of Resource 1.

Resource 1 Secure Delegated Observable Estimation (SDOE)

Public information: $\mathfrak{C}$ a computation class; $N_c, N_t \in \mathbb{N}$; a security parameter $\omega > 0$; and the allowed bias $\epsilon > 0$.
Client's interface: The target computation $\mathsf{C} \in \mathfrak{C}$ to produce the single qubit state ρ to be measured by $O = X$.
Server's interface:

1. The interface is filtered so that when $e = 0$, the interface does not send any information nor take inputs.
2. For $e = 1$, the Resource receives a quantum state σ and F, a list of instructions so that the resource produces $s \in \mathbb{R} \cup \{\mathsf{Abort}\}$.

Processing by the Resource:

1. If $e = 0$, it sets $o = \dfrac{1}{N_c} \sum_{i=1}^{N_c} y_i$ with $y_i = 2Y_i - 1$ where $Y_i \leftarrow\!\$\ \mathcal{B}(p)$, sampled from a Bernoulli distribution $\mathcal{B}(p)$ with $p = \mathrm{Tr}[\rho|+\rangle\langle+|]$.
2. If $e = 1$, it computes s using the transmitted state σ and F.
3. If $s = \mathsf{Abort}$ it forwards $o = \mathsf{Abort}$ to the Client.
4. If $|s - \mathrm{Tr}[\rho O]| \geq \epsilon$ it sets $o = \mathsf{Abort}$ and forwards it to the Client.
5. Otherwise it directly forwards s to the Client.

Here, $e \in \{0, 1\}$ is a flag controlling whether the Server's interface filter is activated or not. Whenever $e = 0$, the ideal resource samples the estimator of $\mathrm{Tr}[\rho O]$ and returns its value if it is within ϵ of the true expectation value. When the Server asks for full access ($e = 1$), the Server receives at most the permitted leakage, i.e. essentially $\mathfrak{C}$ corresponding to all the observable estimation problems that the resource can handle. It also recieves the parameters N_c, N_t, w and ϵ. The Server is then allowed to send a deviation to be applied by the Resource. It takes the form of a quantum state σ and a classical list of quantum and classical instructions that produce either a real number or Abort. If the produced scalar is within ϵ of $\mathrm{Tr}[\rho O]$, then the resource sends the scalar to the Client. Otherwise, it sends Abort.

This definition corresponds to the intuitive notion of secure delegated observable estimation: a malicious Server can only learn the class of observable estimation problems that the resource can tackle, and is only able to influence the result so long as it remains within ϵ of the true expectation value. It also inevitably has the ability to force the protocol to Abort. The pictorial representation of the SDOE resource is provided in Fig. 2.

C. Verifiable blind observable estimation (VBOE)

Considering the SDOE resource defined above, we propose the VBOE protocol to concretely implement this functionality. It is based on the same sequential execution of test and computation rounds as in the RVBQC

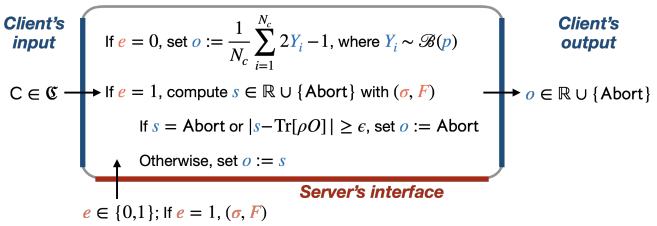


FIG. 2. Schematic illustration of the SDOE resource (Resource 1). The bottom edge of the outer rectangle serves as an interface to the Server. The left edge of the outer rectangle takes inputs from the Client, and the right edge returns outputs to the Client. The variables and equations coloured in blue represent the values generated in the SDOE resource, while those coloured in light red represent the values received at the Server's interface.

(See appendix, Protocol 3) [27], with post-processing and acceptance criteria tailored to observable estimation problems. Here we assume that $\mathfrak{C}$ is a computation class that corresponds to all measurement patterns that can be executed on a graph $G = (V, E)$ with a given flow f.

Protocol 1 Verifiable Blind Observable Estimation (VBOE)

Inputs from Client: The target computation $\mathsf{C} \in \mathfrak{C}$ that produces ρ and allows to measure $\text{Tr}[\rho O]$ with $O = X$.

Protocol:

1. The Client randomly samples indices in $[N]$ for $N = N_c + N_t$ to indicate the locations of test and computation rounds. Let $\mathsf{S_T}$ (resp. $\mathsf{S_C}$) be the index set of test (resp. computation) rounds.
2. For $i \in \mathsf{S_T}$, the Client constructs a test round following the same procedure as for the test rounds of the RVBQC (Step 2 of Protocol 3).
3. Each round, computation or test, is then delegated to the Server using UBQC (Protocol 2).
4. Upon receiving and decoding the result of the computation round $i \in \mathsf{S_C}$, the Client assigns the result to $\tilde{y}_i \in \{-1, 1\}$.
5. Upon receiving the measurement results of test rounds, the Client checks that all the traps have output the expected outcome. If it is the case, the test passed. Otherwise, it failed.
6. If less than ωN_t test rounds failed, it sets $\tilde{o} = \frac{1}{N_c}\sum_{i \in \mathsf{S_C}} \tilde{y}_i$ as the result, otherwise it sets it to $\tilde{o} =$ Abort.

The main modification in VBOE from RVBQC is the post-processing of computation rounds. RVBQC applies classical majority vote over multiple repeated runs to amplify the probability of choosing a correct outcome, while VBOE computes the empirical average over N_c outcomes $\{\tilde{y}_i\}_{i \in \mathsf{S_C}}$. To ensure that the returned value stays within the allowed bound ϵ, the threshold ω needs to be set at the appropriate value. More precisely, we will see in the security proof that it is set in a way that the concentration of probabilities ensures honest executions are always accepted, while deviations that could generate a result too far away from the true value are rejected. The schematic illustration of the VBOE protocol is depicted in Fig. 3.

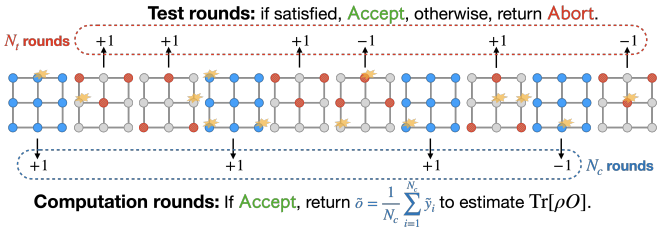


FIG. 3. The schematic illustration of the VBOE protocol. The test rounds have only trap qubits and dummy qubits, shown in red and grey circles, respectively. The computation rounds have only the computation qubits shown in blue circles.

D. Concrete construction of the SDOE resource

The VBOE protocol combines blindness and verifiability and constructs the SDOE resource within negligible error in the AC framework:

Theorem 1 (Composable Security of VBOE). *Let $\mathfrak{C}$ be a class of observable estimation problems that can be estimated using an MBQC pattern on a fixed graph G with a given flow f and chromatic number K. Let $N_c, N_t, \in \mathbb{N}$, let ϵ, ω be constants such that $0 \leq K\omega < \epsilon$. Then, the VBOE protocol (Protocol 1), with N_c computation rounds and N_t test rounds, δ-constructs the SDOE resource. For a constant ratio N_c/N_t, δ is negligible in N_c.*

Following abstract cryptography, to prove this theorem, we need to upper bound the distinguishing advantage between the VBOE protocol and the SDOE resource in the honest (Correctness proof) and malicious (Security proof) settings (see Definition 4 in Appendix B).

Correctness. The proof of correctness relies on the composability of the UBQC protocol. As apparent in Protocol 1, each round is delegated to the server using UBQC (Protocol 2 in Appendix B). Because UBQC perfectly constructs the Blind Delegated Quantum Computation (BDQC) resource (Resource 2 in Appendix B), we can instead perform the proof of correctness using a hybrid protocol where each instantiation of UBQC is replaced by a call to BDQC.

As a result, each outcome y_i for the computation rounds $i \in \mathsf{S_C}$ is processed through $y_i = 2Y_i - 1$, where Y_i is sampled from the Bernoulli distribution $\mathcal{B}(p)$ with probability $p = \frac{1 + \text{Tr}[\rho O]}{2}$. This ensures that the produced empirical average $\tilde{\mu} = \frac{1}{N_c}\sum_{i \in \mathsf{S_C}} \tilde{y}_i$ is obtained from

the same probability distribution as the one used to define the ideal resource. Hence, whenever the ideal resource and the protocol both output the estimated value or both output Abort, their outputs coincide. Consequently, the only distinguishing advantage stems from the two setups having different Abort probabilities.

