# Compiled Nonlocal Games from any Trapdoor Claw-Free Function

Kaniuar Bacho[*1], Alexander Kulpe[†1], Giulio Malavolta[‡2], Simon Schmidt[§1], Michael Walter[¶1]

[1]Ruhr University Bochum
[2]Bocconi University

**Abstract**

A recent work of Kalai et al. (STOC 2023) shows how to compile any multi-player nonlocal game into a protocol with a single computationally-bounded prover. Subsequent works have built on this to develop new cryptographic protocols, where a completely classical client can verify the validity of quantum computation done by a quantum server. Their compiler relies on the existence of quantum fully-homomorphic encryption.

In this work, we propose a new compiler for transforming nonlocal games into single-prover protocols. Our compiler is based on the framework of measurement-based quantum computation. It can be instantiated assuming the existence of *any* trapdoor function that satisfies the claw-freeness property. Leveraging results by Natarajan and Zhang (FOCS 2023) on compiled nonlocal games, our work implies the existence of new protocols to classically verify quantum computation from potentially weaker computational assumptions than previously known.

[*]kano-b@hotmail.de
[†]alexander.kulpe@rub.de
[‡]giulio.malavolta@unibocconi.it
[§]s.schmidt@rub.de
[¶]michael.walter@rub.de

# Contents

# 1 Introduction

A nonlocal game consists of two (or more) non-communicating players interacting with a referee. The game starts with the referee sampling a question for each player, to which they reply with an answer. The referee then decides if the players win or lose based on some publicly computable predicate on the question/answer tuples. Crucially, players are not allowed to communicate during the execution of the protocol, and so they are unaware of the questions/answers of the other players. However, the rules of the game are fixed ahead of time and the players are free to decide on a strategy that maximizes their success probability. Typically, one considers the settings where players are either fully classical, or they are allowed to perform local quantum computations on a (possibly entangled) shared quantum state.

Nonlocal games were introduced in the study of the foundations of quantum mechanics, where a celebrated theorem of Bell [Bel64] showed the existence of a nonlocal game $\mathcal{G}$ in which the maximum success probability of classical players (known as the *classical value* $\omega_c(\mathcal{G})$ of the game) is strictly smaller than the maximum success probability of quantum players (known as the *quantum value* $\omega_q(\mathcal{G})$ of the game). This shows that there exists an experiment that detects a difference between classical and quantum correlations. Ever since, nonlocal games have become an object of study in other disciplines, such as mathematics [Slo17] and computer science [CHTW04, RUV13, CGJV19, Gri19, JNV+21].

Although the presence of multiple non-communicating players seems necessary to prove anything meaningful in the information-theoretic settings, a recent work by Kalai, Lombardi, Vaikuntanathan, and Yang [KLVY23] (henceforth, KLVY) has introduced a general method for converting any $k$-player nonlocal game into a *single-player* game, if the prover is assumed to be computationally bounded. In order to achieve this, they rely on cryptographic assumptions: The basic idea is to ask a single player to simulate the computation of all players in the original nonlocal game and, to ensure that no communication is happening, the question of each player is encrypted under a different key. Clearly, in order for correctness to hold, the player must be able to compute on the encrypted question, which is the reason why the KLVY compiler relies on the existence of *quantum* fully-homomorphic encryption (QFHE) [Mah18a, Bra18].

Such *compiled nonlocal games* have then been shown to be a useful primitive for applications: Natarajan and Zhang [NZ23] showed how to use the compiled version of the CHSH game [CHSH69] as an alternative path for constructing *classical verification of quantum computation*, and a more recent work showed a protocol with a *succinct verifier* [MNZ24], improving on prior work [BKL+22] that relied on stronger cryptographic assumptions. Subsequent work focused on bounding the quantum value of more general classes of nonlocal games [CMM+24, BVB+24, MPW24]. A recent work [KMP+24] established a bound on the quantum value of *all* compiled nonlocal games. Compiled nonlocal games promise a modular framework for constructing quantum cryptographic protocols: One can concentrate on the information-theoretic multi-player setting, which is typically easier and already has a large body of literature, and then simply compile the resulting protocol into a single-player one. Cryptography should take care of the rest.

However, at present we know of only a *single* recipe to transform nonlocal games into compiled ones, i.e., the aforementioned KLVY protocol, which relies on a rather strong cryptographic primitive, namely the existence of quantum homomorphic encryption (QFHE). From a cryptographic perspective, QFHE is a rather strong primitive, both in terms of functional guarantees and in terms of the underlying computational assumptions. There is evidence that the functionality offered by QFHE is not necessary for the applications of compiled nonlocal games and it is therefore natural

to wonder whether this structure is necessary at all for compiled nonlocal games (a more detailed discussion on this is in Section 1.3). Taking a step back, we find this situation unsatisfactory, since it underpins a lack of understanding on this cryptographic process and furthermore it places compiled nonlocal games on potentially thin cryptographic foundations.

The goal of our work is to improve our understanding of this cryptographic process and place compiled nonlocal games on more solid cryptographic foundations. Motivated by this, we investigate alternative compilation methods, based on potentially weaker cryptographic assumptions.

### 1.1 Our Results

In this work, we propose a new approach for compiling multi-player nonlocal games into single-player *compiled* nonlocal games. Our compiler only assumes the existence of a *family of trapdoor claw-free functions* (TCFs), which consists of a sequence of function pairs $(f_0, f_1)$ that are easy to compute, and can be inverted given the trapdoor, while without the trapdoor it is computationally hard to find a *claw*, i.e., two inputs $x_0$ and $x_1$ such that $f_0(x_0) = f_1(x_1)$. Our contributions can be summarized with the following statement.

**Theorem** (Informal)**.** *Let $\mathcal{G}$ be a nonlocal game. If there exists a family of TCFs, then there exists a (quantumly) sound single-player compiled game $\mathcal{G}_{\text{comp}}$.*

We refer to a compiled game as being *quantumly sound* if we can bound the quantum value of $\mathcal{G}_{\text{comp}}$ based only on the properties of $\mathcal{G}$. To achieve this, we show that our compilation strategy achieves a similar "blindness" property as the KLVY compiler (see Lemma 5.4 for an exact definition). As a consequence of this, known approaches to analyze the KLVY compiler also apply to our protocol.

For instance, as a corollary of [KMP+24], we obtain a bound on the quantum value of our compiler for all nonlocal games. Furthermore, invoking the analysis from [NZ23], we obtain a new protocol for classical verification of any BQP computation from any TCF. Crucially, we only require the TCF to satisfy the *claw-freeness* property, whereas prior protocols not based on compiled nonlocal games [Mah18b, GV19] required stronger assumptions, such as the adaptive hardcore bit property (we defer a detailed comparison with existing protocols to Section 1.3). For instance, we obtain the first classical verification for BQP from the *extended linear hidden shift problem*, a conjectured hard problem in isogeny-based cryptography [AMR22]. We summarize the result of combining our approach with the [NZ23] analysis in the following corollary.

**Corollary** (Informal)**.** *If there exists a family of TCFs, then there exists a protocol for BQP verification with a classical verifier.*

At a technical level, our approach is inspired by the work of Gheorghiu and Vidick [GV19], and combines measurement-based quantum computation (MBQC) with a remote state preparation (RSP) protocol. To achieve our goal, we modify the blind quantum computation protocol from [BFK09], allowing the client to operate on an arbitrary server's state, and furthermore to continue the computation "in the plain" after the protocol is concluded. We refer to such a variant as a *half-blind* quantum computation. This protocol achieves information-theoretic security, but it relies on the ability of the client to prepare single-qubit states. In order to make the protocol fully classical, we design a new *blind* RSP protocol that allows the client to delegate the preparation of

such single-qubit state to the server. The main challenge here is proving the blindness of our protocol assuming only the *claw-freeness* of the TCF, which is a search problem. For further details, we refer the reader to Section 1.2.

Overall, our approach provides a modular and fully self-contained method for compiling nonlocal games, based on weak cryptographic primitives, yielding new protocols from weaker computational assumptions than prior work.

### 1.2 Technical Outline

To set some context for our ideas, we first recall the basics of the KLVY compiler [KLVY23], for simplicity focusing on the case of two-player nonlocal games. The protocol consists of an interaction between a (computationally bounded) prover $P$ and classical verifier $V$, and proceeds as follows:

- The verifier samples two questions $(x, y)$ from the underlying (two-player) nonlocal game, and then sends a QFHE encryption of Alice's question $\mathrm{Enc}(x)$ to the prover.
- The prover responds to the verifier with an encrypted answer $\alpha = \mathrm{Enc}(a)$ (in the honest case, it would homomorphically evaluate Alice's POVM).
- The verifier decrypts $\alpha$ to recover $a$, then sends $y$ to the prover in the plain.
- The prover outputs a response $b$ (in the honest case, this is Bob's response).
- The verifier holds a transcript $(x, y, a, b)$ and can determine whether the prover won or not, by simply evaluating the predicate of the nonlocal game.

The intuition is that, since the first question given to the prover is encrypted, it cannot influence the subsequent question-response phase, in any detectable manner. In other words, the encryption forces the single prover to behave "non-locally". Our main observation is that the full-power of QFHE is not necessary to achieve this property, and it can be substituted by much weaker cryptographic machinery, if one is willing to (i) increase the number of rounds of interaction, and (ii) let the verifier's (classical) computation grow with the size of the (quantum) computation performed by the prover. Under these two relaxations, we describe our solution next.

**Half-Blind Quantum Computation.** Let us first consider an easier version of the problem, where we allow the verifier to send qubits to the prover (we will soon see how to remove this assumption). Then a natural idea would be to substitute the QFHE with the *information-theoretically* secure universal blind quantum computing (UBQC) protocol of [BFK09]. Although a naive application of this technique would not work in our setting, let us first recall the basics of the UBQC protocol to gain some context.

The UBQC protocol allows a verifier to harness the power of a quantum prover by applying an $n \times n$ unitary $U$ to the state $|+\rangle^{\otimes n}$, while ensuring that the prover obtains no information about $U$. As a first step, the verifier prepares and sends multiple states in

$$|+_\theta\rangle := \frac{1}{\sqrt{2}}(|0\rangle + e^{i\theta}\,|1\rangle)$$

for uniformly random $\theta \in \{k \cdot \pi/4 \mid k = 0, \ldots, 7\}$ to the prover, which are then entangled in a specific manner using CZ operators, resulting in a highly entangled *resource state*. Then individual

qubits are then measured successively (and adaptively) in a basis chosen by the verifier. At the end of the protocol, the prover holds the state $U|+\rangle^{\otimes n}$, up to some local Pauli operators. Glossing over some details, the role of the verifier is to help the prover preparing the resource state, which he does by sending to the prover states of the form $|+_\theta\rangle$. The *secret angles* $\theta$ are kept private by the verifier. For the remainder of the computation, the verifier assists the prover by sending blinded versions of such angles, which is a fully classical information.

The UBQC protocol gives us almost what we want, except for two problems: (i) The initial state of the computation should not be $|+\rangle^{\otimes n}$ but rather an arbitrary prover-chosen state $|\psi\rangle$, and furthermore (ii) the computation should only affect a subset of the registers of $|\psi\rangle$ (corresponding to Alice's subsystem), whereas the complement should remain untouched, since Bob's computation must happen in the plain. In other words, we want to implement the computation corresponding to $(U \otimes I)|\psi\rangle_{\mathsf{AB}}$, and the Pauli correction errors should appear only in the $\mathsf{A}$ registers.

To solve these problems, we consider a modified version of the UBQC protocol, that we refer to as *half-blind* quantum computation (HBQC) achieving precisely this. In a nutshell, we extend the regular resource state with a series of "dummy" states, that can be thought of implementing the identity. The prover is instructed to entangle their state $|\psi\rangle$ with this state and then proceed to measure them in a fixed basis. Effectively, this allows the prover to *teleport* the state $|\psi\rangle$ onto the original resource state, enabling to continue the computation as in the regular UBQC protocol. To solve the second problem, we simply observe that nothing stops the prover from keeping some entangled state on a separate register and our analysis shows that the Pauli errors do not propagate to this subsystem, thus enabling Bob to finish the computation.

**Making the Verifier Classical.** We now discuss how to make the verifier completely classical. As we can see from the above protocol, the only quantum capability that we assume from the verifier is the ability to prepare random $|+_\theta\rangle$ state and send them to the prover. Thus, a natural idea to make the verifier fully classical, is to resort to *remote state preparation* (RSP) protocols, such as [GV19]. While it is certainly possible to use existing RSP protocols in our context, their security (even the weaker blindness guarantee) rely on the adaptive hardcore bit property of TCFs, a strong assumption that we wish to avoid. So the remaining challenge is to build an RSP protocol, only assuming the claw-freeness property of a TCF.

Our first observation is that we can slightly relax the functionality of the RSP and furthermore that a rather weak blindness property will suffice. Specifically, we will instead delegate the preparation of states of the form

$$Z^b|+_\theta\rangle = Z^b\frac{1}{\sqrt{2}}(|0\rangle + e^{i\theta}|1\rangle)$$

for a random bit $b \in \{0,1\}$ and $\theta \in \{k \cdot \pi/4 \mid k = 0,\ldots,7\}$, both known by the verifier. For blindness, we require that, given the transcript of the RSP protocol, the angle $\theta$ is computationally indistinguishable from a uniformly sampled $\theta^* \leftarrow\!\!\$\ \{k \cdot \pi/4 \mid k = 0,\ldots,7\}$. Importantly, the bit $b$ is never revealed to the distinguisher, and it is instead kept private by the verifier. Omitting some details, we mention here that the extra phase flip $Z^b$ will be *absorbed* in the Pauli errors of the HBQC protocol, which ensures that it is kept hidden from the prover.