Indeed, because the test rounds in the protocol are executed perfectly, they consistently give the correct outcome, and the protocol never aborts. For the ideal resource, this is not the case. Whenever the estimator is further away than ϵ from $\mathrm{Tr}[\rho O]$, the ideal resource returns Abort. The probability of such an event happening is upper bounded, using Hoeffding's bound, by $2\exp\left(-\frac{\epsilon^2}{2}N_c\right)$. We can thus conclude that, for ϵ fixed, the distinguishing advantage in the honest case is a negligible function of N_c. □

Security. The security relies heavily on the composability of UBQC. We start by constructing a simulator that we attach to the Server's interface of the ideal resource. Its purpose is to generate plausible transcripts and help the ideal resource in returning $o \in \mathbb{R} \cup \{\mathsf{Abort}\}$ to the Client's interface so that any distinguisher will be unable to tell apart this situation from running the Client's part of the concrete Protocol 1. This simulator is easily constructed from the simulator designed to prove the security of UBQC.

First, it sets $e = 1$. It then prepares EPR pairs for each qubit that the Server is supposed to receive in Protocol 1. It sends all half EPR pairs to the Server, instructs random measurement angles, and retrieves alleged measurement outcomes. It then forwards the second half of the EPR pairs, the chosen angles and received bits to the ideal resource. It also samples at random indices within $[N_c + N_t]$ to define the sets S_C and S_T. It decides which type of test round is associated with each $i \in \mathrm{S}_\mathsf{T}$ and passes this to the ideal resource.

Following the security proof of UBQC, the information passed per round, together with the nature of the round, either computation or test, is sufficient for the ideal resource to generate per-round measurement results following the same, possibly deviated, probability distributions as the ones obtained when running Protocol 1. From the outcomes of computation rounds, the ideal resource is then instructed to compute the empirical average o and from the test rounds to check that no more than ωN_t failed, in which case it sets $o = \mathsf{Abort}$. Using the computed value o, the ideal resource then performs the steps 3, 4, and 5 mentioned in its definition (Resource 1) and aimed at ensuring that the returned o is within ϵ of $\mathrm{Tr}[\rho O]$.

To proceed further with the proof, we note that the perfect blindness of UBQC ensures that the value of s computed by using the quantum state σ provided by the simulator, i.e. the half EPR pairs, and the classical instructions follows the same distribution as $\tilde{o}$ in Protocol 1. The only difference arrives later when the additional check $|s - \mathrm{Tr}[\rho O]| \geq \epsilon$ is performed by the resource and possibly rejects when the protocol would have accepted.

As a result, the distinguishing advantage for telling apart the concrete protocol from the ideal resource is bounded by the total variation distance between the probability distributions of s and o, or equivalently between o and $\tilde{o}$. Because, conditioned on $o, \tilde{o} \in \mathbb{R}$ or $o, \tilde{o} = \mathsf{Abort}$, the distributions of o and $\tilde{o}$ are the same, the total variation distance reduces to:

$$\begin{aligned} &|\Pr[o = \mathsf{Abort}] - \Pr[\tilde{o} = \mathsf{Abort}]| \\ &\quad = \Pr[\tilde{o} \neq \mathsf{Abort} \wedge |\tilde{o} - \mathrm{Tr}[\rho O]| \geq \epsilon]. \end{aligned} \tag{4}$$

This probability can be upper-bounded in 4 steps:

1. we upper bound the probability that the computation rounds provide an empirical average that is more than γ_1 away from $\mathrm{Tr}[\rho O]$, with $\gamma_1 > 0$;

2. given that $\tilde{o} \neq \mathsf{Abort}$, we then upper-bound the probability that a large number of computation rounds, say $(K\omega + \gamma_2)N_c$ for $\gamma_2 > 0$, have been attacked by the server;

3. we recognise that attacking a fraction ϕ of computation rounds yields a deviated empirical average that is not away by more than 2ϕ from the non-deviated one, due to the binary nature of the observable O, taking either -1 or 1 for each round;

4. we then notice that the probabilities in steps 1 and 2 above are negligible functions of N_c and N_t. Thus, if we set $\gamma_1 + 2(K\omega + \gamma_2) < \epsilon$, we have $\Pr[\tilde{o} \neq \mathsf{Abort} \wedge |\tilde{o} - \mathrm{Tr}[\rho O]| \geq \epsilon]$ negligible in N_c and N_t, as it is upper bounded by the sum of the probabilities of step 1 and 2 as a result of the union bound.

This allows us to conclude that the distinguishing advantage for such a set of parameters is negligible in N_c and N_t, thereby proving the security of the VBOE protocol, and, combined with its correctness, proving Theorem 1.

First, it is straightforward to see that the probability at step 1 is upper-bounded by $2\exp\left(-\frac{\gamma_1^2}{2}N_c\right)$ using Hoeffding's inequality.

The probability of step 2 is upper-bounded in the following way. First, because UBQC reduces the attack by the Server to a convex combination of Pauli deviations before the measurements, we can group the deviation strategies by m, the number of attacked rounds. Now, because the location of test rounds and computation rounds are random, the number of affected computation rounds Z follows a hypergeometric distribution with parameters $N_c + N_t$ for the total number of items, m for the number of marked items and N_c for the number of samples. Similarly, the number of affected test rounds, denoted by X, follows a hypergeometric distribution with the roles N_c and N_t swapped.

Let Y be the number of failed test rounds. The idea is now to upper-bound the probability $\Pr[Z \geq (K\omega + \gamma_2)N_c, Y \leq \omega N_t]$ using the following fact that

$$\begin{aligned}
&\max_m \Pr\left[Z \geq (K\omega+\gamma_2)N_c, Y \leq \omega N_t\right] \\
&= \max\left\{ \max_{m\leq m_0} \Pr\left[Z \geq (K\omega+\gamma_2)N_c, Y \leq \omega N_t\right], \right. \\
&\qquad \left. \max_{m> m_0} \Pr\left[Z \geq (K\omega+\gamma_2)N_c, Y \leq \omega N_t\right] \right\} \\
&< \max_{m\leq m_0} \Pr\left[Z \geq (K\omega+\gamma_2)N_c\right] + \max_{m> m_0} \Pr\left[Y \leq \omega N_t\right],
\end{aligned} \tag{5}$$

where $m_0 = \left(K\omega + \frac{\gamma_2}{2}\right)(N_c + N_t)$.

Within this setting, for a given value of $m \leq m_0$, the tail bound for the hypergeometric distribution (Lemma 2 in Appendix A) gives

$$\begin{aligned}
&\Pr[Z \geq (K\omega+\gamma_2)N_c] \\
&\leq \exp\left(-2\left((K\omega+\gamma_2) - \frac{m}{N_c+N_t}\right)^2 N_c\right) \\
&\leq \exp\left(-\frac{\gamma_2^2}{2}N_c\right).
\end{aligned} \tag{6}$$

To bound $\Pr\left[Y \leq \omega N_t\right]$ with $m > m_0$, recall that test rounds are defined as Protocol 3 in RVBQC and that a given deviation on a test round is detected with probability at least $1/K$, where K is the chromatic number of the underlying graph G. This means that conditioned on X, the number of failed test rounds Y is lower bounded in the usual stochastic order by a binomial distribution with X samples and average $1/K$. This implies that, with $m > m_0$,

$$\begin{aligned}
&\Pr\left[Y \leq \omega N_t\right] \\
&= \Pr\left[Y \leq \omega N_t, X \leq N_t\left(\frac{m_0}{N_c+N_t} - \frac{\gamma_2}{4}\right)\right] \\
&\quad + \Pr\left[Y \leq \omega N_t, X > N_t\left(\frac{m_0}{N_c+N_t} - \frac{\gamma_2}{4}\right)\right] \\
&= \Pr\left[Y \leq \omega N_t, X \leq N_t\left(K\omega + \frac{\gamma_2}{4}\right)\right] \\
&\quad + \Pr\left[Y \leq \omega N_t, X > N_t\left(K\omega + \frac{\gamma_2}{4}\right)\right] \\
&< \Pr\left[X \leq N_t\left(K\omega + \frac{\gamma_2}{4}\right)\right] \\
&\quad + \Pr\left[Y \leq \omega N_t \;\middle|\; X = N_t\left(K\omega + \frac{\gamma_2}{4}\right)\right] \\
&< \exp\left(-\frac{\gamma_2^2}{8}N_t\right) + \exp\left(-\frac{\gamma_2^2}{32K^2}N_t\right),
\end{aligned} \tag{7}$$

where we upper-bound respectively the first term and the second term using the concentration inequalities in Lemma 2 and Lemma 1 in Appendix A.