Our proof is inspired by the proof technique from [BGKM+23], introduced in the context of test of qubits, although our protocol is, to the best of our knowledge, new. We present an informal description of the protocol next. As the first step, we ask the prover, starting from an arbitrary

one-qubit state $\alpha|0\rangle + \beta|1\rangle$, to prepare a claw state of the form

$$\alpha|0, x_0\rangle + \beta|1, x_1\rangle \quad \text{where} \quad f_0(x_0) = f_1(x_1) = y,$$

which can be efficiently computed. The verifier then sends two random bit strings $r_0$ and $r_1$, and the prover applies the isometry

$$\alpha|0, x_0\rangle + \beta|1, x_1\rangle \mapsto \alpha|0, x_0, -(x_0 \cdot r_0)\rangle + \beta|1, x_1, x_1 \cdot r_1\rangle = \alpha|0, x_0, -z_0\rangle + \beta|1, x_1, z_1\rangle,$$

where the third register has dimension $n$ (chosen below) and $x_i \cdot r_i \in \{0, 1\}$ is computed modulo two. The prover then applies the quantum Fourier transform $\mathrm{QFT}_n$ on the third register to obtain

$$\frac{1}{\sqrt{n}} \sum_{d' \in \mathbb{Z}_n} (\omega_n^{-d' \cdot z_0} \alpha|0, x_0\rangle + \omega_n^{d' \cdot z_1} \beta|1, x_1\rangle)|d'\rangle.$$

The prover measures the last register to obtain an outcome $d' \in \mathbb{Z}_n$. To ensure a proper distribution of the angle, we actually want it to be the case that $d' = 1$ (otherwise, it is not hard to see that the angle may belong to a subgroup, and thus will not be uniformly distributed). Our idea to achieve this is simply to use post-selection, i.e., aborting the execution if this event does not occur. For small enough $n$, this will happen with constant probability. Thus, we can henceforth assume that our state is of the form

$$\alpha|0, x_0\rangle + \omega_n^{z_0 + z_1} \beta|1, x_1\rangle.$$

up to a global phase. Afterwards, the prover measures the second register in the Hadamard basis, yielding the outcome $d$, and is left with

$$\alpha|0\rangle + \beta(-1)^b \omega_n^\theta |1\rangle,$$

where $(b, \theta) = (d \cdot (x_0 \oplus x_1), x_0 \cdot r_0 + x_1 \cdot r_1)$ can be efficiently calculated by the verifier using the trapdoor information to recover $x_0$ and $x_1$.

We are almost done, except that now the sum of $z_0 + z_1$ is done modulo $n$, whereas in order to appeal to the standard Goldreich-Levin theorem, we would like it to happen modulo 2.[1] The key idea is to the repeat the protocol three times (with independently sampled TCFs), with $n = 2$, $n = 4$, and $n = 8$, starting with $|+\rangle$ as the initial input state and using the output quantum state as the new input state for the next round. This results in a state of the form:

$$\begin{aligned}\frac{1}{\sqrt{2}}(|0\rangle + (-1)^{b_1 \oplus b_2 \oplus b_3} \omega_2^{\theta_1} \omega_4^{\theta_2} \omega_8^{\theta_3} |1\rangle) &= \frac{1}{\sqrt{2}}(|0\rangle + (-1)^{b_1 \oplus b_2 \oplus b_3} \omega_8^{4\theta_1 + 2\theta_2 + \theta_3} |1\rangle) \\ &= \frac{1}{\sqrt{2}}(|0\rangle + (-1)^b \omega_8^\theta |1\rangle).\end{aligned}$$

The blindness of our protocol follows from rewriting $\theta$ as $4\theta_1' + 2\theta_2' + \theta_3'$, where the $\theta_i' \in \{0, 1\}$ contains the term $x_0 \cdot r_0 \oplus x_1 \cdot r_1$, padded with independently sampled variables. Starting from $\theta_1'$, we sequentially appeal to the quantum Goldreich-Levin theorem [AC01, CLLZ21] to substitute $\theta_i'$ with a uniformly random bit, which must be indistinguishable, otherwise there would exist an extractor outputting $(x_0, x_1)$, i.e., breaking the claw-freeness of the TCF.

[1] Note that simply fixing $n = 2$ does not work either, since in that case we do not range over all the desired angles; we only obtain 0 or $\pi$.

**Soundness.** We briefly discuss why our compilation strategy is sound. We prove a general soundness result for all nonlocal games (in the limit of the security parameter), by showing that our protocol satisfies the necessary condition to appeal to the recent result of [KMP+24]. Loosely speaking, all needs to be shown is that the internal state of the prover at the end of the interaction with the verifier is computationally independent from Alice's question. This follows immediately by the blindness of the classical HBQC protocol. A similar argument connects our compiler with the result of [NZ23], thus yielding a classical verification of quantum computation protocol from any TCF. We omit most details here, and we refer the reader to the technical sections.

### 1.3 Related Work

The work that is technically the closest to ours is the aforementioned KLVY compiler [KLVY23], which relies on the existence of QFHE. While this is technically incomparable with the existence of TCFs, we can discuss concrete instances to compare the underlying computational assumptions. To the best of our knowledge, there are two approaches to build QFHE (with a classical client): One assuming the hardness of the learning with errors (LWE) problem [Mah18a, Bra18] and one assuming the existence of indistinguishability obfuscation plus any dual-mode[2] TCF [GV24]. This means that, prior to our work, compiled nonlocal games were known to exist under either of these two sets of assumptions.

On the other hand, TCFs can be constructed assuming either LWE [BKVV20], cryptographic group actions [AMR22], the Quadratic Residuosity problem, or the Decisional Diffie-Hellman problem [KCVY22]. In particular, this means that we obtain compiled nonlocal games under any of the above computational assumptions, which is a much broader set than what was previously known.

We also compare our classical verification protocol, that we obtain by combining our protocol with [NZ23], with existing approaches not explicitly based on nonlocal games. To the best of our knowledge, all existing protocols [Mah18b, GV19] are based on TCFs with the *adaptive hardcore bit* property. The latter is a stronger (decisional) assumption compared to the plain claw-freeness, that requires the indistinguishability between a random bit and some adversarially chosen predicate computed on a claw. To exemplify this difference, it suffices to recall that proving the adaptive hardcore bit property of TCFs based on cryptographic group actions [AMR22] requires a new non-standard variant of the XOR-Lemma, whose proof is currently an open problem. On the other hand, the claw-freeness of the same construction can be proven using the extended LHS assumption (in fact, the search version of it).

We also mention, as an exception to the claim made above, a recent work of Brakerski et al. [BGKM+23], that shows how to classically test qubits from any TCFs. Although it is conceivable that their protocol can be extended to verify any quantum computation, as shown in [Vid20], the known approach to prove soundness requires the TCF to at least satisfy the dual-mode property (see [Vid20] for details).

To summarize, we are not aware, prior to our work, of any classical verification protocol for BQP that only relies on the claw-freeness of the TCF, without assuming any extra property.

Finally, we mention a recent work by Arora et al. [ABCC24] that proposes a compiler for contextuality games, a generalization of nonlocal games. Although we do not explore the extension in the present work, we expect that our approach can be generalized to contextuality games as well,

[2] A TCF is *dual mode* if there exists a computationally indistinguishable family of function pairs where claws do not exist, i.e., the two functions have disjoint domains.

combining our protocol with the techniques from [ABCC24].

## 2 Preliminaries

We denote the security parameter by $\lambda \in \mathbb{N}$. A function $\mathsf{negl}$ is called *negligible* if it vanishes faster than the absolute value of any inverse polynomial. Unless explicitly pointed out, the inner product of two bit strings $a, b \in \{0,1\}^n$ of length $n$ is defined as

$$a \cdot b := \bigoplus_{i=1}^{n} a_i \cdot b_i \in \{0,1\},$$

where $a_i$ and $b_i$ refer to the $i$-th bit of the strings, respectively. Given two bit strings $r_0$ and $r_1$, we denote their concatenation by $r_0 \parallel r_1$. We denote by $\omega_n = e^{2\pi i/n}$ the $n$-th root of unity. For $a, b \in \mathbb{Z}$, the notation $[\![a, b]\!]$ is used to indicate the set $\{a, a+1, a+2, \ldots, b\}$. Moreover, we define $[n] := [\![1, n]\!] = \{1, \ldots, n\}$ for an integer $n$. We use the notation $x \leftarrow \mu$ to denote that $x$ is drawn from a probability distribution $\mu$. We define the set

$$\Theta := \{k \cdot \pi/4 \mid k = 0, \ldots, 7\},$$

since we will be working extensively with it.

### 2.1 Quantum Information

In quantum mechanics, physical systems are often identified with Hilbert spaces $\mathcal{H}$, and the states of the system are identified with positive semidefinite operators (PSD) $\rho$ with unit trace, called *density operators*. A state is called *pure* if the density operator has rank one, and otherwise it is called *mixed*. Any unit vector $|v\rangle \in \mathcal{H}$ determines a pure state by the formula $\rho = |v\rangle\langle v|$, and conversely any pure state can be written in this way, hence the two concepts are often identified.

A *measurement* with a finite outcome set $\mathcal{O}$ is described by a collection of bounded operators $\{A_a\}_{a\in\mathcal{O}}$ acting on $\mathcal{H}$ such that $\sum_{a\in\mathcal{O}} A_a^\dagger A_a = I$. If the system is in state $\rho$, then the probability of obtaining outcome $a$ is given by $p(a) = \mathrm{tr}(A_a^\dagger A_a \rho)$, after which the state of the system is described by $A_a \rho A_a^\dagger / p(a)$. The probabilities of measurement outcomes only depend on the operators $M_a := A_a^\dagger A_a$. A collection of operators $\{M_a\}_{a\in\mathcal{O}}$ such as these which satisfy $\sum_{a\in\mathcal{O}} M_a = I$ is called a *POVM*, which is short for positive operator-valued measure, with outcomes in $\mathcal{O}$. *Observables* are self-adjoint elements $B = B^\dagger \in \mathcal{B}(H)$, and their *quantum expectation value* with respect to the state $\rho$ is given by $\mathrm{tr}(\rho B)$. This can be related to the preceding if one takes $\mathcal{O}$ to be the set of eigenvalues of $B$ (assuming it is finite) and $A_a$ as the corresponding spectral projections. We will often discuss apparatuses with multiple measurement settings, labeled by some index set $\mathcal{I}$, but the same set of outcomes $\mathcal{O}$ for each setting. This will be denoted by $\{\{M_{xa}\}_{a\in\mathcal{O}} : x \in \mathcal{I}\}$, where $\{M_{xa}\}_{a\in\mathcal{O}}$ is a POVM (or measurement) with outcomes in $\mathcal{O}$ for each $x \in \mathcal{I}$. We often abbreviate and write this as $\{M_{xa}\}_{a\in\mathcal{O},x\in\mathcal{I}}$ when clear from context. A *subnormalized state* is a PSD operator with trace less than or equal to 1 (in the case of pure states, it corresponds to a pure state with norm less than or equal to 1). Operationally, this corresponds to post-selecting on some measurement outcome, without renormalizing the state.

Throughout this work, we denote the usual Pauli operators by

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \text{ and } Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

Moreover, we denote the controlled $X$, the controlled $Z$, the Hadamard and $T$ gate by

$$\mathrm{CX}, \mathrm{CZ}, H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \text{ and } T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}.$$

**Quantum Computing.** A *quantum circuit* is a unitary operator that operates on the Hilbert space $\mathcal{H} = (\mathbb{C}^2)^{\otimes k}$ for some number $k \in \mathbb{N}$ of qubits and is given by the composition of unitary gates, each acting on only one or two qubits (taken from some fixed universal gate set). The size of a quantum circuit is the number of gates used in that circuit. The qubits are typically split into input qubits and auxiliary qubits, which are assumed to be initialized in the $|0\rangle$ state unless stated otherwise. If a classical outcome is desired, a subset of the qubits is measured after the unitary circuit has been applied. A *quantum polynomial-time (QPT) algorithm* consists of a family of quantum circuits $\{C_n\}_{n\in\mathbb{N}}$ and a deterministic polynomial-time Turing machine that on input $1^n$ outputs a description of $C_n$.

A family of POVMs $\{\{\Pi_{n,i}\}_{i\in I_n}\}_{n\in\mathbb{N}}$ is *QPT-implementable* if there exists a QPT algorithm with quantum circuits $\{C_n\}_{n\in\mathbb{N}}$ such that $C_n$ realizes the POVM $\{\Pi_{n,i}\}_{i\in I_n}$, i.e., measuring some output qubits and post-processing gives rise to the same probabilities as the POVM.

A *probabilistic polynomial-time (PPT) algorithm* is a probabilistic Turing machine with a polynomial time bound, meaning that there exists a polynomial $p$ such that for every input $x \in \{0,1\}^*$ the machine halts after at most $p(|x|)$ steps. Any PPT algorithm can be converted into a QPT algorithm (with $C_\lambda$ a quantum circuit with $\lambda$ input qubits that when given as input $|x\rangle$ and if a suitable number of qubits is measured, implements the same behavior as the PPT algorithm on any bitstring $x$ of length $|x| = \lambda$).

We say that two variables $X_0$ and $X_1$ are *computationally indistinguishable* if for any QPT-implementable algorithm, the probability that the algorithm returns 1 on input $X_0$ is negligibly close to the probability that the algorithm returns 1 on input $X_1$. We often abbreviate computational indistinguishability by $X_0 \approx_c X_1$.

**Quantum Goldreich-Levin.** We recall the quantum Goldreich-Levin theorem [AC01], specifically the version with auxiliary input that was proven in [CLLZ21].

**Theorem 2.1** (Quantum Goldreich-Levin [CLLZ21])**.** *If there exists a quantum algorithm that, given a random $r$ and an auxiliary quantum input $\rho_x$ for random $x$, computes $r\cdot x$ with probability at least $1/2+\varepsilon$; then there exists a quantum algorithm that takes $\rho_x$ and extracts $x$ with probability $4\varepsilon^2$.*

## 2.2 Trapdoor Claw-Free Functions

We recall the definition of a trapdoor claw-free function (TCF). The definition that we use in this work is taken mostly from [BGKM+23].

**Definition 2.2** (Trapdoor Claw-Free Function)**.** *Let $\lambda$ be the security parameter. A* trapdoor claw-free function (TCF) *consists of a family of injective function pairs $(f_{0,\lambda}, f_{1,\lambda})$ and finite sets $\mathcal{X}_\lambda$ and $\mathcal{Y}_\lambda$ with*

$$\{f_{b,\lambda} : \mathcal{X}_\lambda \to \mathcal{Y}_\lambda\}_{(b,\lambda)\in\{0,1\}\times\mathbb{N}},$$

*where we omit the subscript $\lambda$ when it is clear from the context. Additionally, a TCF pair is augmented with two algorithms.*

- $\mathsf{Gen}(1^\lambda)$: *On input the security parameter in unary* $1^\lambda$*, the polynomial-time generation algorithm outputs a function pair* $(f_0, f_1)$ *and a trapdoor* $\mathsf{td}$.
- $\mathsf{Invert}(\mathsf{td}, y)$: *On input an image* $y \in \mathcal{Y}$ *and the trapdoor* $\mathsf{td}$*, the polynomial-time deterministic inversion algorithm returns two preimages* $(x_0, x_1)$.

*We require a TCF to satisfy the following properties:*

- *(Correctness) For all* $\lambda \in \mathbb{N}$*, all* $x \in \mathcal{X}$*, and all* $b \in \{0,1\}$*, it holds that:*

$$f_0(x_0) = f_1(x_1) = y \quad \text{where } (x_0, x_1) \leftarrow \mathsf{Invert}(\mathsf{td}, f_b(x)) \text{ and } ((f_0, f_1), \mathsf{td}) \leftarrow \mathsf{Gen}(1^\lambda).$$

- *(Efficient Superposition) There exists a QPT algorithm that, on input the description of the functions* $(f_0, f_1)$*, prepares the state*

$$\frac{1}{\sqrt{|\mathcal{X}|}} \sum_{x \in \mathcal{X}} |x\rangle .$$

- *(Claw-Freeness) For all QPT algorithms* $A^*$ *there exists a negligible function* $\mathsf{negl}$ *such that for all* $\lambda \in \mathbb{N}$ *it holds that:*

$$\Pr\left[(x_0^*, x_1^*) \leftarrow A^*(f_0, f_1) : f_0(x_0^*) = f_1(x_1^*)\right] \leq \mathsf{negl}(\lambda).$$

*where* $((f_0, f_1), \mathsf{td}) \leftarrow \mathsf{Gen}(1^\lambda)$.

In addition to the above properties, it will also be convenient for us to assume that one can efficiently check membership in $\mathcal{Y}$ given the trapdoor. This is without loss of generality since one can always do so by running the inversion algorithm and checking if it is successful. We will also assume that there exists an embedding of the set $\mathcal{X}$ into the bitstrings $\{0,1\}^{p(\lambda)}$ for some fixed polynomial $p$. Such an embedding always exists.

TCFs can be constructed from a variety of computational assumptions, such as the Quadratic Residuosity or the Decisional Diffie-Hellman problem [KCVY22]. In [BCM+18], a variant called *noisy* TCF was constructed based on the Learning with Errors (LWE) problem. This was extended to the Ring-LWE assumption in [BKVV20]. It is easy to verify that all the results in this work also apply to the case of noisy TCFs; however, for notational convenience, we describe our protocols using regular TCFs. Finally, TCFs were recently constructed based on general cryptographic group actions, such as isogenies on elliptic curves, in [AMR22].

### 2.3 Measurement-Based Quantum Computation

In the following, we recall the basics of the *measurement-based quantum computation* (MBQC) model [RB01]. Loosely speaking, in MBQC, one prepares a highly entangled *resource state* (independent of the computation to be carried out) and then successively (and adaptively) measuring individual qubits in an appropriate basis so that the remaining qubits are in the desired quantum state at the end. The MBQC protocol can be divided into the following procedures:

1. (State Preparation) In the first step, we are given qubits in the quantum state $|+\rangle$, which we entangle in a specific way using the CZ operator to build the resource state.

2. (Computation) In the next step, we perform one-qubit measurements on almost all qubits in a fixed order and in a specific basis, which depends on the previous measurement outcomes. These measurements can be viewed as implementing a unitary operation such that the remaining qubits are in the desired quantum state.

Thus, the resource state and the one-qubit measurements are all that are needed to perform arbitrary quantum computations.

**Universal Blind Quantum Computation.** The *universal blind quantum computation* (UBQC) protocol [BFK09] considers the setting where a client delegates a quantum computation to a server that possesses the required quantum computational resources. The only requirement for the client is that he is able to prepare specific single-qubit states, which he will then send to the server. The protocol relies on a universal resource state, referred to as the brickwork state, which we recall in the following.

**Definition 2.3** ([BFK09, Definition 1])**.** *A* brickwork state $\mathcal{G}_{n\times m}$*, where* $m \equiv 5 \mod 8$*, is an entangled state of* $n \times m$ *qubits constructed as follows (see Fig. 1 for an example):*

1. *Prepare all qubits in state* $|+\rangle$ *and assign to each qubit an index* $(i, j)$*,* $i$ *being a row* $(i \in [n])$ *and* $j$ *being a column* $(j \in [m])$*.*
2. *For each row, apply the operator* CZ *on qubits* $(i, j)$ *and* $(i, j + 1)$ *where* $j \in [m - 1]$*.*
3. *For each column* $j \equiv 3 \mod 8$ *and each odd row* $i$*, apply the operator* CZ *on qubits* $(i, j)$ *and* $(i + 1, j)$ *and also on qubits* $(i, j + 2)$ *and* $(i + 1, j + 2)$*.*
4. *For each column* $j \equiv 7 \mod 8$ *and each even row* $i$*, apply the operator* CZ *on qubits* $(i, j)$ *and* $(i + 1, j)$ *and also on qubits* $(i, j + 2)$ *and* $(i + 1, j + 2)$*.*

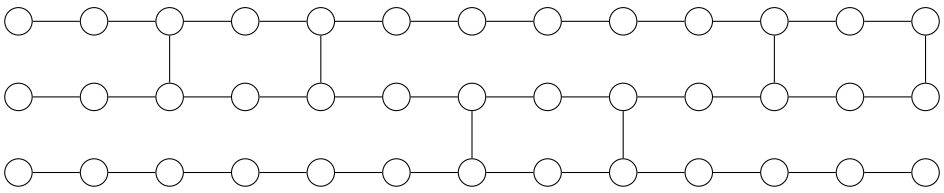

Figure 1: The brickwork state $\mathcal{G}_{3\times 13}$. The circles represent qubits in the $|+\rangle$ state, and the edges represent the CZ operator applied to both connected qubits (note that the CZ operator is symmetric, meaning it does not matter which qubit is the control and which is the target).

As a notational convention, $n$ always represents the number of rows, and $m$ always represents the number of columns/layers. We define

$$|+_\theta\rangle := \frac{1}{\sqrt{2}}\left(|0\rangle + e^{i\theta}\,|1\rangle\right)$$

and

$$|-_\theta\rangle := \frac{1}{\sqrt{2}}\left(|0\rangle - e^{i\theta}\,|1\rangle\right)$$

for an arbitrary $\theta \in \mathbb{R}$. We say that we measure a qubit in the $\theta$-basis if we measure in the basis $\{|+_\theta\rangle, |-_\theta\rangle\}$. This is the same as saying that we are measuring the binary observable

$$|+_\theta\rangle\langle+_\theta| - |-_\theta\rangle\langle-_\theta| = \begin{pmatrix} 0 & e^{i\theta} \\ e^{-i\theta} & 0 \end{pmatrix}.$$

Now, the following theorem states that we can essentially perform arbitrary quantum computations using a brickwork state and one-qubit measurements. Moreover, the proof explains how to concretely implement this MBQC procedure.

**Theorem 2.4** (Universality [BFK09, Theorem 1])**.** *The brickwork state $\mathcal{G}_{n\times m}$ is universal for quantum computation. Furthermore, we only require single-qubit measurements under angles in $\Theta$, and measurements can be done layer-by-layer.*

Let us briefly describe the internal workings of the proof to establish the necessary notation. The proof of the theorem relies on the well-known fact that $\{\mathrm{CX}, H, T\}$ is a universal gate set, meaning that all unitaries $U'$ can be efficiently approximated by a unitary $U$, which can be expressed as a finite sequence of gates from the aforementioned set.

The proof of Theorem 2.4 shows that for any arbitrary unitary $U'$, there exists a measurement pattern $\{\phi_{x,y}\}_{x\in[n],y\in[m-1]}$ that implements a unitary $U$ which approximates $U'$. A *measurement pattern* for $\mathcal{G}_{n\times m}$ consists of a series of angles $\phi_{x,y} \in \Theta$, one for each position $(x, y) \in [n] \times [m-1]$. The measurement of the brickwork state begins with the leftmost column and proceeds from top to bottom, i.e., starting by measuring the qubit at position $(1, 1)$, then $(2, 1)$, up to $(n, 1)$, before moving on to the next column, and so on. During the MBQC procedure, the actual measurement angle $\phi'_{x,y}$ at position $(x, y)$ depends on $\phi_{x,y}$ and the outcomes of previous measurements.

The rule to update the angles is described in [DK06]. First, define $f : [n] \times [m-1] \to [n] \times [m]$ by $f(x, y) = (x, y + 1)$. In terms of the brickwork state, this function maps a qubit to the qubit directly to its right. Next, define the *$X$-dependencies* of the qubit at position $(x, y)$ by the set

$$D_{x,y} := f^{-1}(x, y) = \begin{cases} \emptyset & \text{if } y = 1 \\ \{(x, y-1)\} & \text{if } y > 1 \end{cases} \qquad \forall x \in [n], y \in [m]$$

and the *$Z$-dependencies* by

$$D'_{x,y} := \{(a, b) \mid b < y, (x, y) \in N(f(a, b))\} \qquad \forall x \in [n], y \in [m],$$

where $N(x, y)$ denotes the set of neighbours of $(x, y)$ in the brickwork state — all vertices that are connected to $(x, y)$. We will measure the brickwork state in the order described earlier (from left to right and top to bottom). Denote the measurement outcome at position $(x, y)$ by $s_{x,y}$. We define

$$s^X_{x,y} := \bigoplus_{i\in D_{x,y}} s_i = \begin{cases} 0 & \text{if } y = 1 \\ s_{x,y-1} & \text{if } y > 1 \end{cases} \qquad \forall x \in [n], y \in [m]$$

and

$$s^Z_{x,y} := \bigoplus_{i\in D'_{x,y}} s_i \qquad \forall x \in [n], y \in [m].$$

Finally, we define the modified measurement angles by

$$\phi'_{x,y} = (-1)^{s^X_{x,y}} \cdot \phi_{x,y} + s^Z_{x,y} \cdot \pi.$$

Thus, $\phi'_{x,y}$ depends on the outcomes of at most two previous layers. Using these as the actual measurement angles in the MBQC procedure leads to the quantum state

$$\left(X^{s^X_{1,m}} Z^{s^Z_{1,m}} \otimes \ldots \otimes X^{s^X_{n,m}} Z^{s^Z_{n,m}}\right) U \left|+\right\rangle^{\otimes n}$$

at the end.

# 3 Blind Remote State Preparation

## 3.1 Definition

In the following, we provide a definition of a remote state preparation (RSP) protocol for the special case of states of the form

$$\left|+_\theta\right\rangle := \frac{1}{\sqrt{2}}(\left|0\right\rangle + e^{i\theta}\left|1\right\rangle),$$

where $\theta \in \Theta$.

**Definition 3.1** (Remote State Preparation)**.** *A* remote state preparation (RSP) *protocol consists of a pair of interactive algorithms $(V, P)$, with the security parameter in unary $1^\lambda$ as input: A classical probabilistic polynomial-time algorithm $V$, called the* verifier*, and a quantum polynomial-time algorithm $P$, called the* prover*. We require the protocol to satisfy the following properties:*

- *(Correctness) The protocol successfully terminates with a probability of at least $1/\mathsf{poly}(\lambda)$, which is inverse-polynomial in the security parameter. Furthermore, upon successful completion, the honest prover $P$ holds the state*

  $$Z^b \left|+_\theta\right\rangle$$

  *for some bit $b \in \{0,1\}$ and angle $\theta \in \Theta$. On the other hand, the verifier holds the pair $(b, \theta)$.*

- *(Blindness) Consider the following experiment $\mathrm{Exp}(1^\lambda, V, P^*)$ played between an honest verifier $V$ and a possibly malicious prover $P^*$.*

  - *The players engage in the interactive RSP protocol. If the protocol does not terminate successfully, the experiment aborts.*
  - *Let $(b, \theta)$ be the output of $V$.*
  - *The verifier flips a coin $c \leftarrow\!\!\$\ \{0,1\}$. If $c = 0$, $V$ sets $\theta' := \theta$, otherwise $V$ samples a uniform $\theta' \leftarrow\!\!\$\ \Theta$.*
  - *$V$ sends $\theta'$ to $P^*$, who returns a bit $c'$.*
  - *The experiment outputs 1 if $c' = c$ and 0 otherwise.*

*We say that an RSP protocol is blind if for all QPT adversaries $P^*$ there exists a negligible function* $\mathsf{negl}$ *such that for all $\lambda \in \mathbb{N}$ it holds that:*

$$\Pr\left[\mathrm{Exp}(1^\lambda, V, P^*) = 1 \mid \text{no abort}\right] \leq \frac{1}{2} + \mathsf{negl}(\lambda).$$

### 3.2 Our Protocol

We present here our main RSP protocol, assuming the existence of any TCF $(\mathsf{Gen}, \mathsf{Invert})$, as described in Section 2.2. Remember that there always exists an embedding of the set $\mathcal{X}$ into the bitstrings $\{0,1\}^{p(\lambda)}$ for some fixed polynomial $p$. We start with describing a subroutine that the prover $P$ and the verifier $V$ will run in our main protocol.

**Subroutine.** The input and output of the protocol are:

- (Input) The protocol is parameterized by the security parameter in unary $1^\lambda$, an integer $n$ and the prover $P$ holds a state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$.
- (Output) At the end of the interaction, the verifier holds a pair $(b, \theta) \in \{0,1\} \times \{0,1,2\}$ and the prover holds the state $\alpha|0\rangle + \beta(-1)^b \omega_n^\theta |1\rangle$.