Because both Equations (6) and (7) provide bounds that are negligible in N_c and N_t, this shows that the distinguishing advantage provided by the SDOE resource rejecting more often than the VBOE protocol is indeed negligible in N_c for a fixed ratio N_c/N_t. Hence, we conclude that VBOE is constructing SDOE with negligible error in N_c provided that $K\omega$ is below and bounded away from ϵ by a constant, so the positive constants γ_1 and γ_2 can be set such that $\gamma_1 + 2(K\omega + \gamma_2) < \epsilon$. □

III. DISCUSSION

We have introduced the Verifiable Blind Observable Estimation (VBOE) protocol, which extends the capabilities of verifiable blind delegated quantum computation to the critical task of observable estimation. Our main result, Theorem 1, establishes that VBOE achieves composable security with exponentially negligible error and polynomial-time execution, addressing a longstanding gap in the verification of near-term quantum algorithms.

A. Enabling security in the quantum utility regime

Our primary contribution is the extension of the available toolkit for the verification of quantum computations in the Abstract Cryptography framework. This enables embedding observable estimation in a broader, possibly hybrid, secure computational task while not having to reprove its security in this wider context. This is achieved by introducing a new conceptual tool: an ideal resource (SDOE) that formally incorporates bounded estimation error, reflecting the statistical and possibly noisy nature of near-term algorithms. This extension is not merely a technicality; it provides the only known path to achieving efficient polynomial-time verification with an exponentially small soundness error for this broad and practical class of computations. The proposed VBOE protocol is the first to construct this new resource and to outline a path to realising it in practice.

This unlocks the strongest composable security model for a vast set of near-term applications, whose security analysis was previously out of reach. For instance, in recent works, observable estimation problems, such as out-of-time-order correlations, have been put forward as milestone experiments because their quantum advantage is verifiable using another quantum hardware or natural quantum system, while lying beyond the reach of known classical simulation methods [21, 30]. However, even when verification is attempted using another quantum device, the reliability of the outcome remains conceptually unresolved: without a principled cryptographic framework, the trustworthiness of the devices involved cannot be established a priori. Our framework directly addresses this gap by providing a composable, device-agnostic verification protocol for observable estimation outcomes without requiring trust in any particular quantum prover.

Our framework also provides a formal abstract cryptography security guarantee for recent applications that make use of quantum verification protocols. In particular, Inajetovic et al. [31] introduce a verifiable end-to-end delegated variational quantum algorithm (MB-DVQA) by invoking a variant of RVBQC as a subroutine to ensure the correctness of individual optimisation steps. While their work rigorously establishes the correctness and verifiability of the algorithmic outcome, a composable cryptographic security analysis of the overall construction is left open. Using our result, the MB-DVQA protocol can be analysed at the level of cryptographic resources, relying on the composable security of VBOE for delegated observable estimation tasks that were previously unavailable.

B. Low implementation overhead

A key practical advantage of VBOE is its low implementation overhead, which sets it apart from previous approaches. Conventional methods for verifying observable estimation with exponential security, such as those based on RVBQC, would typically need to verify both the acquisition of measurement outcomes and the empirical average. This would either blow up the space overhead if the outcomes are acquired in parallel, or require long-term memory to store the partial sum of outcomes if the outcomes are acquired sequentially. In both cases, we would need a fault-tolerant quantum computer.

In contrast, the VBOE protocol is designed to preserve the structure of the original target computation generating a single measurement outcome per round. By letting the client average over the outputs of the computation rounds, VBOE suppresses the circuit overhead of previous approaches while maintaining the security guarantees. As a result, VBOE offers a significantly more practical verification strategy for observable estimation tasks, particularly in regimes where quantum resources are limited.

We envisage that applying our method on such platforms would enable end-to-end verification for more meaningful computational tasks, beyond proof-of-principle experiments. Following the first experimental demonstration of UBQC on photonic devices [32], recent work has reported the successful implementation of RVBQC on trapped-ion platforms [26]. With the rapid progress of quantum hardware, a growing number of high-quality quantum devices with remote access are becoming available [9, 10]. Building on these developments, the proposed VBOE protocol can likewise be executed on existing quantum platforms, where more practically relevant applications based on observable estimation are within reach. This represents a significant step toward bridging theoretical protocols and practical, device-level implementations of verifiable delegated quantum computation.

C. Compatibility with quantum error mitigation

By its very nature as a protocol for verifying expectation values, VBOE is naturally compatible with near-term and early fault-tolerant error-suppression techniques. A key open question is how to securely integrate quantum error mitigation (QEM) [33–38], which aims to recover noise-free expectation values through classical post-processing with limited quantum overhead. Integrating QEM with VBOE has the potential to enhance the noise robustness of the protocol by reducing abort probabilities, while simultaneously enabling credible error mitigation with explicit cryptographic guarantees.

Another promising direction lies in combining multiparty variants of VBOE with device-efficient near-term techniques, such as circuit cutting and hybrid tensor-network methods [39–42]. Such a synergy could lead to a secure quantum-classical framework in which large-scale simulation tasks are decomposed into smaller quantum and classical subroutines and distributed across multiple remote parties.

ACKNOWLEDGMENTS

B.Y. acknowledges the insightful and fruitful discussions with Dominik Leichtle and Jinge Bao from the University of Edinburgh, and Eliott Mamon from Sorbonne Université. All authors received funding from the ANR research grants ANR-21-CE47-0014 (SecNISQ), ANR-22-PNCQ-0002 (HQI).

AUTHOR CONTRIBUTIONS

The idea was conceived by B.Y. All authors contributed to the theoretical results. E.K. and H.O. supervised the project.

COMPETING INTERESTS

The authors declare no competing interests.

Appendix A: Concentration inequalities

Lemma 1 (Hoeffding's inequality)**.** *Let $X_1, \ldots, X_n$ be $n \in \mathbb{N}$ independent random variables such that $a_i \leq X_i \leq b_i$ almost surely for each $i \in [n]$. Let $S_n = \sum_{i=1}^{n} X_i$. Then for any $t > 0$,*

$$\begin{aligned} \Pr[S_n - \mathbb{E}[S_n] \leq -t] &\leq \exp\left(-\frac{2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right), \\ \Pr[S_n - \mathbb{E}[S_n] \geq t] &\leq \exp\left(-\frac{2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right). \end{aligned} \tag{A1}$$

Definition 2 (Hypergeometric distribution)**.** Let $N, K, n \in \mathbb{N}$ with $0 \leq n, K \leq N$. A stochastic variable X is said to follow the hypergeometric distribution, denoted as $X \sim \text{Hypergeometric}\,(N, K, n)$, if its probability mass function is described by

$$\text{Hypergeometric}\,(N, K, n)\,(k) = \frac{\binom{K}{k}\binom{N-K}{n-k}}{\binom{N}{n}}. \quad \text{(A2)}$$

One possible interpretation is to see X as the number of marked items when choosing n items from a set of size N containing K marked items, without replacement.

Lemma 2 (Concentration for the hypergeometric distribution)**.** *Let $X \sim \text{Hypergeometric}\,(N, K, n)$ be a random variable and $0 < t < \frac{K}{N}$. It then holds that*

$$\Pr\left[X \leq \left(\frac{K}{N} - t\right) n\right] \leq \exp\left(-2t^2 n\right). \quad \text{(A3)}$$

As a corollary, we obtain the tail inequality

$$\Pr[X \leq \lambda] \leq \exp\left(-2n\left(\frac{K}{N} - \frac{\lambda}{n}\right)^2\right). \quad \text{(A4)}$$

Let also $\lambda > 0$ be a positive value. Using Serfling's bound for the hypergeometric distribution, it holds that

$$\Pr\left[\sqrt{n}\left(\frac{X}{n} - \frac{N}{K}\right) \geq \lambda\right] \leq \exp\left(-\frac{2\lambda^2}{1 - \frac{n-1}{N}}\right). \quad \text{(A5)}$$

As a corollary, we obtain the concentration inequality of hypergeometric distribution symmetric to Eq. (A4)*,*

$$\Pr[X \geq \lambda] \leq \exp\left(-2n\left(\frac{K}{N} - \frac{\lambda}{n}\right)^2\right). \quad \text{(A6)}$$

Appendix B: Composable security of delegated quantum computation protocols

Delegated quantum computation protocols [23, 27, 43] allow clients with limited quantum capabilities, such as single-qubit state preparation and communication, to delegate tasks to a powerful server while retaining security guarantees such as blindness and verifiability. The security of these protocols can be rigorously analysed within the Abstract Cryptography (AC) framework [28], which formalises composable security in a modular way. This modular framework enables composable security guarantees without requiring incremental and exhaustive proofs of the entire protocol whenever individual components are combined.