The interaction between $P$ and $V$ proceeds as follows:

- (Verifier 1st Message) Sample $((f_0, f_1), \mathsf{td}) \leftarrow \mathsf{Gen}(1^\lambda)$ and send $(f_0, f_1)$ to $P$.
- (Prover 1st Message) Prepare the state

$$|\psi\rangle \otimes \frac{1}{\sqrt{|\mathcal{X}|}} \sum_{x \in \mathcal{X}} |x\rangle = \frac{1}{\sqrt{|\mathcal{X}|}} \sum_{x \in \mathcal{X}} \alpha|0,x\rangle + \beta|1,x\rangle .$$

  Then, apply the isometric mapping that evaluates $f_b$ coherently on input the second register, with the function controlled on the first register, to obtain the state

$$\frac{1}{\sqrt{|\mathcal{X}|}} \sum_{x \in \mathcal{X}} \alpha|0,x,f_0(x)\rangle + \beta|1,x,f_1(x)\rangle .$$

  Measure the last register to obtain some $y \in \mathcal{Y}$, with the residual state being

$$\alpha|0,x_0\rangle + \beta|1,x_1\rangle$$

  where $f_0(x_0) = f_1(x_1) = y$. Send $y$ to $V$.
- (Verifier 2nd Message) Check if $y \in \mathcal{Y}$ and abort if not. Sample two strings $r_0, r_1 \leftarrow\!\!\$\ \{0,1\}^{p(\lambda)}$ uniformly at random. Send $(r_0, r_1)$ to $P$.
- (Prover 2nd Message) Consider the isometric mapping

$$M : (b, x_b) \mapsto (b, x_b, (-1)^{1-b}(x_b \cdot r_b))$$

  where the inner product $z_b := x_b \cdot r_b \in \{0,1\}$ is computed over $\mathbb{Z}_2$ and then parsed as an element of $\mathbb{Z}_n$. Apply $M$ to the current state to compute

$$\alpha|0,x_0,-(x_0 \cdot r_0)\rangle + \beta|1,x_1,x_1 \cdot r_1\rangle = \alpha|0,x_0,-z_0\rangle + \beta|1,x_1,z_1\rangle .$$

  Apply $\mathrm{QFT}_n$ to the last register to obtain

$$\frac{1}{\sqrt{n}} \sum_{d' \in \mathbb{Z}_n} (\omega_n^{-d' \cdot z_0} \alpha|0,x_0\rangle + \omega_n^{d' \cdot z_1} \beta|1,x_1\rangle)|d'\rangle ,$$

where $-d' \cdot z_0, d' \cdot z_1 \in \mathbb{Z}_n$. Measure the last register in the computational basis and abort if the output $d' \neq 1$. The state becomes

$$\omega_n^{-z_0}\alpha\,|0, x_0\rangle + \omega_n^{z_1}\beta\,|1, x_1\rangle \equiv \alpha\,|0, x_0\rangle + \omega_n^{z_0+z_1}\beta\,|1, x_1\rangle\,.$$

Conditioning on not aborting, measure the second register in the Hadamard basis to obtain some $d \in \{0,1\}^{p(\lambda)}$, and return the state

$$\alpha\,|0\rangle + \beta(-1)^{d\cdot(x_0\oplus x_1)}\omega_n^{z_0+z_1}\,|1\rangle\,.$$

Send $d$ to $V$.

- (Verifier Output) Recompute $(x_0, x_1) \leftarrow \mathsf{Invert}(\mathsf{td}, y)$ and set $b := d \cdot (x_0 \oplus x_1)$ and $\theta := z_0 + z_1 = x_0 \cdot r_0 + x_1 \cdot r_1 \in \{0, 1, 2\}$, where the sum is computed over $\mathbb{Z}$.

**Main Protocol.** Our main protocol uses the above defined subroutine and proceeds in three steps.

- Run the above protocol with $n = 2$ and set $|+\rangle$ to be $P$'s input state. Let $(b_1, \theta_1)$ be the output of $V$, and let $|\psi_1\rangle$ be the output of $P$.
- Run the above protocol with $n = 4$ and set $|\psi_1\rangle$ to be $P$'s input state. Let $(b_2, \theta_2)$ be the output of $V$, and let $|\psi_2\rangle$ be the output of $P$.
- Run the above protocol with $n = 8$ and set $|\psi_2\rangle$ to be $P$'s input state. Let $(b_3, \theta_3)$ be the output of $V$, and let $|\psi_3\rangle$ be the output of $P$.

The prover $P$ returns the final state $|\psi_3\rangle$, whereas the verifier $V$ sets

$$b := b_1 \oplus b_2 \oplus b_3 \text{ and } \theta := 4\theta_1 + 2\theta_2 + \theta_3 \mod 8$$

and must multiply $\theta$ by $\pi/4$ to obtain the angle.

We are now in the position of analyzing the protocol and we start with correctness.

**Theorem 3.2.** *The RSP protocol as described above is correct.*

*Proof.* First, observe that the probability that all three subprotocols do not abort is 1/64, and consequently so is the success probability of our RSP. Starting with the initial state $|+\rangle$, we can track the evolution of the state. The first iteration implements the mapping

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \mapsto \frac{1}{\sqrt{2}}(|0\rangle + (-1)^{b_1}\omega_2^{\theta_1}\,|1\rangle),$$

whereas the second results into the state

$$\frac{1}{\sqrt{2}}(|0\rangle + (-1)^{b_1}\omega_2^{\theta_1}\,|1\rangle) \mapsto \frac{1}{\sqrt{2}}(|0\rangle + (-1)^{b_1\oplus b_2}\omega_2^{\theta_1}\omega_4^{\theta_2}\,|1\rangle),$$

and finally, from the last iteration, we obtain

$$\begin{aligned}\frac{1}{\sqrt{2}}(|0\rangle + (-1)^{b_1\oplus b_2}\omega_2^{\theta_1}\omega_4^{\theta_2}\,|1\rangle) &\mapsto \frac{1}{\sqrt{2}}(|0\rangle + (-1)^{b_1\oplus b_2\oplus b_3}\omega_2^{\theta_1}\omega_4^{\theta_2}\omega_8^{\theta_3}\,|1\rangle)\\ &= \frac{1}{\sqrt{2}}(|0\rangle + (-1)^{b_1\oplus b_2\oplus b_3}\omega_8^{4\theta_1+2\theta_2+\theta_3}\,|1\rangle)\\ &= \frac{1}{\sqrt{2}}(|0\rangle + (-1)^{b}\omega_8^{\theta}\,|1\rangle),\end{aligned}$$

as desired. □

Finally, we prove that the protocol satisfies blindness.

**Theorem 3.3.** *If* $(\mathsf{Gen}, \mathsf{Invert})$ *is a claw-free TCF, then the protocol as described above satisfies blindness.*

*Proof.* We use the following simple fact: For $a_1, a_2, a_3 \in \{0,1\}$, we have

$$a_1 + a_2 + a_3 = 2 \cdot \mathrm{MSB}(a_1 + a_2 + a_3) + (a_1 \oplus a_2 \oplus a_3),$$

where MSB returns the most significant bit of the number represented in binary using two bits, which is well-defined since the sum is in $\{0,1,2,3\}$. The proof follows directly by considering the integer $a_1 + a_2 + a_3$ in its binary representation using two bits.
Using this, the following equations are true over $\mathbb{Z}$:

$$\begin{aligned} 4 \cdot \theta_1 + 2 \cdot \theta_2 + \theta_3 &= 4 \cdot (z_{1,0} + z_{1,1}) + 2 \cdot (z_{2,0} + z_{2,1}) + (z_{3,0} + z_{3,1}) \\ &= 4 \cdot (z_{1,0} + z_{1,1}) + 2 \cdot (z_{2,0} + z_{2,1} + \tilde{z}_3) + (z_{3,0} \oplus z_{3,1}) \\ &= 4 \cdot (z_{1,0} + z_{1,1} + \tilde{z}_2) + 2 \cdot (z_{2,0} \oplus z_{2,1} \oplus \tilde{z}_3) + (z_{3,0} \oplus z_{3,1}), \end{aligned}$$

where

$$\tilde{z}_3 := \mathrm{MSB}(z_{3,0} + z_{3,1}) \text{ and } \tilde{z}_2 := \mathrm{MSB}(z_{2,0} + z_{2,1} + \tilde{z}_3).$$

Now, we have

$$\begin{aligned} \theta &= 4 \cdot \theta_1 + 2 \cdot \theta_2 + \theta_3 \mod 8 \\ &= 4 \cdot (z_{1,0} + z_{1,1} + \tilde{z}_2) + 2 \cdot (z_{2,0} \oplus z_{2,1} \oplus \tilde{z}_3) + (z_{3,0} \oplus z_{3,1}) \mod 8 \\ &= 4 \cdot (z_{1,0} \oplus z_{1,1} \oplus \tilde{z}_2) + 2 \cdot (z_{2,0} \oplus z_{2,1} \oplus \tilde{z}_3) + (z_{3,0} \oplus z_{3,1}) \mod 8 \\ &= 4 \cdot (z_{1,0} \oplus z_{1,1} \oplus \tilde{z}_2) + 2 \cdot (z_{2,0} \oplus z_{2,1} \oplus \tilde{z}_3) + (z_{3,0} \oplus z_{3,1}) \\ &= 4 \cdot \theta'_1 + 2 \cdot \theta'_2 + \theta'_3. \end{aligned}$$

We now gradually change the way we compute $\theta$ in the blindness experiments through a hybrid argument. First, we claim that the following distributions are computationally indistinguishable:

$$\theta = 4\theta'_1 + 2\theta'_2 + \theta'_3 \approx_c 4\theta^*_1 + 2\theta'_2 + \theta'_3$$

where $\theta^*_1 \leftarrow\!\!\$\ \{0,1\}$. Recall that

$$\theta'_1 = z_{1,0} \oplus z_{1,1} \oplus \tilde{z}_2 = x_{1,0} \cdot r_{1,0} \oplus x_{1,1} \cdot r_{1,1} \oplus \tilde{z}_2.$$

Since $\tilde{z}_2$ is independent from

$$x_{1,0} \cdot r_{1,0} \oplus x_{1,1} \cdot r_{1,1} = (x_{1,0} \parallel x_{1,1}) \cdot (r_{1,0} \parallel r_{1,1}),$$

it suffices to show that the latter is computationally indistinguishable from uniform. This follows by the Theorem 2.1 (Quantum Goldreich-Levin), as otherwise there would exist an efficient extractor for $(x_{1,0} \parallel x_{1,1})$, contradicting the claw-freeness of the TCF. Repeating the same argument, we can conclude that

$$\theta \approx_c 4\theta^*_1 + 2\theta'_2 + \theta'_3 \approx_c 4\theta^*_1 + 2\theta^*_2 + \theta'_3 \approx_c 4\theta^*_1 + 2\theta^*_2 + \theta^*_3,$$

where $\theta^*_1, \theta^*_2, \theta^*_3 \leftarrow\!\!\$\ \{0,1\}$. This completes our proof. □

**Boosting Correctness.** Our protocol as described above succeeds with constant probability. As standard in this context, we can increase the success probability to be exponentially close to 1 by repeating the protocol sequentially a number of times polynomial in the security parameter $\lambda$. Security follows by a union bound. Therefore, we will henceforth assume that our RSP protocol succeeds with probability $1 - \mathsf{negl}(\lambda)$.

**Remark 3.4.** *Note that we can use the same strategy to remotely construct states in $|+_\theta\rangle$ for $\theta \leftarrow\!\!\$ \{k \cdot \pi/2^{m-1} \mid k \in \mathbb{Z}_{2^m}\}$ for any $m \in O(1)$. Simply follow the same steps in the main protocol, beginning with $n = 2$ and ending with $n = 2^m$. The blindness proof is very similar; one simply needs to extend the fact at the beginning of the blindness proof to an arbitrary number of bits $a_i$ and use their binary representation to replace the sums with their XORs, so that the quantum Goldreich-Levin theorem can be applied again.*

# 4 Half-Blind Quantum Computation

In the standard UBQC protocol (Section 2.3), the client requests the server to apply an $n \times n$ unitary $U$ to the fixed quantum state $|+\rangle^{\otimes n}$. In this section, we generalize this to the setting where the unitary is applied to an arbitrary quantum state of the server (that is possibly entangled with some internal register of the server). More formally, we want to blindly implement the computation $(U \otimes I)|\psi\rangle$, where $|\psi\rangle$ is an arbitrary state held by the server. Henceforth, we refer to this task as *half-blind quantum computation* (HBQC).

## 4.1 Half-Blind Quantum Computation

Our first observation is that, instead of using the $|+\rangle^{\otimes n}$ state as the first layer in the brickwork state, we can use any $n$-qubit quantum state $|\psi\rangle$, which would then result in $U|\psi\rangle$ (up to some local Pauli operators). This fact is well-known in the MBQC literature and was used for instance in [MDF17], in the context of cluster states. Furthermore, if $|\psi\rangle$ consists of more than $n$ qubits and the first $n$ qubits are used in the MBQC procedure, then the overall computation is simply $(U \otimes I)|\psi\rangle$ (up to some local Pauli operators), as the MBQC procedure implements gates independently of the input state.

The remaining challenge is to show how the client hides his measurement angles. To address this, we extend the regular brickwork state $\mathcal{G}_{n\times m}$, where we use our measurement pattern, to a larger brickwork state $\mathcal{G}_{n\times(m+8)}$ by introducing eight layers of dummy $|+\rangle^{\otimes n}$ states between the input layer and the remaining $m-1$ layers (see Figs. 2 and 3 for examples). The first eight layers are then measured in the 0-basis to implement the identity, and the remaining qubits are measured according to the original measurement pattern. Note that, after these eight layers, the server measures qubits that were prepared by the client with the injected randomness. In this procedure, loosely speaking, we teleport the $|\psi\rangle$ state to the point where the randomness has already been injected, which has the same effect as directly injecting the randomness into $|\psi\rangle$. While we could achieve this with just two additional layers instead of eight (as two layers already implement an identity gate), using eight layers preserves the topology of the brickwork state, making the write-up more convenient.

**Our HBQC Protocol.** We formally describe our protocol in the following, and we refer the reader to Sections 2 and 2.3 for background and notational conventions. The input and output of

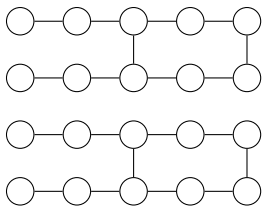

Figure 2: The brickwork state $\mathcal{G}_{4\times 5}$.

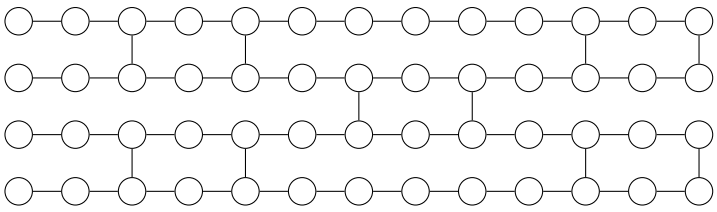

Figure 3: The brickwork state $\mathcal{G}_{4\times 13}$.

the protocol are:

- (Input) The client $V$ has an $n$-qubit unitary map $U$, represented as a sequence of measurement angles $\{\phi_{x,y} \in \Theta\}_{x\in[n],y\in[m-1]}$ of a measurement-based quantum computation over a brickwork state $\mathcal{G}_{n\times m}$.
  The server $P$ inputs the first $n$ qubits of a quantum state $|\psi\rangle$.
- (Output) At the end of the interaction, the client holds the measurement outcome of measuring the first $n$ qubits of $(U \otimes I)|\psi\rangle$ in the standard basis and the server holds the post-measurement state of the remaining qubits.

For the interaction, we define $m' := m + 8$ and the new measurement pattern

$$\begin{aligned} \varphi_{x,y} &:= 0 && \forall x \in [n], y \in [8] \\ \varphi_{x,y} &:= \phi_{x,y-8} && \forall x \in [n], y \in [\![9, m'-1]\!] \end{aligned}$$

for the larger brickwork state $\mathcal{G}_{n\times m'}$. The interaction between $V$ and $P$ proceeds as follows:

- (State Preparation)
  1. For the column $y = 1$, and each row $x \in [n]$, $P$ uses his input qubits (the first $n$ qubits from his quantum state $|\psi\rangle$).
  2. For each column $y \in [\![2, 8]\!]$, and each row $x \in [n]$, $P$ creates qubits in the $|+\rangle$ state.
  3. For each column $y \in [\![9, m'-1]\!]$, and each row $x \in [n]$, $V$ prepares the state $\left|+_{\theta_{x,y}}\right\rangle$, where $\theta_{x,y} \leftarrow\!\!\$\ \Theta$, and sends the qubit to $P$.
  4. For the column $y = m'$, and each row $x \in [n]$, $P$ creates qubits in the $|+\rangle$ state, which are used as the final output layer.
  5. $P$ entangles the qubits by applying CZ operators between the pairs of qubits specified by the pattern of the brickwork state $\mathcal{G}_{n\times m'}$.
- (Computation)

  For column $y = 1, \ldots, 8$:
  For row $x = 1, \ldots, n$:
  1. $V$ computes the updated measurement angle $\varphi'_{x,y}$, to take previous measurement outcomes received from $P$ into account.
  2. $V$ transmits $\delta_{x,y} := \varphi'_{x,y}$ to $P$.
  3. $P$ measures in the $\delta_{x,y}$-basis and transmits the result $b_{x,y} \in \{0, 1\}$ to $V$.

4. $V$ sets $s_{x,y} := b_{x,y}$.

For column $y = 9, \ldots, m' - 1$:
For row $x = 1, \ldots, n$:

1. $V$ computes the updated measurement angle $\varphi'_{x,y}$, to take previous measurement outcomes received from $P$ into account.
2. $V$ computes $\delta_{x,y} := \varphi'_{x,y} - \theta_{x,y} + r_{x,y}\pi$, where $r_{x,y} \leftarrow\!\!\$\ \{0,1\}$, and transmits it to $P$.
3. $P$ measures in the $\delta_{x,y}$-basis and transmits the result $b_{x,y} \in \{0,1\}$ to $V$.
4. $V$ calculates $s_{x,y} := b_{x,y} \oplus r_{x,y}$.

- (Measurement)

1. $P$ measures the remaining $n$ qubits in the standard basis and sends the outcome $a' \in \{0,1\}^n$ to $V$.
2. $V$ computes the actual outcome $a := \left(s^X_{1,m'} \parallel \ldots \parallel s^X_{n,m'}\right) \oplus a'$.

**Correctness.** We prove that the protocol implements the desired functionality.

**Theorem 4.1.** *The HBQC protocol as described above is correct, i.e., if both parties honestly follow the protocol, the output will be correct.*

*Proof.* The measurement pattern $\{\phi_{x,y}\}_{x\in[n],y\in[m-1]}$ implements, by definition, the unitary $U$. The 0-basis measurements in the first eight layers implement an identity gate (see Figure 5 in [BFK09]), meaning that $\{\varphi_{x,y}\}_{x\in[n],y\in[m'-1]}$ still implements $U$.

We now argue that the added randomness in the measurement angles $\delta$ and the quantum states $|+_\theta\rangle$ cancels out during the computation, so that we still perform the same quantum computation according to the standard MBQC procedure. We define the *Z-rotation by an angle* $\theta \in \mathbb{R}$ as

$$R_Z(\theta) := \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}.$$

Note that the CZ operator commutes with both $R_Z(\theta) \otimes I$ and $I \otimes R_Z(\theta)$, as all are diagonal matrices (in the standard basis). Thus, the state preparation phase is equivalent to first preparing the brickwork state and then applying the $Z$-rotations to the specific qubits, rather than doing it the other way around. Moreover, a $\varphi'$-basis measurement on a state $|\gamma\rangle$ is the same as a $(\varphi' - \theta)$-basis measurement on a state $R_Z(\theta)\,|\gamma\rangle$.

In the protocol, we measure in the $\delta$-basis, where $\delta := \varphi' - \theta + r\pi$. If $r = 0$, $P$'s measurement has the same effect as $V$'s target $\varphi'$-basis measurement; if $r = 1$, all $V$ needs to do is flip the outcome to get again the target $\varphi'$-basis measurement, since $R_Z(r\pi) = Z^r$ and $Z\,|+_\theta\rangle = |-_\theta\rangle$. This shows that the protocol yields the same outcome as directly using the MBQC procedure with the measurement pattern $\{\varphi_{x,y}\}_{x\in[n],y\in[m'-1]}$, without any added randomness. Therefore, after the computation phase, the server holds the quantum state $(U' \otimes I)\,|\psi\rangle$, where

$$U' := \left(X^{s^X_{1,m'}} Z^{s^Z_{1,m'}} \otimes \ldots \otimes X^{s^X_{n,m'}} Z^{s^Z_{n,m'}}\right) U.$$

Obtaining the outcome $a'$ by measuring the first $n$ qubits of $(U' \otimes I)|\psi\rangle$ is equivalent to obtaining outcome $a$ by measuring the first $n$ qubits of $(U \otimes I)|\psi\rangle$, since the $X^{s^X_{i,m'}}$ operators only flip the bits at the corresponding positions, depending on the values of $s^X_{i,m'}$, while the $Z^{s^Z_{i,m'}}$ operators have no effect (just introducing a global phase). In other words:

$$(\langle a'| \otimes I)(U' \otimes I)|\psi\rangle = \pm(\langle a| \otimes I)(U \otimes I)|\psi\rangle .$$

This also immediately shows that the post-measurement state of $(U' \otimes I)|\psi\rangle$ with measurement outcome $a'$, is the same as that of $(U \otimes I)|\psi\rangle$ with measurement outcome $a$, up to a global phase. □

**Blindness.** Let us first define the security of the protocol. Intuitively, the following should hold: A malicious server should be unable to distinguish between the possible computations chosen by the client, based on the information it receives during the protocol. However, note that the server does learn the dimensions of the brickwork state $(n, m)$, which provide an upper bound on the size of the client's computation. This information is modeled as a leakage to the server.

To formalize this intuition, recall that any quantum adversary can be modeled as a sequence of unitaries, acting on the message registers along with an internal register containing the adversary's workspace and sufficiently-many ancillas. Thus, when defining blindness we can without loss of generality consider only the state that the adversary holds at the end of the execution. More precisely, for a given input $W = \{\phi_{x,y}\}_{x\in[n],y\in[m-1]}$ (encoded as a sequence of measurement angles), we define $\sigma_{W,a}$ to be the (subnormalized) state held by the prover in the end of the protocol, corresponding to the output of the verifier being $a$, and conditioned on the input of the protocol being $W$. We define information-theoretical blindness in the following.

**Definition 4.2** (Information-Theoretical Blindness)**.** *The HBQC protocol is* information-theoretically blind while leaking at most $L(\cdot)$*, the dimensions of the used brickwork state, for all provers and for all possible inputs $W_0$ and $W_1$ with $L(W_0) = L(W_1)$, we have that*

$$\sum_a \sigma_{W_0,a} = \sum_a \sigma_{W_1,a}.$$

Note that this definition is different from that presented in [BFK09]. Our definition is implied by theirs, and we choose this alternative formulation as it will be more convenient to generalize to the computational settings and ultimately will allow us to connect with applications. Let us now prove the following helping lemma.

**Lemma 4.3.** *For all $\theta \in \mathbb{R}$, we have $|+_\theta\rangle\langle+_\theta| + |+_{\theta+\pi}\rangle\langle+_{\theta+\pi}| = I$.*

*Proof.* Follows by direct calculation:

$$\begin{aligned}
|+_\theta\rangle\langle+_\theta| + |+_{\theta+\pi}\rangle\langle+_{\theta+\pi}| &= \frac{1}{2}\begin{pmatrix} 1 & e^{-i\theta} \\ e^{i\theta} & 1 \end{pmatrix} + \frac{1}{2}\begin{pmatrix} 1 & e^{-i\theta-i\pi} \\ e^{i\theta+i\pi} & 1 \end{pmatrix} \\
&= \frac{1}{2}\begin{pmatrix} 1 & e^{-i\theta} \\ e^{i\theta} & 1 \end{pmatrix} + \frac{1}{2}\begin{pmatrix} 1 & -e^{-i\theta} \\ -e^{i\theta} & 1 \end{pmatrix} \\
&= I.
\end{aligned}$$

□

**Theorem 4.4.** *The HBQC protocol is information-theoretically blind while leaking at most the dimensions of the brickwork state, i.e., the pair* $(n, m)$.

*Proof.* The proof follows the same argument as in [BFK09], up to minor syntactical adjustments. Let $W = \{\phi_{x,y}\}_{x\in[n],y\in[m-1]}$ be an arbitrary input with $L(W) = (n, m)$. Note that throughout the execution of the protocol the server receives $(n, m)$, along with the following information

$$\{\varphi'_{x,y}\}_{x\in[n],y\in[8]}, \left\{\left|+_{\theta_{x,y}}\right\rangle, \varphi'_{x,y} - \theta_{x,y} + r_{x,y}\pi\right\}_{x\in[n],y\in[\![9,m'-1]\!]}.$$

The first tuple can be ignored for the analysis, as this information is something the server can compute on its own, since

$$\varphi'_{x,y} = (-1)^{s^X_{x,y}} \cdot \varphi_{x,y} + s^Z_{x,y} \cdot \pi = s^Z_{x,y} \cdot \pi \qquad \forall x \in [n], y \in [8]$$

and the server knows $s_{x,y} = b_{x,y}$ for all $x \in [n], y \in [8]$, hence also $s^Z_{x,y}$. We are left with

$$\left\{\left|+_{\theta_{x,y}}\right\rangle, \varphi'_{x,y} - \theta_{x,y} + r_{x,y}\pi\right\}_{x\in[n],y\in[\![9,m'-1]\!]} = \{\left|+_{\tau_{x,y}+r_{x,y}\pi}\right\rangle, \varphi'_{x,y} - \tau_{x,y}\}_{x\in[n],y\in[\![9,m'-1]\!]}$$

by defining $\tau_{x,y} := \theta_{x,y} - r_{x,y}\pi$. Now, consider $\tau_{x,y}$ to be sampled uniformly at random from $\Theta$ instead of $\theta_{x,y}$. The distribution remains unchanged.

We now argue that, from the server's perspective, each qubit is independently in the maximally mixed state, and that each angle is independently and uniformly distributed in $\Theta$. To do this, we begin by considering the information from the last layer, i.e.,

$$\{\left|+_{\tau_{x,y}+r_{x,y}\pi}\right\rangle, \varphi'_{x,y} - \tau_{x,y}\}_{x\in[n],y=m'-1}.$$

Note that $r_{x,m'-1}$ only appears in the quantum state $|+_{\tau_{x,m'-1}+r_{x,m'-1}\pi}\rangle$, and for example, not in

$$\varphi'_{i,m'-1} = (-1)^{s^X_{i,m'-1}} \cdot \varphi_{i,m'-1} + s^Z_{i,m'-1} \cdot \pi$$

for $i \in [n]$, since only the measurement outcomes $s_{j,k} = b_{j,k} \oplus r_{j,k}$ from the previous layers appear in the formula. Thus, $r_{x,m'-1}$ for $x \in [n]$ is independent of everything else and hidden from the server, meaning the server receives the maximally mixed state $I/2$ by Lemma 4.3. Therefore, only $\varphi'_{x,m'-1} - \tau_{x,m'-1}$ depends on $\tau_{x,m'-1}$, which is then also uniformly random and independent of everything else. Consequently, the qubits in layer $y = m' - 1$ are maximally mixed, and the corresponding angles are independently uniformly distributed.

We can now inductively move on to the previous layer, say layer $y_i$, and apply the same reasoning, where the key observation is that $r_{x,y_i}$ no longer depends on the angles defined in subsequent layers. To summarize, we have shown that the view of the server consists of the classical messages:

$$\{\tau^*_{x,y} : \tau^*_{x,y} \leftarrow\!\$\ \Theta\}_{x\in[n],y\in[\![9,m'-1]\!]}$$

and all qubits are in the maximally mixed state. We can conclude that the view of the server is perfectly independent of $W$, which proves the desired implication. □

### 4.2 Classical Half-Blind Quantum Computation

Finally, we show how to make the verifier in the above protocol completely classical, at the cost of introducing computational assumptions. We refer to this task as *classical half-blind quantum computation* (CHBQC). The protocol is identical to the one presented in Section 4.1, except for the following two modifications:

- We replace step 3 in the State Preparation phase with any blind RSP protocol satisfying the properties in Section 3. For all $(x, y)$, denote by $(t_{x,y}, \theta_{x,y})$ the output of the verifier.
- In step 2 of the Computation phase (for $y \geq 9$), we instead define

$$\delta_{x,y} := \varphi'_{x,y} - (\theta_{x,y} + t_{x,y}\pi) + r_{x,y}\pi.$$

We remark that, given that we introduced the security parameter in the protocol, all inputs will also implicitly depend on $\lambda$ as well, although we omit this dependency when clear from the context. Next, we show that the protocol is still correct.