In what follows, we first introduce the AC framework, then review the universal blind quantum computation (UBQC) protocol [43] based on MBQC, and explain how AC captures its security. Finally, we review the Verifiable Blind Quantum Computation (VBQC) protocol [23] and the Robust VBQC (RVBQC) protocol [27] combined with their security guarantees.

1. Abstract cryptography (AC)

Abstract cryptography is a cryptographic framework designed to be top-down and axiomatic to analyse the security of a protocol in an arbitrarily adversarial environment. In contrast to the conventional game-based security that analyses each specific adversarial scenario, the AC framework provides universal composable security. Composably secure protocols within the AC framework will maintain their security when composed with each other in parallel or in series, ensuring a modular composition of security for the entire combined protocol as well.

The AC framework consists of abstract systems with well-distinguished and labelled interfaces to transmit information to other systems. Systems are classified into resources, converters, filters, and distinguishes. The AC framework aims to construct a new secure resource $\pi\mathcal{R}$ from an available resource $\mathcal{R}$ and a protocol π by showing the security of π. Here, the resource $\mathcal{R}$ is an abstract system with an index set of interfaces $\mathcal{I}$ for mediating transcripts. The protocol $\pi = \{\pi_i\}_{i\in\mathcal{I}}$ is a set of converters π_i indexed by $\mathcal{I}$, where each converter is a two-interface system mediating between the resource and an external party.

A protocol π is proved to be secure by showing the statistical indistinguishability between the constructed resource $\pi\mathcal{R}$ and the ideal resource $\mathcal{S}$, i.e. any distinguisher cannot distinguish with high probability the two resources $\pi\mathcal{R}$ and $\mathcal{S}$. In concrete terms, the distinguisher is an abstract system that interacts with a resource, attempting to decide whether it is connected to a real resource or an ideal one. It may send inputs, receive outputs, and exploit any observable behaviour in order to distinguish the two resources. Ultimately, the distinguisher must output a single bit indicating its guess: for instance, outputting 1 if the distinguisher believes it is interacting with the constructed resource $\pi\mathcal{R}$ and 0 otherwise. The formal definition of statistical indistinguishability between two resources can be stated as follows.

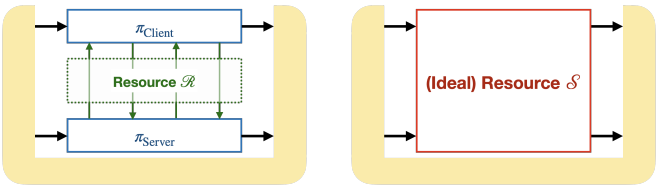
Definition 3 (Statistical Indistinguishability of Resources)**.** Let $\epsilon > 0$, and let $\mathcal{R}_1$ and $\mathcal{R}_2$ be two resources with the same input and output interfaces. The resources are ϵ-statistically-indistinguishable if, for any unbounded distinguisher $\mathcal{D}$, the following holds:

$$\left|\Pr\left[\mathcal{D}\left(\mathcal{R}_1\right) = 1\right] - \Pr\left[\mathcal{D}\left(\mathcal{R}_2\right) = 1\right]\right| \leq \epsilon, \quad \text{(B1)}$$

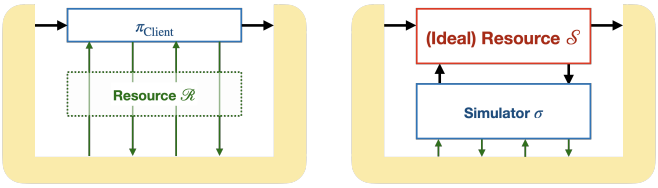
which is denoted by $\mathcal{R}_1 \approx_\epsilon \mathcal{R}_2$, and ϵ is referred to as distinguishing advantage.

Here, the distinguishing advantage ϵ quantifies how much better a distinguisher can perform than random

guessing. If two resources are completely indistinguishable, the success probability is $\frac{1}{2}$ (the same as random guessing), yielding $\epsilon = 0$. Otherwise, the distinguishing advantage is ϵ, the distinguisher can succeed with probability $\frac{1}{2} + \epsilon$.



(a) correctness: $\pi\mathcal{R} \approx_\epsilon \mathcal{S}$



(b) security: $\pi_{\text{Client}}\mathcal{R} \approx_\epsilon \mathcal{S}\sigma$

FIG. 4. The schematic illustrations of correctness and security are depicted in (a) and (b), respectively. On the basis of a secure resource $\mathcal{R}$ as an established channel between the Client and the Server, the protocol $\pi = (\pi_{\text{Client}}, \pi_{\text{Server}})$ constructs a new resource $\pi\mathcal{R} = \pi_{\text{Client}}\mathcal{R}\pi_{\text{Server}}$. The transcripts are written as arrows, and the yellow object is the distinguisher that manages the input and output transcripts between the resource and the protocol of interest.

When constructing a resource $\pi\mathcal{R}$ from a resource $\mathcal{R}$ and a protocol π, the security of π is then characterised by the indistinguishability between $\pi\mathcal{R}$ and the ideal resource $\mathcal{S}$. Here, we restrict to the two-party case with an honest "Client" and a potentially malicious "Server". The following definition defines how well the protocol π constructs $\mathcal{S}$ from $\mathcal{R}$.

Definition 4 (Construction of Resources)**.** Let $\epsilon > 0$. We say that a two-party protocol π, between an honest Client and a potentially malicious Server, ϵ-statistically-constructs resource $\mathcal{S}$ from resource $\mathcal{R}$ if,

- it is correct: $\pi\mathcal{R} \approx_\epsilon \mathcal{S}$, i.e. when the Server is honest, the client-side outputs between $\pi\mathcal{R}$ and $\mathcal{S}$ are ϵ-statistically indistinguishable;
- it is secure against the malicious Server, i.e. there exists a simulator σ such that $\pi_{\text{Client}}\mathcal{R} \approx_\epsilon \mathcal{S}\sigma$, where π_{Client} is π's Client side protocol.

Intuitively, correctness ensures that the protocol behaves as intended when all parties are honest, while security guarantees that malicious behaviour can be emulated in the ideal world by a simulator, thereby preserving composable security. The existence of such a simulator implies that the use of $\pi\mathcal{R}$ with a malicious Server is still well-indistinguishable from using the ideal resource $\mathcal{S}$, which is designed to be secure. The schematic illustrations of correctness and security in Definition 4 are presented in Fig. 4.

Using the definitions above, we can state the following general composition theorem [28] that guarantees the additive accumulation of distinguishing advantage when composing two statistically secure protocols.

Theorem 2 (General Composition of Resources [28])**.** *Let $\mathcal{R}$, $\mathcal{S}$ and $\mathcal{T}$ be resources, α, β and* id *be protocols, where protocol* id *does not modify the resource it is applied to. Let $\circ$ and $\mid$ denote the sequential and parallel composition of protocols and resources, respectively. Then the following implications hold:*

- *Sequential composability: if $\alpha\mathcal{R} \approx_{\epsilon_\alpha} \mathcal{S}$ and $\beta\mathcal{S} \approx_{\epsilon_\beta} \mathcal{T}$, then $(\beta \circ \alpha)\,\mathcal{R} \approx_{\epsilon_\alpha + \epsilon_\beta} \mathcal{T}$.*
- *Context insensitivity: if $\alpha\mathcal{R} \approx_{\epsilon_\alpha} \mathcal{S}$, then $(\alpha \mid \text{id})\,(\mathcal{R} \mid \mathcal{T}) \approx_{\epsilon_\alpha} (\mathcal{S} \mid \mathcal{T})$.*

Combining these two properties yields the composability of protocols.

2. Universal blind quantum computation (UBQC)

Resource 2 Blind Delegated Quantum Computation (BDQC)

Public Information: $(\mathfrak{C}, G, f)$ defined as below.
Inputs at the Client's interface: The target computation $\mathsf{C} \in \mathfrak{C}$, its associated measurement pattern C that contains the graph $G = (V, E)$, the input and output sets $\mathrm{I}, \mathrm{O} \subseteq V$, the measurement angles $\{\phi_v\}_{v \in V}$, and the flow f.
Process at the Server's interface:

1. The Resource receives from the Server $e \in \{0, 1\}$, a flag whether to leak information to Server.
2. If $e = 1$, the Resource sends to the Server the allowed leakage $l_{\mathfrak{C}} = (\mathfrak{C}, G, f)$.
3. The Resource receives at its Server's interface the deviation $(\rho_{\mathrm{R}}, \mathsf{F})$ as a pair of an ancillary state ρ_{R} and a CPTP map F.