**Theorem 4.5.** *The CHBQC protocol as described above is correct, i.e., if both parties honestly follow the protocol, the output will be correct.*

*Proof.* This follows directly from the correctness of both the HBQC protocol and the blind RSP protocol. Note that the RSP protocol prepares states in

$$Z^{t_{x,y}} \left|+_{\theta_{x,y}}\right\rangle = \left|+_{\theta_{x,y}+t_{x,y}\pi}\right\rangle.$$

Now, let $\theta^*_{x,y} := \theta_{x,y} + t_{x,y}\pi$ be the regular angle used in the HBQC protocol, which only appears in the quantum state and the measurement angle $\delta_{x,y}$. The $\delta_{x,y}$ is in the CHBQC protocol also appropriately modified and so correctness follows directly from the correctness of the HBQC protocol. □

Before proving blindness against QPT attackers, we present a formal definition of computational blindness. Analogously as for the information-theoretic version of the definition, for a given family of inputs $W = \{W_\lambda\}_{\lambda\in\mathbb{N}}$, we denote by $\sigma^\lambda_{W,a}$ be the (subnormalized) state of the prover in the end of the protocol run with security parameter $\lambda$, corresponding to the output of the verifier being $a_\lambda$, and conditioned on the input of the protocol being $W_\lambda$.

**Definition 4.6** (Computational Blindness)**.** *The CHBQC protocol is computationally blind while leaking at most $L(\cdot)$, if for all families of inputs $W_0 = \{W_{\lambda,0}\}_{\lambda\in\mathbb{N}}$ and $W_1 = \{W_{\lambda,1}\}_{\lambda\in\mathbb{N}}$ such that $L(W_{\lambda,0}) = L(W_{\lambda,1})$ and any family of QPT-implementable POVMs $\{M_\lambda, I - M_\lambda\}_{\lambda\in\mathbb{N}}$, there exists a negligible function* $\mathsf{negl}$ *such that for all $\lambda \in \mathbb{N}$ it holds that:*

$$\left|\sum_{a_\lambda} \mathrm{tr}(\sigma^\lambda_{W_0,a} M_\lambda) - \sum_{a_\lambda} \mathrm{tr}(\sigma^\lambda_{W_1,a} M_\lambda)\right| \leq \mathsf{negl}(\lambda).$$

**Theorem 4.7.** *The CHBQC protocol is computationally blind while leaking at most the dimensions of the brickwork state, i.e., the pair $(n, m)$.*

*Proof.* The proof follows the same structure as that of Theorem 4.4. Let $W = \{\phi_{x,y}\}_{x\in[n],y\in[m-1]}$ be an arbitrary input with $L(W) = (n, m)$. The view of the distinguisher consists of the transcript of the RSP protocol, along with the classical variables

$$\{\delta_{x,y} := \varphi'_{x,y}\}_{x\in[n],y\in[8]} \text{ and } \left\{\delta_{x,y} := \varphi'_{x,y} - (\theta_{x,y} + t_{x,y}\pi) + r_{x,y}\pi\right\}_{x\in[n],y\in[\![9,m'-1]\!]}$$

where the first tuple does not depend on $W$ and it only depends on public information that the server has, and therefore it can be ignored. We proceed via a hybrid argument where, starting from the last layer, we substitute each $\delta_{x,y}$ with a uniformly sampled $\delta^*_{x,y} \leftarrow\!\!\$\ \Theta$. To see why each hybrid is computationally indistinguishable from the previous one, it suffices to observe that

$$\begin{aligned}\delta_{x,y} &\equiv \varphi'_{x,y} - (\theta_{x,y} + t_{x,y}\pi) + r_{x,y}\pi \\ &\equiv \varphi'_{x,y} - \theta_{x,y} + r^*_{x,y}\pi \\ &\approx_c \varphi'_{x,y} - \theta^*_{x,y} + r^*_{x,y}\pi \\ &\equiv \delta^*_{x,y}\end{aligned}$$

where $r^*_{x,y} \leftarrow\!\!\$\ \{0,1\}$ and $\theta^*_{x,y} \leftarrow\!\!\$\ \Theta$. The second equivalence follows since $r_{x,y}$ is sampled uniformly and independently of $t_{x,y}$ and thus $r_{x,y} \oplus t_{x,y} \in \{0,1\}$ is uniformly distributed as well. The computational indistinguishability follows by the blindness of the RSP.

Finally, in the last hybrid we can see that the view of the adversary consists of some transcripts of the RSP protocol and a set of randomly sampled $\{\delta^*_{x,y}\}_{x,y}$, and in particular is perfectly independent of $W$. Thus, no computationally bounded distinguisher can tell apart two executions for $W_0$ and $W_1$ such that $L(W_0) = L(W_1)$, concluding our proof. □

# 5 A New Compiler for Nonlocal Games

## 5.1 Nonlocal Games

In the following, we briefly review the definition of nonlocal games and quantum strategies for these games.

**Definition 5.1** (Nonlocal Game)**.** *A* (two-player) nonlocal game *is a tuple*

$$\mathcal{G} = (\mathcal{I}_A, \mathcal{I}_B, \mathcal{O}_A, \mathcal{O}_B, \mu, V),$$

*which describes a game involving two non-communicating players, Alice and Bob, who interact with a referee. The sets $\mathcal{I}_A, \mathcal{I}_B, \mathcal{O}_A$, and $\mathcal{O}_B$ are finite. The elements of $\mathcal{I}_A$ (resp. $\mathcal{I}_B$) are referred to as the* questions for Alice *(resp.* questions for Bob*), while the elements of $\mathcal{O}_A$ (resp. $\mathcal{O}_B$) are called the* answers of Alice *(resp.* answers of Bob*). Moreover,*

$$\mu : \mathcal{I}_A \times \mathcal{I}_B \to [0, 1]$$

*is a probability distribution, and*

$$V : \mathcal{O}_A \times \mathcal{O}_B \times \mathcal{I}_A \times \mathcal{I}_B \to \{0, 1\}$$

*is the* verification function*. In the game, the referee samples a question pair $(x, y) \leftarrow \mu$, sending $x$ to Alice and $y$ to Bob. Alice and Bob then return answers $a \in \mathcal{O}_A$ and $b \in \mathcal{O}_B$, respectively. The referee evaluates $V(a, b, x, y)$ to determine the outcome: The players win if the result is 1 and lose if the result is 0.*

We may also use the notation $V(a,b|x,y)$ instead of $V(a,b,x,y)$ to emphasize that this represents the value of answers $a,b$ given questions $x,y$. All information about the game $\mathcal{G}$ is available to the players before the game starts. This allows them to agree on a strategy in advance. However, once the game begins, the players are not allowed to communicate. We will now define what a quantum strategy is.

**Definition 5.2.** *A* quantum strategy *for a nonlocal game $\mathcal{G}$ consists of the following:*

- *A bipartite state $|\psi\rangle \in \mathcal{H}_A \otimes \mathcal{H}_B$.*
- *For every $x \in \mathcal{I}_A$, a POVM $\{A_{xa}\}_{a\in\mathcal{O}_A}$ acting on $\mathcal{H}_A$ with outcomes $a \in \mathcal{O}_A$.*
- *For every $y \in \mathcal{I}_B$, a POVM $\{B_{yb}\}_{b\in\mathcal{O}_B}$ acting on $\mathcal{H}_B$ with outcomes $b \in \mathcal{O}_B$.*

*In such a quantum strategy, the probability of Alice and Bob answering $a$ and $b$, when receiving $x$ and $y$ is given by $p(a,b|x,y) = \langle\psi| A_{xa} \otimes B_{yb} |\psi\rangle$.*

## 5.2 Our Compiler

We present our compiler for nonlocal games. For convenience, we only consider the special case of two-player games, but our compiler can be adapted in a straightforward manner to $k$-player games, akin to [KLVY23]. We describe our compiler as a general transformation that turns a two-player nonlocal game $\mathcal{G}$ into a different (compiled) game $\mathcal{G}_{\text{comp}}$, which consists of only one verifier and a single prover.

More concretely, let $\mathcal{G} = \{\mathcal{G}_\lambda\}_{\lambda\in\mathbb{N}} = \{\mathcal{I}_{\lambda,A}, \mathcal{I}_{\lambda,B}, \mathcal{O}_{\lambda,A}, \mathcal{O}_{\lambda,B}, \mu_\lambda, V_\lambda\}_{\lambda\in\mathbb{N}}$ be a family of two-player nonlocal games, and let $U = \{U_\lambda\}_{\lambda\in\mathbb{N}} = \{U_{\lambda,x}\}_{\lambda\in\mathbb{N}, x\in\mathcal{I}_{\lambda,A}}$, where $U_{\lambda,x}$ are the unitaries corresponding to an optimal strategy of Alice for $\mathcal{G}_\lambda$. We assume without loss of generality that, for any given $\lambda$, the measurement patterns for all $U_{\lambda,x}$ are of the same size, which can be achieved by padding with identities to the size of the largest one.

On input of the security parameter $(1^\lambda)$ the prover and the verifier engage in the following interactive protocol:

- The verifier samples a question pair $(x,y) \leftarrow \mu_\lambda$.
- The verifier and the prover engage in the CHBQC protocol (Section 4.2) with the verifier's input being $U_{\lambda,x}$ and the prover's state $|\psi\rangle$ being arbitrary. Let $a'$ be the output of the prover and let $a$ be the output of the verifier derived from $a'$.
- The verifier sends $y$ to the prover in plain.
- The prover replies with some $b$.
- The verifier accepts if $a \in \mathcal{O}_{\lambda,A}$ and $b \in \mathcal{O}_{\lambda,B}$, and if $V_\lambda(a,b|x,y) = 1$.

The completeness of the protocol is immediate, i.e., if $\mathcal{G}_\lambda$ admits a strategy that succeeds with probability $\omega_\lambda$, then simply running such a strategy sequentially wins $(\mathcal{G}_\lambda)_{\text{comp}}$ with probability negligibly close to $\omega_\lambda$.

**Definition 5.3.** *A* QPT strategy *for a family of compiled games $\mathcal{G} = \{\mathcal{G}_\lambda\}_\lambda$ is a QPT algorithm $\{W_\lambda\}_\lambda$. The quantum prover behaves as follows: When receiving the question $y \in \mathcal{I}_{\lambda,B}$, the prover applies $W_\lambda$ to $|y\rangle$ along with the post-measurement state of the CHBQC protocol. The prover measures a suitable number of qubits and respond with the measurement outcome $b$.*

The prover's behavior can be described by POVMs $\{B^\lambda_{yb}\}_{b\in\mathcal{O}_{\lambda,B}}$ where

$$B^\lambda_{yb} = (\langle b| \otimes I)W^\dagger_\lambda(|y\rangle\langle y| \otimes I)W_\lambda(|b\rangle \otimes I).$$

The reason why we define QPT strategies in terms of algorithms instead of POVMs is that the QPT assumptions are easier to state in the former way.

**Soundness Analysis.** Let $\sigma^\lambda_{x,a}$ be the (subnormalized) state of the prover after the execution of the CHBQC protocol on security parameter $\lambda$, corresponding to the output of the verifier being $a \in \mathcal{O}_{\lambda,A}$, and conditioned on the input of the protocol being $x \in \mathcal{I}_{\lambda,A}$. By the computational blindness of the CHBQC protocol, we can immediately deduce the following.

**Lemma 5.4.** *For all $x, x' \in \mathcal{I}_{\lambda,A}$ and any family of QPT-implementable POVMs $\{M_\lambda, I - M_\lambda\}_{\lambda\in\mathbb{N}}$, there exists a negligible function* negl *such that for all $\lambda \in \mathbb{N}$ it holds that:*

$$\left|\sum_{a\in\mathcal{O}_{\lambda,A}} \mathrm{tr}(\sigma^\lambda_{x,a}M_\lambda) - \sum_{a\in\mathcal{O}_{\lambda,A}} \mathrm{tr}(\sigma^\lambda_{x',a}M_\lambda)\right| \leq \mathsf{negl}(\lambda). \tag{1}$$

For the case of constant games (i.e., $\mathcal{G}_\lambda = \mathcal{G}$ for some fixed nonlocal game $\mathcal{G}$), soundness of the compiler can be proven using the same analysis as in [KMP+24]. The only step that has to be slightly generalized is that [NZ23, Lemma 8] has to be proven for more general states

$$\sigma^\lambda_x := \sum_{a\in\mathcal{O}_{\lambda,A}} \sigma^\lambda_{x,a},$$

instead of states of the form

$$\rho^\lambda_x := \mathop{\mathbb{E}}_{c_1,\ldots,c_m=\mathrm{Enc}(x_\lambda)} \sum_{\alpha_1,\ldots,\alpha_m} (A^{c_1}_{\lambda,\alpha_1}) \otimes \cdots \otimes (A^{c_m}_{\lambda,\alpha_m})(|\psi_\lambda\rangle\langle\psi_\lambda|)^{\otimes m}(A^{c_1}_{\lambda,\alpha_1})^\dagger \otimes \cdots \otimes (A^{c_m}_{\lambda,\alpha_m})^\dagger,$$

where $A$ denotes Alice's POVM in the KLVY compiler (we refer to [KMP+24] for precise definitions of the operators). We prove this generalization in the following.