Outputs at the Client's interface: The Resource sets $\vec{\mathrm{b}} := \mathrm{Tr}\left[\mathsf{F}\left(|\vec{b}\rangle\langle\vec{b}| \otimes \rho_{\mathrm{R}}\right)\right] \in \{0, 1\}^{|\mathrm{O}|}$, where $\vec{b} \in \{0, 1\}^{|\mathrm{O}|}$ is the correct output following the procedure of measurement pattern C corresponding to the target computation C. The Resource returns $\vec{\mathrm{b}}$ at the Client's interface.

The UBQC protocol achieves perfect blindness for the Client to delegate its quantum computation to the untrusted Server, when the Client can prepare and send a sequence of single qubits through quantum communication. The procedure of UBQC is based on the measurement-based quantum computation (MBQC) model grounded in the principle of gate teleportation [44–48]. In this model, the computation proceeds by first preparing a highly entangled resource state, typically a

graph state, and then performing a sequence of adaptive single-qubit measurements in rotated bases. The measurement outcomes determine subsequent measurement angles, enabling the realisation of arbitrary quantum operations. More formally, the procedure of MBQC is defined as the measurement pattern in Definition 5.

Protocol 2 Universal Blind Quantum Computation (UBQC)

Inputs from Client: The target computation $\mathsf{C} \in \mathfrak{C}$, its associated measurement pattern C that contains the graph $G = (V, E)$, the input and output sets $\mathrm{I}, \mathrm{O} \subseteq V$, the measurement angles $\{\phi_v\}_{v \in V}$, and the flow f.

Protocol:

1. The Client sends the graph's description $(G, \mathrm{I}, \mathrm{O})$ to the Server.
2. The Client generates secret parameters:
   (a) (X randomisation) The Client chooses a random bit $\mathrm{a}_v^{\mathrm{init}} \in \{0,1\}$ for $v \in \mathrm{I}$ and sets $\mathrm{a}_v^{\mathrm{init}} = 0$ for $v \in V \backslash \mathrm{I}$. The Client also computes $\mathrm{a}_v^{\mathrm{prop}} = \bigoplus_{j \in N_G(v)} \mathrm{a}_j^{\mathrm{init}} \in \{0,1\}$ for all $v \in V$.
   (b) (Z randomisation) The Client chooses a random bit $\mathrm{r}_v \in \{0,1\}$ for all $v \in V$.
   (c) (randomisation for blindness) The Client chooses a random $\theta_v \in \Theta$ for all $v \in V$.
3. The Client prepares and sends to the Server all single qubits for $v \in V$. For $v \in \mathrm{I}$, the Client sequentially sends each qubit in $\left(\bigotimes_{v \in \mathrm{I}} \mathsf{Rz}_v(\theta_v)\mathsf{X}_v^{\mathrm{a}_v^{\mathrm{init}}}\right)[\rho_{\mathrm{init}}]$. For $v \in V \setminus \mathrm{I}$, the Client sends $|+_{\theta_v}\rangle$.
4. The Server applies a CZ gate between qubits v_1 and v_2 if $(v_1, v_2) \in E$.
5. For each $v \in V$, the Client and Server interactively perform the MBQC process. Once the Client receives the measurement outcome $\mathrm{b}_j \in \{0,1\}$ for all $j \in S_{X,v} \cup S_{Z,v}$, where $S_{X,v} = f^{-1}(v), S_{Z,v} = \{j \mid v \in N_G(f(j))\}$, the Client computes the adaptive angle update ϕ'_v. The Client then computes the measurement angle δ_v masked with $\mathrm{a}_v^{\mathrm{init}}$, $\mathrm{a}_v^{\mathrm{prop}}$, and r_v for the QOTP randomisation, and θ_v for the blindness:
$$\begin{aligned} s_{X,v} &= \bigoplus_{j \in S_{X,v}} \mathrm{b}_j \oplus \mathrm{r}_j, \quad s_{Z,v} = \bigoplus_{j \in S_{Z,v}} \mathrm{b}_j \oplus \mathrm{r}_j, \\ \phi'_v &= (-1)^{s_{X,v}} \phi_v + s_{Z,v}\pi, \\ \delta_v &= (-1)^{\mathrm{a}_v^{\mathrm{init}}} \phi'_v + \theta_v + (\mathrm{r}_v + \mathrm{a}_v^{\mathrm{prop}})\,\pi. \end{aligned} \tag{B2}$$
   Note that $s_{X,v} = s_{Z,v} = 0$ for $v \in \mathrm{I}$. The Client sends to the Server the angle δ_v and the Server returns to the Client a bit $\mathrm{b}_v \in \{0,1\}$ as a measurement result of qubit v with basis $\{|+_{\delta_v}\rangle, |-_{\delta_v}\rangle\}$.
6. The Client returns $\vec{\mathrm{b}} \oplus \vec{\mathrm{r}} \in \{0,1\}^{|\mathrm{O}|}$ as the final output, where $\vec{\mathrm{b}} \in \{0,1\}^{|\mathrm{O}|}$ (resp. $\vec{\mathrm{r}}$) is a bitstring of the binaries b_v (resp. r_v) for all $v \in \mathrm{O}$.

Definition 5 (Measurement Pattern)**.** A pattern in the MBQC model is given by a graph $G = (V, E)$, input and output vertex sets $\mathrm{I}, \mathrm{O} \subseteq V$, a flow function f which induces a partial order $\preceq_G$ of the qubits V, and a set of measurement angles $\{\phi_v\}_{v \in V}$ in the X-Y plane of the Bloch sphere.

To rigorously define the task of blind delegation of quantum computation, we introduce the Blind Delegated Quantum Computation (BDQC) resource, where, by design, the server never learns the precise computation but instead only the class of computation that the delegated task belongs to $\mathfrak{C}$, i.e. the prepared graph G and its flow f. This resource can then be constructed perfectly using the UBQC protocol [43], described in Protocol 2. Note that Protocol 2 is adapted to classical outputs only.

One can describe the composable security of the UBQC protocol by the words of the AC framework. To state this, one first defines an ideal resource as a hypothetical system that achieves the desired functionality, which is secure by definition. For the case of UBQC, the ideal functionality returns potentially biased output by keeping the blindness of the computation up to the allowed leakage $l_{\mathfrak{C}} = (\mathfrak{C}, G, f)$. This is formally defined as the BDQC resource (Resource 2) that enables the server to influence the outcome by modelling a potential deviation, while leaking no information to the server beyond the prescribed nature of leakage. The security is then analysed by evaluating the indistinguishability between the UBQC protocol and the BDQC resource with respect to the correctness and security against the malicious Server defined in Definition 4.

The UBQC protocol is shown to achieve perfect composable security [49], i.e. the UBQC protocol and the BDQC resource are perfectly indistinguishable. The blindness, which is the only functionality of interest in the BDQC resource, is realised by the Client's use of the quantum one-time pad (QOTP) [50–54] to randomise measurement angles and measurement results. Formally, the security of UBQC is stated as follows.

Theorem 3 (Security of UBQC [49])**.** *The UBQC protocol (Protocol 2) perfectly constructs the BDQC resource (Resource 2) leaking only public information* $(\mathfrak{C}, G, f)$.

3. Robust verifiable blind quantum computation (RVBQC)

While the UBQC protocol ensures blindness, the client may also wish to verify the result of the computation, i.e. ensure that the provided result has not been tempered with. This is expressed by the Secure Delegated Quantum Computation (SDQC) resource in Resource 3.

Clearly, the SDQC resource either provides the expected result or aborts depending on the flag bit d transmitted by the malicious server. The Verifiable Blind Quantum Computation (VBQC) protocol [23] constructs

Resource 3 Secure Delegated Quantum Computation (SDQC)

Public Information: $(\mathfrak{C}, G, f, N)$ defined as below.
Inputs at the Client's interface: The target computation $\mathsf{C} \in \mathfrak{C}$, its associated measurement pattern C that contains the graph $G = (V, E)$, the input and output sets $\mathrm{I}, \mathrm{O} \subseteq V$, the measurement angles $\{\phi_v\}_{v \in V}$, the flow f, and the number of total rounds N.
Process at the Server's interface:

1. Receive from Server $e \in \{0, 1\}$, a flag whether to leak information to Server.
2. If $e = 1$, send to Server the allowed leakage $l_{\mathfrak{C}} = (\mathfrak{C}, G, f, N)$.
3. Receive from Server $d \in \{0, 1\}$, a flag whether to deviate the computation.

Outputs at the Client's interface: Let $\vec{b} \in \{0, 1\}^{|\mathrm{O}|}$ be the correct output following the procedure of measurement pattern C corresponding to C.