**Lemma 5.5** ([NZ23, Lemma 8])**.** *Let $\lambda \in \mathbb{N}$ be a security parameter. For any two efficiently sampleable distributions $\{D_{\lambda,1}\}, \{D_{\lambda,2}\}$ over plaintext Alice questions, for any efficiently preparable state $\sigma^\lambda_x$ (where $\sigma^\lambda_x$ arises from this new compiler), and for any two-outcome measurement $\{M_\lambda, I - M_\lambda\}$ that can be implemented by a circuit with size* $\mathsf{poly}(\lambda)$ *acting on $m = \mathsf{poly}(\lambda)$ copies of $\sigma^\lambda_x$, there exists a negligible function* $\mathsf{negl}(\lambda)$ *such that, for all $\lambda \in \mathbb{N}$ it holds that*

$$\left|\mathop{\mathbb{E}}_{x\leftarrow D_{\lambda,1}} \mathrm{tr}((\sigma^\lambda_x)^{\otimes m}M_\lambda) - \mathop{\mathbb{E}}_{x\leftarrow D_{\lambda,2}} \mathrm{tr}((\sigma^\lambda_x)^{\otimes m}M_\lambda)\right| \leq \mathsf{negl}(\lambda). \tag{2}$$

This statement can be reduced to Lemma 5.4 by a simple hybrid argument.

*Proof.* Let $\{M_\lambda, I - M_\lambda\}$ be a two-outcome measurement that can be implemented by a circuit with size $\mathsf{poly}(\lambda)$ acting on $m$ copies of $\sigma_x^\lambda$ such that Eq. (2) does not hold, i.e.

$$m_\lambda := \left| \mathop{\mathbb{E}}_{x \leftarrow D_{\lambda,1}} \mathrm{tr}((\sigma_x^\lambda)^{\otimes m} M_\lambda) - \mathop{\mathbb{E}}_{x \leftarrow D_{\lambda,2}} \mathrm{tr}((\sigma_x^\lambda)^{\otimes m} M_\lambda) \right| > \mathsf{negl}(\lambda).$$

Then we can construct a two-outcome measurement $\{N_\lambda, I - N_\lambda\}$ that can be implemented by a circuit with size $\mathsf{poly}'(\lambda)$ acting on $\sigma_x^\lambda$ such that Eq. (1) does not hold as follows. Given input $\sigma_x^\lambda$ with $x \leftarrow D_{\lambda,1}$ or $x \leftarrow D_{\lambda,2}$, choose an index $i \in \{1, \ldots, \mathsf{poly}(\lambda)\}$ uniformly random, prepare the state $(\sigma_{x_1}^\lambda)^{\otimes i-1} \otimes (\sigma_x^\lambda) \otimes (\sigma_{x_2}^\lambda)^{\otimes \mathsf{poly}(\lambda)-i}$ where $x_1 \leftarrow D_{\lambda,1}$ and $x_2 \leftarrow D_{\lambda,2}$, and apply $M_\lambda$ to this prepared state. Then, we have

$$\begin{aligned}
&\left| \mathop{\mathbb{E}}_{x \leftarrow D_{\lambda,1}} \mathrm{tr}(\sigma_x^\lambda N_\lambda) - \mathop{\mathbb{E}}_{x \leftarrow D_{\lambda,2}} \mathrm{tr}(\sigma_x^\lambda N_\lambda) \right| \\
&= \frac{1}{\mathsf{poly}(\lambda)} \Bigg| \sum_{i=1}^{\mathsf{poly}(\lambda)} \mathop{\mathbb{E}}_{x_1 \leftarrow D_{\lambda,1}} \mathop{\mathbb{E}}_{x_2 \leftarrow D_{\lambda,2}} \mathrm{tr}\Big( (\sigma_{x_1}^\lambda)^{\otimes i} \otimes (\sigma_{x_2}^\lambda)^{\otimes \mathsf{poly}(\lambda)-i} M_\lambda \Big) \\
&\qquad\qquad - \mathop{\mathbb{E}}_{x_1 \leftarrow D_{\lambda,1}} \mathop{\mathbb{E}}_{x_2 \leftarrow D_{\lambda,2}} \mathrm{tr}\Big( (\sigma_{x_1}^\lambda)^{\otimes i-1} \otimes (\sigma_{x_2}^\lambda)^{\otimes \mathsf{poly}(\lambda)-i+1} M_\lambda \Big) \Bigg| \\
&= \frac{1}{\mathsf{poly}(\lambda)} \left| \mathop{\mathbb{E}}_{x \leftarrow D_{\lambda,1}} \mathrm{tr}\Big( (\sigma_x^\lambda)^{\otimes \mathsf{poly}(\lambda)} M_\lambda \Big) - \mathop{\mathbb{E}}_{x \leftarrow D_{\lambda,2}} \mathrm{tr}\Big( (\sigma_x^\lambda)^{\otimes \mathsf{poly}(\lambda)} M_\lambda \Big) \right| \\
&\geq \frac{1}{\mathsf{poly}^*(\lambda)}.
\end{aligned}$$

This contradicts Lemma 5.4 for $x \leftarrow D_{\lambda,1}, x' \leftarrow D_{\lambda,2}$. □

Once we have established this fact, the proofs of [NZ23, Lemma 15-17] (see also [CMM+24, Lemma 2.21]) follows identically. This in turn is the only result in the proof of soundness [KMP+24], where IND-CPA security of the QFHE scheme is used. By proving [NZ23, Lemma 8] for this compiler, the following proposition, and consequently the soundness of this proposed compiler for *constant* games, follows as an immediate corollary.

**Proposition 5.6** ([KMP+24, Proposition 4.6])**.** *Consider any nonlocal game $\mathcal{G}$ and a QPT strategy for the compiled game $\mathcal{G}_{comp}$ (which is the same for all $\lambda$). Let $x, x' \in \mathcal{I}_A$, and let $P = P(\{B_{yb}\})$ be a polynomial in noncommuting variables $\{B_{yb}\}_{y \in \mathcal{I}_B, b \in \mathcal{O}_B}$. Then there exists a negligible function $\eta$ such that, for all $\lambda \in \mathbb{N}$,*

$$\left| \mathrm{tr}\left( \sigma_x^\lambda \, P(\{B_{yb}^\lambda\}) \right) - \mathrm{tr}\left( \sigma_{x'}^\lambda \, P(\{B_{yb}^\lambda\}) \right) \right| \leq \eta(\lambda),$$

*and where $\{B_{yb}^\lambda\}_{b \in \mathcal{O}_B}$ are POVMs for $y \in \mathcal{I}_B$, corresponding to the measurements that lead to the prover's second reply.*

### 5.3 Classical Verification of Quantum Computation

In [NZ23], the KLVY compiler is used to design a fully classical protocol to verify any BQP computation. They build a protocol for the verification of the $(XX, ZZ)$-local Hamiltonian problem

(with only $X$ and $Z$ measurements), that is well-known to be QMA-complete. The protocol consists of a nonlocal game, that in turn is a combination of the CHSH game and the commutation game, compiled via the KLVY transformation into a single-prover protocol. Differently from the settings discussed above, the nonlocal game is no longer constant, and it is instead a sequence of games indexed by the security parameter, thus requiring a different analysis from what discussed above. Fortunately, the only point in [NZ23] where the KLVY compiler is invoked is in the proof of Lemma 5.5. Thus, our protocol and analysis can be directly plugged in as a replacement of the KLVY compiler, and the remainder of the analysis follows in verbatim from [NZ23].

**The [NZ23] Verification Protocol.** We recall the verification protocol as described in [NZ23]. Let $\{H_\lambda\}_\lambda$ be a family of Hamiltonians with

$$H_\lambda := \sum_{W,i,j} p_{\lambda,W,i,j} W(e_i + e_j)$$

where $W \in \{X, Z\}$, $e_i \in \{0,1\}^\lambda$ is the $i$-th unit, and $\sum_{W,i,j} p_{\lambda,W,i,j} = 1$ form a probability distribution, with $H_\lambda$ acting on $\lambda$ qubits. Let $D_X^\lambda$ be the distribution specified by $H_\lambda$, conditioned on $W = X$, and $D_Z^\lambda$ be the distribution conditioned on $W = Z$.

The following nonlocal game $\mathcal{G} = \{\mathcal{G}_\lambda\}$ allows one to certify whether the smallest eigenvalue of $H_\lambda$ is smaller or equal than $\alpha_\lambda$ or greater equal than $\beta_\lambda$ where $\beta_\lambda - \alpha_\lambda = 1/\mathsf{poly}(\lambda)$. Let $\kappa_\lambda = \Theta((\beta_\lambda - \alpha_\lambda)^2)$.

- The verifier samples the questions $q_A$ and $q_B$ as follows (padding all $q_A$ so that they have the same length).
  - (CHSH) With probability $(1 - \kappa_\lambda)/2$, sample $a \leftarrow\$ \{0,1\}^\lambda$ uniformly and $b \leftarrow\$ D_X^\lambda$, conditioned on $a \cdot b = 1$. Sample also $x, y \leftarrow\$ \{0,1\}$. Set $q_A := (\text{CHSH}, (a, b, x))$ and $q_B := y$.
  - (Commutation) With probability $(1-\kappa_\lambda)/2$, sample $a \leftarrow\$ \{0,1\}^\lambda$ uniformly and $b \leftarrow\$ D_X^\lambda$, conditioned on $a \cdot b = 0$. Sample also $y \leftarrow\$ \{0,1\}$. Set $q_A := (\text{Commute}, (a, b))$ and $q_B := y$.
  - (Teleport) With probability $\kappa_\lambda$, sample $y \leftarrow\$ \{0,1\}$. Set $q_A := \text{Teleport}$ and $q_B := y$.
- Send $q_A$ to Alice and $q_B$ to Bob, and receive $s_A$ and $s_B$, respectively.
- The verifier accepts if the following conditions are satisfied (depending on the subprotocol that was selected in the previous round).
  - (CHSH) Let $s_A \in \{0,1\}$ and $s_B \in \{0,1\}^\lambda$. Accept if:

$$s_A + (1 - y)(a \cdot s_B) + y(b \cdot s_B) = x \cdot y.$$

  - (Commutation) Let $s_A \in \{0,1\}^2$ and $s_B \in \{0,1\}^\lambda$. Accept if:

$$(1 - y)(a \cdot s_B) + y(b \cdot s_B) = s_{A,y}.$$

– (Teleport) Let $s_A \in \{0,1\}^{2\lambda}$ and $s_B \in \{0,1\}^{\lambda}$. Sample

$$w := \begin{cases} 0 & \text{w.p. } \sum_{i,j} p_{\lambda,X,i,j} \\ 1 & \text{w.p. } \sum_{i,j} p_{\lambda,Z,i,j} \end{cases}.$$

If $w \neq q_B$ accept, else sample a term $W(e_i + e_j) \leftarrow\!\$\ D^{\lambda}_W$ where $W = X$ if $w = 0$ and $W = Z$ otherwise. Then:

* If $W = X$ compute outcome $(-1)^{s_{B,i}+s_{B,j}+s_{A,i}+s_{A,j}}$ and accept if this is $-1$.
* If $W = Z$ compute outcome $(-1)^{s_{B,i}+s_{B,j}+s_{A,\lambda+i}+s_{A,\lambda+j}}$ and accept if this is $-1$.

**Analysis.** We analyze the soundness of the protocol, when compiled through the procedure described in Section 5.2.

If we fix $a$ and $b$ the CHSH and commutation subtest reduce to the CHSH and commutation game, respectively. In the commutation game, Alice receives an empty question and answers with some $a \in \{0,1\}^2$. Bob receives a question $y \in \{0,1\}$ and answers with $b \in \{0,1\}$. The players win the game if $b = a_y$, i.e. the answer of Bob coincides with the $y$th bit in Alice's answer. This game has the property that in any perfect quantum strategy of the game, the observables of Bob commute, hence the name.

In the CHSH game the verifier prepares question $(x, y) \leftarrow\!\$\ \{0,1\}^2$ and sends $x$ to Alice and $y$ to Bob. Alice answers with $a \in \{0,1\}$ and Bob with $b \in \{0,1\}$. The players win the game if $x \cdot y = a \oplus b$. The optimal winning probability of a quantum strategy is $\omega_{\text{CHSH}} = \frac{1}{2} + \frac{1}{2\sqrt{2}}$. In the compiled game the winning probability of a QPT strategy can be bounded by $\omega_{\text{CHSH}} + \mathsf{negl}(\lambda)$ for some negligible function $\mathsf{negl}$ (cf. [NZ23, Lemma 24]). Since the proof only uses Proposition 5.6 (which follows from Lemma 5.5) and Jensen's inequality, the same result holds for our compiler.

If the winning probability of the employed strategy is $\varepsilon$-close to $\omega_{\text{CHSH}}$, then the Alice and Bob operators approximately anti-commute. This is also the case for the corresponding compiled game. To prove this we follow [NZ23] closely. Note however that whenever vector norm inequalities are used in the proof, we cannot directly transfer this to our setting, but instead we can leverage the Frobenius norm defined as $||O||_F = \sqrt{\text{tr}(O^\dagger O)}$. To see this, consider the expression $\sum_\alpha ||O|\psi_{c\alpha}\rangle||^2$, where $\psi_{c\alpha} = A^c_\alpha |\psi\rangle$. It holds

$$\begin{aligned} \sum_\alpha ||O|\psi_{c\alpha}\rangle||^2 &= \sum_\alpha \langle\psi_{c\alpha}| O^\dagger O |\psi_{c\alpha}\rangle \\ &= \sum_\alpha \text{tr}\Big(|\psi_{c\alpha}\rangle\langle\psi_{c\alpha}| O^\dagger O\Big) \\ &= \text{tr}\left(\underbrace{\sum_\alpha |\psi_{c\alpha}\rangle\langle\psi_{c\alpha}|}_{=:\rho} O^\dagger O\right) \\ &= ||O\sqrt{\rho}||_F^2. \end{aligned}$$

Recall that in a QPT strategy for the compiled CHSH game, the prover has POVM's $\{B^\lambda_{yb}\}_{b\in\{0,1\}}$ for $y \in \{0,1\}$. We define observables $B^\lambda_y := B^\lambda_{y0} - B^\lambda_{y1}$ for $y \in \{0,1\}$. Furthermore, let $\{B_0, B_1\} := B_0B_1 + B_1B_0$ denote the anticommutator and let $[B_0, B_1] := B_0B_1 - B_1B_0$ denote the commutator of $B_0$ and $B_1$.