1. If $d = 0$, set $\vec{\mathrm{b}} := \vec{b}$ and return $(\mathsf{Acc}, \vec{\mathrm{b}})$ to the Client.
2. If $d = 1$, return $(\mathsf{Rej}, \perp)$.

the SDQC resource with negligible distinguishability by embedding "trap" qubits and "dummy qubits" into the UBQC protocol so that it can both execute the computation while probing the behaviour of the server.

More precisely, the trap qubits are single-qubit deterministic computations whose outcomes are efficiently simulable by the Client while remaining hidden from the Server. Specifically, a trap qubit is initialised into $|+_\theta\rangle$ and measured on the $\{|+_\theta\rangle, |-_\theta\rangle\}$ basis, outputting the eigenvalue 1 under its honest execution. The dummy qubits serve to isolate the trap qubits on the MBQC pattern and to mask the locations of the traps. Note that our framework is not restricted to isolated trap qubits according to the graph colouring, and one can also use further optimised dummyless traps [55] and general traps [56].

With this trappification scheme, the Client can detect deviations if a trap is affected by a harmful deviation that can non-trivially affect the computation. However, the embedded traps and dummies among computation qubits would increase the qubit overhead as they enlarge the MBQC pattern. Besides, VBQC requires fault tolerance for the security amplification to achieve a negligible construction error in the AC framework.

These issues are solved for BQP computations by the Robust VBQC (RVBQC) protocol [27], keeping the verifiability guarantees of VBQC while leveraging a minimal overhead construction and introducing robustness against noise. As described in Protocol 3, the protocols consist of N_c computation rounds and N_t test rounds containing trap and dummy qubits. A threshold $0 < \omega < 1$ is introduced to allow a constant number of test failures without aborting the computation, thus ensuring robustness against noise. For the computation rounds, the protocol repeats the delegated task multiple times and applies a classical majority vote over the outputs, enabling the Client to classically amplify correctness and achieve negligible construction error for the SDQC resource in the AC framework [27, 56].

Protocol 3 Robust VBQC (RVBQC) [27]

Inputs from Client: The target computation $\mathsf{C} \in \mathfrak{C}$, the graph $G = (V, E)$, the flow f, and the K-colouring $\{\mathsf{V}_k\}_{k=1}^K$ of G.
Protocol:

1. The Client randomly samples indices in $[N]$ for $N = N_c + N_t$ to indicate the locations of test and computation rounds. Let S_T (resp. S_C) be the index set of test (resp. computation) rounds.
2. For $i \in \mathrm{S}_\mathsf{T}$, the Client constructs a test round following the same procedure as for the test rounds of the RVBQC (Protocol 3).
   (a) The Client chooses uniformly at random a colour $\mathsf{V}_j \in \{\mathsf{V}_k\}_{k=1}^K$ to specify the trap qubits.
   (b) The Client sends qubits to the Server. If $v \notin \mathsf{V}_j$ (dummy), the Client chooses a bit $\mathrm{d}_v \in \{0, 1\}$ uniformly at random and sends the state $|d_v\rangle$. Otherwise, the Client chooses $\theta_v \in \Theta$ at random and sends the state $|+_{\theta_v}\rangle$.
   (c) The Server performs CZ gates between all its qubits corresponding to an edge in the set E.
   (d) For $v \in V$, the Client sends a measurement angle δ_v, the Server measures the appropriate corresponding qubit in the basis $\{|+_{\delta_v}\rangle, |-_{\delta_v}\rangle\}$, returning outcome b_v to the Client. The angle δ_v is defined as follows:
      - If $v \notin \mathsf{V}_j$ (dummy): the Client chooses δ_v from Θ uniformly at random.
      - If $v \in \mathsf{V}_j$ (trap): the Client chooses $\mathrm{r}_v \in \{0, 1\}$ uniformly at random and sets $\delta_v = \theta_v + \mathrm{r}_v \pi$.
   (e) For all $v \in \mathsf{V}_j$ (traps), the Client computes $\mathrm{d}_v = \bigoplus_{k \in N_G(v)} d_k \in \{0, 1\}$, the sum over the values of neighbouring dummies of the trap qubit v in the ith round. the Client then verifies whether $\mathrm{b}_v = \mathrm{r}_v \oplus \mathrm{d}_v$ holds for all $v \in \mathsf{V}_j$. If this does not hold, the test round is considered failed.
3. For $i \in \mathrm{S}_\mathsf{C}$, the Client delegates to the Server the computation round using UBQC (Protocol 2).
4. If more than ωN_t test rounds failed, the Client returns Abort. Otherwise, the Client performs a majority vote over the outputs of the computation rounds: if there exists an outcome that appears in more than half of the computation rounds, the Client returns that outcome; otherwise, the Client returns Abort.

[1] P. Shor, Algorithms for quantum computation: discrete logarithms and factoring, in *Proceedings 35th Annual Symposium on Foundations of Computer Science* (1994) pp. 124–134.

[2] L. K. Grover, A fast quantum mechanical algorithm for database search, in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96 (Association for Computing Machinery, New York, NY, USA, 1996) p. 212–219.

[3] J. Preskill, Quantum Computing in the NISQ era and beyond, Quantum **2**, 79 (2018).

[4] J. Preskill, Quantum computing and the entanglement frontier (2012).

[5] O. Lanes, M. Beji, A. D. Corcoles, C. Dalyac, J. M. Gambetta, L. Henriet, A. Javadi-Abhari, A. Kandala, A. Mezzacapo, C. Porter, S. Sheldon, J. Watrous, C. Zoufal, A. Dauphin, and B. Peropadre, A framework for quantum advantage (2025).

[6] H.-Y. Huang, S. Choi, J. R. McClean, and J. Preskill, The vast world of quantum advantage (2025).

[7] Z. Ni, S. Li, X. Deng, Y. Cai, L. Zhang, W. Wang, Z.-B. Yang, H. Yu, F. Yan, S. Liu, *et al.*, Beating the break-even point with a discrete-variable-encoded logical qubit, Nature **616**, 56 (2023).

[8] R. S. Gupta, N. Sundaresan, T. Alexander, C. J. Wood, S. T. Merkel, M. B. Healy, M. Hillenbrand, T. Jochym-O'Connor, J. R. Wootton, T. J. Yoder, *et al.*, Encoding a magic state with beyond break-even fidelity, Nature **625**, 259 (2024).

[9] A. C. Hughes, R. Srinivas, C. M. Löschnauer, H. M. Knaack, R. Matt, C. J. Ballance, M. Malinowski, T. P. Harty, and R. T. Sutherland, Trapped-ion two-qubit gates with ¿99.99% fidelity without ground-state cooling (2025).

[10] A. Ransford, M. S. Allman, J. Arkinstall, J. P. Campora, S. F. Cooper, R. D. Delaney, J. M. Dreiling, B. Estey, C. Figgatt, A. Hall, A. A. Husain, A. Isanaka, C. J. Kennedy, N. Kotibhaskar, I. S. Madjarov, K. Mayer, A. R. Milne, A. J. Park, A. P. Reed, R. Ancona, M. P. Andersen, P. Andres-Martinez, W. Angenent, L. Argueta, B. Arkin, L. Ascarrunz, W. Baker, C. Barnes, J. Bartolotta, J. Berg, R. Besand, B. Bjork, M. Blain, P. Blanchard, R. Blume-Kohout, M. Bohn, A. Borgna, D. Y. Botamanenko, R. Boutelle, N. Brown, G. T. Buckingham, N. Q. Burdick, W. C. Burton, V. Carey, C. J. Carron, J. Chambers, J. Children, V. E. Colussi, S. Crepinsek, A. Cureton, J. Davies, D. Davis, M. DeCross, D. Deen, C. Delaney, D. DelVento, B. J. DeSalvo, J. Dominy, R. Duncan, V. Eccles, A. Edgington, N. Erickson, S. Erickson, C. T. Ertsgaard, B. Evans, T. Evans, M. I. Fabrikant, A. Fischer, C. Foltz, M. Foss-Feig, D. Francois, B. Freyberg, C. Gao, R. Garay, J. Garvin, D. M. Gaudiosi, C. N. Gilbreth, J. Giles, E. Glynn, J. Graves, A. Hansen, D. Hayes, L. Heidemann, B. Higashi, T. Hilbun, J. Hines, A. Hlavaty, K. Hoffman, I. M. Hoffman, C. Holliman, I. Hooper, B. Horning, J. Hostetter, D. Hothem, J. Houlton, J. Hout, R. Hutson, R. T. Jacobs, T. Jacobs, M. Johannsen, J. Johansen, L. Jones, S. Julian, R. Jung, A. Keay, T. Klein, M. Koch, R. Kondo, C. Kong, A. Kosto, A. Lawrence, D. Liefer, M. Lollie, D. Lucchetti, N. K. Lysne, C. Lytle, C. MacPherson, A. Malm, S. Mather, B. Mathewson, D. Maxwell, L. McCaffrey, H. McDougall, R. Mendoza, M. Mills, R. Morrison, L. Narmour, N. Nguyen, L. Nugent, S. Olson, D. Ouellette, J. Parks, Z. Peters, J. Petricka, J. M. Pino, F. Polito, M. Preidl, G. Price, T. Proctor, M. Pugh, N. Ratcliff, D. Raymondson, P. Rhodes, C. Roman, C. Roy, C. Ryan-Anderson, F. B. Sanchez, G. Sangiolo, T. Sawadski, A. Schaffer, P. Schow, J. Sedlacek, H. Semenenko, P. Shevchuk, S. Shore, P. Siegfried, K. Singhal, S. Sivarajah, T. Skripka, L. Sletten, B. Spaun, R. T. Sprenkle, P. Stoufer, M. Tader, S. F. Taylor, T. H. Thompson, R. Tobey, A. Tran, T. Tran, G. Vittorini, C. Volin, J. Walker, S. White, D. Wilson, Q. Wolf, C. Wringe, K. Young, J. Zheng, K. Zuraski, C. H. Baldwin, A. Chernoguzov, J. P. Gaebler, S. J. Sanders, B. Neyenhuis, R. Stutz, and J. G. Bohnet, Helios: A 98-qubit trapped-ion quantum computer (2025).

[11] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. S. L. Brandao, D. A. Buell, B. Burkett, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, A. Dunsworth, E. Farhi, B. Foxen, A. Fowler, C. Gidney, M. Giustina, R. Graff, K. Guerin, S. Habegger, M. P. Harrigan, M. J. Hartmann, A. Ho, M. Hoffmann, T. Huang, T. S. Humble, S. V. Isakov, E. Jeffrey, Z. Jiang, D. Kafri, K. Kechedzhi, J. Kelly, P. V. Klimov, S. Knysh, A. Korotkov, F. Kostritsa, D. Landhuis, M. Lindmark, E. Lucero, D. Lyakh, S. Mandrà, J. R. McClean, M. McEwen, A. Megrant, X. Mi, K. Michielsen, M. Mohseni, J. Mutus, O. Naaman, M. Neeley, C. Neill, M. Y. Niu, E. Ostby, A. Petukhov, J. C. Platt, C. Quintana, E. G. Rieffel, P. Roushan, N. C. Rubin, D. Sank, K. J. Satzinger, V. Smelyanskiy, K. J. Sung, M. D. Trevithick, A. Vainsencher, B. Villalonga, T. White, Z. J. Yao, P. Yeh, A. Zalcman, H. Neven, and J. M. Martinis, Quantum supremacy using a programmable superconducting processor, Nature **574**, 505–510 (2019).

[12] H.-S. Zhong, H. Wang, Y.-H. Deng, M.-C. Chen, L.-C. Peng, Y.-H. Luo, J. Qin, D. Wu, X. Ding, Y. Hu, P. Hu, X.-Y. Yang, W.-J. Zhang, H. Li, Y. Li, X. Jiang, L. Gan, G. Yang, L. You, Z. Wang, L. Li, N.-L. Liu, C.-Y. Lu, and J.-W. Pan, Quantum computational advantage using photons, Science **370**, 1460 (2020), https://www.science.org/doi/pdf/10.1126/science.abe8770.

[13] Y. Kim, A. Eddins, S. Anand, K. X. Wei, E. van den Berg, S. Rosenblatt, H. Nayfeh, Y. Wu, M. Zaletel, K. Temme, and A. Kandala, Evidence for the utility of quantum computing before fault tolerance, Nature **618**, 500–505 (2023).

[14] J. Robledo-Moreno, M. Motta, H. Haas, A. Javadi-Abhari, P. Jurcevic, W. Kirby, S. Martiel, K. Sharma, S. Sharma, T. Shirakawa, I. Sitdikov, R.-Y. Sun, K. J. Sung, M. Takita, M. C. Tran, S. Yunoki, and A. Mezzacapo, Chemistry beyond the scale of exact diagonalization on a quantum-centric supercomputer, Science Advances **11**, eadu9991 (2025), https://www.science.org/doi/pdf/10.1126/sciadv.adu9991.

[15] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, Quantum computational chemistry, Rev. Mod. Phys. **92**, 015003 (2020).

[16] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin,

S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, *et al.*, Variational quantum algorithms, Nature Reviews Physics **3**, 625 (2021).

[17] M. Larocca, S. Thanasilp, S. Wang, K. Sharma, J. Biamonte, P. J. Coles, L. Cincio, J. R. McClean, Z. Holmes, and M. Cerezo, Barren plateaus in variational quantum computing, Nature Reviews Physics **7**, 174–189 (2025).

[18] H. Mhiri, R. Puig, S. Lerch, M. S. Rudolph, T. Chotibut, S. Thanasilp, and Z. Holmes, A unifying account of warm start guarantees for patches of quantum landscapes (2025).

[19] M. Cerezo, M. Larocca, D. García-Martín, N. L. Diaz, P. Braccia, E. Fontana, M. S. Rudolph, P. Bermejo, A. Ijaz, S. Thanasilp, E. R. Anschuetz, and Z. Holmes, Does provable absence of barren plateaus imply classical simulability?, Nature Communications **16**, 10.1038/s41467-025-63099-6 (2025).

[20] S. Aaronson and Y. Zhang, On verifiable quantum advantage with peaked circuit sampling, arXiv preprint arXiv:2404.14493 (2024).

[21] X. Mi and K. Kechedzhi, A verifiable quantum advantage, `https://research.google/blog/a-verifiable-quantum-advantage/` (2025), google Quantum AI blog post related to "Observation of constructive interference at the edge of quantum ergodicity", Nature 646, 825–830 (2025).

[22] A. Gheorghiu, M. J. Hoban, and E. Kashefi, A simple protocol for fault tolerant verification of quantum computation, Quantum Science and Technology **4**, 015009 (2018).

[23] J. F. Fitzsimons and E. Kashefi, Unconditionally verifiable blind quantum computation, Physical Review A **96**, 012303 (2017).

[24] C. Gustiani, D. Leichtle, J. Miller, R. Grassie, D. Mills, and E. Kashefi, On-chip verified quantum computation with an ion-trap quantum processing unit, Phys. Rev. Lett. , (2025).

[25] B. Polacchi, D. Leichtle, G. Carvacho, G. Milani, N. Spagnolo, M. Kaplan, E. Kashefi, and F. Sciarrino, Experimental verifiable multiclient blind quantum computing on a qline architecture, Phys. Rev. Lett. **134**, 200603 (2025).

[26] P. Drmota, D. Nadlinger, D. Main, B. Nichol, E. Ainley, D. Leichtle, A. Mantri, E. Kashefi, R. Srinivas, G. Araneda, *et al.*, Verifiable blind quantum computing with trapped ions and single photons, Physical Review Letters **132**, 150604 (2024).

[27] D. Leichtle, L. Music, E. Kashefi, and H. Ollivier, Verifying bqp computations on noisy devices with minimal overhead, PRX Quantum **2**, 040302 (2021).

[28] U. Maurer and R. Renner, Abstract cryptography, in *The Second Symposium on Innovations in Computer Science, ICS 2011*, edited by B. Chazelle (Tsinghua University Press, 2011) pp. 1–21.

[29] R. Babbush, R. King, S. Boixo, W. Huggins, T. Khattar, G. H. Low, J. R. McClean, T. O'Brien, and N. C. Rubin, The grand challenge of quantum applications (2025).

[30] D. A. Abanin, R. Acharya, L. Aghababaie-Beni, G. Aigeldinger, A. Ajoy, R. Alcaraz, I. Aleiner, T. I. Andersen, M. Ansmann, F. Arute, K. Arya, A. Asfaw, N. Astrakhantsev, J. Atalaya, R. Babbush, D. Bacon, B. Ballard, J. C. Bardin, C. Bengs, A. Bengtsson, A. Bilmes, S. Boixo, G. Bortoli, A. Bourassa, J. Bovaird, D. Bowers, L. Brill, M. Broughton, D. A. Browne, B. Buchea, B. B. Buckley, D. A. Buell, T. Burger, B. Burkett, N. Bushnell, A. Cabrera, J. Campero, H.-S. Chang, Y. Chen, Z. Chen, B. Chiaro, L.-Y. Chih, D. Chik, C. Chou, J. Claes, A. Y. Cleland, J. Cogan, S. Cohen, R. Collins, P. Conner, W. Courtney, A. L. Crook, B. Curtin, S. Das, L. De Lorenzo, D. M. Debroy, S. Demura, M. Devoret, A. Di Paolo, P. Donohoe, I. Drozdov, A. Dunsworth, C. Earle, A. Eickbusch, A. M. Elbag, M. Elzouka, C. Erickson, L. Faoro, E. Farhi, V. S. Ferreira, L. F. Burgos, E. Forati, A. G. Fowler, B. Foxen, S. Ganjam, G. Garcia, R. Gasca, É. Genois, W. Giang, C. Gidney, D. Gilboa, R. Gosula, A. G. Dau, D. Graumann, A. Greene, J. A. Gross, H. Gu, S. Habegger, J. Hall, I. Hamamura, M. C. Hamilton, M. Hansen, M. P. Harrigan, S. D. Harrington, S. Heslin, P. Heu, O. Higgott, G. Hill, J. Hilton, S. Hong, H.-Y. Huang, A. Huff, W. J. Huggins, L. B. Ioffe, S. V. Isakov, J. Iveland, E. Jeffrey, Z. Jiang, X. Jin, C. Jones, S. Jordan, C. Joshi, P. Juhas, A. Kabel, D. Kafri, H. Kang, A. H. Karamlou, K. Kechedzhi, J. Kelly, T. Khaire, T. Khattar, M. Khezri, S. Kim, R. King, P. V. Klimov, A. R. Klots, B. Kobrin, A. N. Korotkov, F. Kostritsa, R. Kothari, J. M. Kreikebaum, V. D. Kurilovich, E. Kyoseva, D. Landhuis, T. Lange-Dei, B. W. Langley, P. Laptev, K.-M. Lau, L. Le Guevel, J. Ledford, J. Lee, K. Lee, Y. D. Lensky, S. Leon, B. J. Lester, W. Y. Li, A. T. Lill, W. Liu, W. P. Livingston, A. Locharla, E. Lucero, D. Lundahl, A. Lunt, S. Madhuk, F. D. Malone, A. Maloney, S. Mandrà, J. M. Manyika, L. S. Martin, O. Martin, S. Martin, Y. Matias, C. Maxfield, J. R. McClean, M. McEwen, S. Meeks, A. Megrant, X. Mi, K. C. Miao, A. Mieszala, Z. Minev, R. Molavi, S. Molina, S. Montazeri, A. Morvan, R. Movassagh, W. Mruczkiewicz, O. Naaman, M. Neeley, C. Neill, A. Nersisyan, H. Neven, M. Newman, J. H. Ng, A. Nguyen, M. Nguyen, C.-H. Ni, M. Y. Niu, L. Oas, T. E. O'Brien, W. D. Oliver, A. Opremcak, K. Ottosson, A. Petukhov, A. Pizzuto, J. Platt, R. Potter, O. Pritchard, L. P. Pryadko, C. Quintana, G. Ramachandran, C. Ramanathan, M. J. Reagor, J. Redding, D. M. Rhodes, G. Roberts, E. Rosenberg, E. Rosenfeld, P. Roushan, N. C. Rubin, N. Saei, D. Sank, K. Sankaragomathi, K. J. Satzinger, A. Schmidhuber, H. F. Schurkus, C. Schuster, T. Schuster, M. J. Shearn, A. Shorter, N. Shutty, V. Shvarts, V. Sivak, J. Skruzny, S. Small, V. Smelyanskiy, W. C. Smith, R. D. Somma, S. Springer, G. Sterling, D. Strain, J. Suchard, P. Suchsland, A. Szasz, A. Sztein, D. Thor, E. Tomita, A. Torres, M. M. Torunbalci, A. Vaishnav, J. Vargas, S. Vdovichev, G. Vidal, B. Villalonga, C. V. Heidweiller, S. Waltman, S. X. Wang, B. Ware, K. Weber, T. Weidel, T. Westerhout, T. White, K. Wong, B. W. K. Woo, C. Xing, Z. J. Yao, P. Yeh, B. Ying, J. Yoo, N. Yosri, G. Young, A. Zalcman, C. Zhang, Y. Zhang, N. Zhu, and N. Zobrist, Observation of constructive interference at the edge of quantum ergodicity, Nature **646**, 825 (2025).

[31] M. Inajetovic, P. Wallden, and A. Pappa, Verifiable end-to-end delegated variational quantum algorithms (2025).

[32] S. Barz, E. Kashefi, A. Broadbent, J. F. Fitzsimons, A. Zeilinger, and P. Walther, Demonstration of blind quantum computing, Science **335**, 303 (2012), https://www.science.org/doi/pdf/10.1126/science.1214707.

[33] Y. Li and S. C. Benjamin, Efficient variational quantum simulator incorporating active error minimization, Phys. Rev. X **7**, 021050 (2017).

[34] K. Temme, S. Bravyi, and J. M. Gambetta, Error mitigation for short-depth quantum circuits, Physical review letters **119**, 180509 (2017).

[35] S. Endo, S. C. Benjamin, and Y. Li, Practical quantum error mitigation for near-future applications, Phys. Rev. X **8**, 031027 (2018).

[36] B. Yang, R. Raymond, and S. Uno, Efficient quantum readout-error mitigation for sparse measurement outcomes of near-term quantum devices, Physical Review A **106**, 012423 (2022).

[37] S. Endo, Z. Cai, S. C. Benjamin, and X. Yuan, Hybrid quantum-classical algorithms and quantum error mitigation, Journal of the Physical Society of Japan **90**, 032001 (2021).

[38] Z. Cai, R. Babbush, S. C. Benjamin, S. Endo, W. J. Huggins, Y. Li, J. R. McClean, and T. E. O'Brien, Quantum error mitigation, Reviews of Modern Physics **95**, 045005 (2023).

[39] T. Peng, A. W. Harrow, M. Ozols, and X. Wu, Simulating large quantum circuits on a small quantum computer, Physical review letters **125**, 150504 (2020).

[40] X. Yuan, J. Sun, J. Liu, Q. Zhao, and Y. Zhou, Quantum simulation with hybrid tensor networks, Physical Review Letters **127**, 040501 (2021).

[41] H. Harada, Y. Suzuki, B. Yang, Y. Tokunaga, and S. Endo, Density matrix representation of hybrid tensor networks for noisy quantum devices, Quantum **9**, 1823 (2025).

[42] B. Yang, N. Yoshioka, H. Harada, S. Hakkaku, Y. Tokunaga, H. Hakoshima, K. Yamamoto, and S. Endo, Resource-efficient generalized quantum subspace expansion, Phys. Rev. Appl. **23**, 054021 (2025).

[43] A. Broadbent, J. Fitzsimons, and E. Kashefi, Universal blind quantum computation, in *2009 50th Annual IEEE Symposium on Foundations of Computer Science* (IEEE, 2009).

[44] D. Gottesman and I. L. Chuang, Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations, Nature **402**, 390–393 (1999).

[45] R. Raussendorf and H. J. Briegel, A one-way quantum computer, Phys. Rev. Lett. **86**, 5188 (2001).

[46] E. Knill, R. Laflamme, and G. J. Milburn, A scheme for efficient quantum computation with linear optics, Nature **409**, 46–52 (2001).

[47] V. Danos, E. Kashefi, and P. Panangaden, The measurement calculus, J. ACM **54**, 8–es (2007).

[48] H. J. Briegel, D. E. Browne, W. Dür, R. Raussendorf, and M. Van den Nest, Measurement-based quantum computation, Nature Physics **5**, 19–26 (2009).

[49] V. Dunjko, J. F. Fitzsimons, C. Portmann, and R. Renner, Composable security of delegated quantum computation (2014).

[50] E. Knill, Fault-tolerant postselected quantum computation: Threshold analysis (2004).

[51] O. Kern, G. Alber, and D. L. Shepelyansky, Quantum error correction of coherent errors by randomization, The European Physical Journal D **32**, 153–156 (2005).

[52] C. Dankert, R. Cleve, J. Emerson, and E. Livine, Exact and approximate unitary 2-designs and their application to fidelity estimation, Phys. Rev. A **80**, 012304 (2009).

[53] M. R. Geller and Z. Zhou, Efficient error models for fault-tolerant architectures and the pauli twirling approximation, Phys. Rev. A **88**, 012314 (2013).

[54] J. J. Wallman and J. Emerson, Noise tailoring for scalable quantum computation via randomized compiling, Phys. Rev. A **94**, 052325 (2016).

[55] T. Kapourniotis, E. Kashefi, D. Leichtle, L. Music, and H. Ollivier, Asymmetric secure multi-party quantum computation with weak clients against dishonest majority, Quantum Science and Technology **10**, 025015 (2025).

[56] T. Kapourniotis, E. Kashefi, D. Leichtle, L. Music, and H. Ollivier, Unifying quantum verification and error-detection: theory and tools for optimisations, Quantum Science and Technology **9**, 035036 (2024).