**Lemma 5.7** ([NZ23, Lemma 34])**.** *For any strategy that succeeds in the compiled CHSH game with probability $\omega_{\text{CHSH}} - \varepsilon(\lambda)$, there exists a negligible function $\mathsf{negl}$, such that for all $\lambda$ it holds that*

$$\operatorname{tr}\Big(\sigma_0^\lambda |\{B_0^\lambda, B_1^\lambda\}|^2\Big) \leq \mathcal{O}(\varepsilon(\lambda) + \mathsf{negl}(\lambda))$$

*Proof.* We define the distribution $\mu^\lambda$ similarly to [NZ23]. First, let $a = (-1)^{a'}$ where $a'$ is the outcome of Alice's computation for plaintext 0, i.e. in our compiler the output of the verifier derived from the output of the prover after the CHBQC protocol with Alice plaintext question 0. Then, measure the observable $(B_0^\lambda + B_1^\lambda)/\sqrt{2}$ on the post-measurement state of the CHBQC protocol to obtain an outcome $b$. From the SOS certificate for the CHSH game given in [NZ23], we get

$$\mathop{\mathbb{E}}_{\mu^\lambda}[((a-b)^2] \leq \mathcal{O}(\varepsilon(\lambda) + \mathsf{negl}(\lambda)).$$

After rewriting the expectation we see that

$$||((B_0^\lambda + B_1^\lambda) - \sqrt{2}I)\sqrt{\sigma_{0,0}^\lambda}||_F^2 + ||((B_0^\lambda + B_1^\lambda) + \sqrt{2}I)\sqrt{\sigma_{0,1}^\lambda}||_F^2 \leq \mathcal{O}(\varepsilon(\lambda) + \mathsf{negl}(\lambda)).$$

Using this and the fact that $(B_0^\lambda + B_1^\lambda)^2 = 2I + \{B_0^\lambda, B_1^\lambda\}$ we get that

$$\begin{aligned}
\operatorname{tr}\Big(\sigma_0^\lambda |\{B_0^\lambda, B_1^\lambda\}|^2\Big) &= \operatorname{tr}\Big(\sigma_{0,0}^\lambda |\{B_0^\lambda, B_1^\lambda\}|^2\Big) + \operatorname{tr}\Big(\sigma_{0,1}^\lambda |\{B_0^\lambda, B_1^\lambda\}|^2\Big) \\
&= \operatorname{tr}\Big(\sigma_{0,0}^\lambda ((B_0^\lambda + B_1^\lambda)^2 - 2I)^2\Big) + \operatorname{tr}\Big(\sigma_{0,1}^\lambda ((B_0^\lambda + B_1^\lambda)^2 - 2I)^2\Big) \\
&= ||((B_0^\lambda + B_1^\lambda)^2 - 2I)\sqrt{\sigma_{0,0}^\lambda}||_F^2 + ||((B_0^\lambda + B_1^\lambda)^2 - 2I)\sqrt{\sigma_{0,1}^\lambda}||_F^2 \\
&= ||((B_0^\lambda + B_1^\lambda) + \sqrt{2}I)((B_0^\lambda + B_1^\lambda) - \sqrt{2}I)\sqrt{\sigma_{0,0}^\lambda}||_F^2 \\
&\quad + ||((B_0^\lambda + B_1^\lambda) - \sqrt{2}I)((B_0^\lambda + B_1^\lambda) + \sqrt{2}I)\sqrt{\sigma_{0,1}^\lambda}||_F^2 \\
&\leq (2+\sqrt{2})^2 ||((B_0^\lambda + B_1^\lambda) - \sqrt{2}I)\sqrt{\sigma_{0,0}^\lambda}||_F^2 \\
&\quad + (2+\sqrt{2})^2 ||((B_0^\lambda + B_1^\lambda) + \sqrt{2}I)\sqrt{\sigma_{0,1}^\lambda}||_F^2 \\
&\leq \mathcal{O}(\varepsilon(\lambda) + \mathsf{negl}(\lambda))
\end{aligned}$$

□

In the nonlocal game of the verification protocol, Bob receives questions $y \in \{0,1\}$ and answers with $s \in \{0,1\}^\lambda$. Therefore, we have Bob POVM's $\{B_{ys}^\lambda\}_{s\in\{0,1\}^\lambda}$ for $y \in \{0,1\}$ in each QPT strategy for the compiled game. We define observables

$$\hat{Z}^\lambda(a) := \sum_{s\in\{0,1\}^\lambda} (-1)^{a\cdot s} B_{0s}^\lambda, \qquad \hat{X}^\lambda(b) := \sum_{t\in\{0,1\}^\lambda} (-1)^{b\cdot t} B_{1t}^\lambda.$$

The following results about the subtests in the protocol follow verbatim.

**Lemma 5.8** ([NZ23, Lemma 36])**.** *Suppose the prover's strategy succeeds in the CHSH subtest with probability at least $\omega_{\text{CHSH}} - \varepsilon(\lambda)$. Then, there exists a negligible function $\mathsf{negl}(\lambda)$, such that for all $\lambda$,*

$$\mathop{\mathbb{E}}_{\substack{(a,b)\leftarrow\$\{0,1\}^\lambda\times D_X^\lambda;\\ a\cdot b=1}} \operatorname{tr}\Big(\sigma_{(\text{CHSH},(a,b,0))}^\lambda |\{\hat{Z}^\lambda(a), \hat{X}^\lambda(b)\}|^2\Big) \leq \mathcal{O}(\varepsilon(\lambda) + \mathsf{negl}(\lambda))$$

**Lemma 5.9** ([NZ23, Lemma 37]). *Suppose the prover's strategy succeeds in the commutation subtest with probability at least $1-\varepsilon(\lambda)$. Then,*

$$\mathop{\mathbb{E}}_{\substack{(a,b)\leftarrow\$\{0,1\}^\lambda\times D_X^\lambda;\\ a\cdot b=0}} \operatorname{tr}\left(\sigma^\lambda_{(\text{Commute},(a,b))}|[\hat{Z}^\lambda(a),\hat{X}^\lambda(b)]|^2\right)\leq\mathcal{O}(\varepsilon(\lambda))$$

**Lemma 5.10** ([NZ23, Lemma 38]). *Suppose the prover's strategy succeeds in the CHSH subtest with probability at least $\omega_{\text{CHSH}}-\varepsilon(\lambda)$ and in the commutation subtest with probability at least $1-\varepsilon(\lambda)$. Then, there exists a negligible function $\mathsf{negl}(\lambda)$, such that for all $\lambda$,*

$$\mathop{\mathbb{E}}_{(a,b)\leftarrow\$\{0,1\}^\lambda\times D_X^\lambda} \operatorname{tr}\left(\sigma^\lambda_{\text{Teleport}}|(-1)^a\hat{Z}^\lambda(a)\hat{X}^\lambda(b)-\hat{X}^\lambda(b)\hat{Z}^\lambda(a)|^2\right)\leq\mathcal{O}(\varepsilon(\lambda)+\mathsf{negl}(\lambda))$$

Having the results regarding the subtests in place, we can define the isometry analogously.

**Lemma 5.11** ([NZ23, Lemma 39]). *For any $u_1,u_2\in\{0,1\}$, there exists a negligible function $\mathsf{negl}$, such that for all $\lambda$ it holds that*

$$\mathop{\mathbb{E}}_{(a,b)\leftarrow\$\{0,1\}^\lambda\times D_X^\lambda}\sum_{\substack{a;\\ a_i=u_1\\ a_j=u_2}}\operatorname{tr}\left(\sigma^\lambda_{\text{Teleport},a}|(-1)^{a\cdot b}\hat{Z}^\lambda(a)\hat{X}^\lambda(b)\hat{Z}^\lambda(a)-\hat{X}^\lambda(b)|\right)\leq\mathcal{O}\left(\sqrt{\varepsilon(\lambda)+\mathsf{negl}(\lambda)}\right)$$

Let $\mathcal{H}_Q$ and $\mathcal{H}_A$ be two copies of $(\mathbb{C}^2)^{\otimes\lambda}$. The $\lambda$-qubit SWAP isometry $V:\mathcal{H}_{\text{Prover}}\to\mathcal{H}_{\text{Prover}}\otimes\mathcal{H}_Q\otimes\mathcal{H}_A$ is defined by:

$$V|\phi\rangle=\left(\frac{1}{2^\lambda}\sum_{u,v\in\{0,1\}^\lambda}\hat{Z}^\lambda(u)\hat{X}^\lambda(v)\otimes I\otimes Z(u)X(v)\right)|\phi\rangle\otimes\left|\phi^+\right\rangle^{\otimes\lambda}.$$

Furthermore, let $H_X^\lambda$ and $H_Z^\lambda$ denote $H_\lambda$ restricted to the $XX$ and $ZZ$ terms, respectively. We use the notation $\hat{\mathbb{E}}[H_X^\lambda]$ and $\hat{\mathbb{E}}[H_Z^\lambda]$ for the expected value of the outcome computed by the verifier in a teleport round, conditioned on $w=q_B$ and the verifier choosing an $XX$ term or $ZZ$ term, respectively.

**Lemma 5.12** ([NZ23, Lemma 43]). *Define $\rho_s:=\operatorname{tr}_{\text{Prover},A}[V\sigma^\lambda_{\text{Teleport},s}V^\dagger]$. Then, assuming that the prover passes with probability $\omega_{\text{CHSH}}-\varepsilon(\lambda)$ in the CHSH subtest and with probability $1-\varepsilon(\lambda)$ in the commutation subtest, there exists a negligible function $\mathsf{negl}$, such that for all $\lambda$ it holds that*

$$\left|\sum_{u_1,u_2}(-1)^{u_1+u_2}\sum_{\substack{s;\\ s_i=u_1\\ s_j=u_2}}\mathop{\mathbb{E}}_{b\leftarrow\$D_X^\lambda}\operatorname{tr}(X(b)\rho_s)-\hat{\mathbb{E}}[H_X^\lambda]\right|\leq\mathcal{O}\left(\sqrt{\varepsilon(\lambda)+\mathsf{negl}(\lambda)}\right)$$

**Lemma 5.13** ([NZ23, Lemma 44]). *Define $\rho_s:=\operatorname{tr}_{\text{Prover},A}[V\sigma^\lambda_{\text{Teleport},s}V^\dagger]$. Then*

$$\sum_{v_1,v_2}(-1)^{v_1+v_2}\sum_{\substack{a;\\ a_{n+i}=v_1\\ a_{n+j}=v_2}}\mathop{\mathbb{E}}_{a\leftarrow\$D_Z^\lambda}\operatorname{tr}(Z(a)\rho_s)=\hat{\mathbb{E}}[H_Z^\lambda].$$

**Lemma 5.14** ([NZ23, Lemma 45])**.** *Assuming that the prover passes with probability $\omega_{\mathrm{CHSH}} - \varepsilon(\lambda)$ in the CHSH subtest and with probability $1 - \varepsilon(\lambda)$ in the commutation subtest, there exists a state $\rho$ and a negligible function $\mathsf{negl}$ such that for all $\lambda$*

$$\mathop{\mathbb{E}}_{a \leftarrow \$ D_Z^\lambda} \mathrm{tr}(Z(a)\rho) = \hat{\mathbb{E}}[H_Z^\lambda],$$
$$\left| \mathop{\mathbb{E}}_{b \leftarrow \$ D_X^\lambda} \mathrm{tr}(X(b)\rho) - \hat{\mathbb{E}}[H_X^\lambda] \right| \leq \mathcal{O}\left(\sqrt{\varepsilon(\lambda) + \mathsf{negl}(\lambda)}\right)$$

**Lemma 5.15.** *The winning probability in the Teleport subtest is*

$$\omega_{\lambda,\mathrm{Tel}} = 1 - \frac{1}{4}\left(\sum_{i,j} p_{\lambda,Z,i,j}\, \hat{\mathbb{E}}[H_Z^\lambda] + \sum_{i,j} p_{\lambda,X,i,j}\, \hat{\mathbb{E}}[H_X^\lambda]\right).$$

*Proof.*

$$\omega_{\lambda,\mathrm{Tel}} = \Pr(w \neq q_B) + \Pr(w = q_B) \cdot \Pr(\text{Verifier outputs } -1)$$

By definition of $\hat{\mathbb{E}}[H_W^\lambda]$ for $W \in \{X, Z\}$ we get

$$\begin{aligned}\Pr(\text{Verifier outputs } -1) &= \Pr(w=0)\frac{1 - \hat{\mathbb{E}}[H_Z^\lambda]}{2} + \Pr(w=1)\frac{1 - \hat{\mathbb{E}}[H_X^\lambda]}{2} \\ &= \sum_{i,j} p_{\lambda,Z,i,j}\frac{1 - \hat{\mathbb{E}}[H_Z^\lambda]}{2} + \sum_{i,j} p_{\lambda,X,i,j}\frac{1 - \hat{\mathbb{E}}[H_X^\lambda]}{2}\end{aligned}$$

Since $\Pr(w \neq q_B) = \Pr(w = q_B) = \frac{1}{2}$ we have

$$\begin{aligned}\Pr(\text{Verifier outputs } -1) &= \frac{1}{2} + \frac{1}{2}\left(\sum_{i,j} p_{\lambda,Z,i,j}\frac{1 - \hat{\mathbb{E}}[H_Z^\lambda]}{2} + \sum_{i,j} p_{\lambda,X,i,j}\frac{1 - \hat{\mathbb{E}}[H_X^\lambda]}{2}\right) \\ &= 1 - \frac{1}{4}\left(\sum_{i,j} p_{\lambda,Z,i,j}\, \hat{\mathbb{E}}[H_Z^\lambda] + \sum_{i,j} p_{\lambda,X,i,j}\, \hat{\mathbb{E}}[H_X^\lambda]\right)\end{aligned}$$

□

Overall, we obtain the following new implication.

**Proposition 5.16.** *Let $\{H_\lambda\}_\lambda$ be a family of $(XX - ZZ)$-Hamiltonians and let $\xi_{\lambda,H}$ be the lowest eigenvalue. Assuming the existence of a family of claw-free trapdoor functions, then there exists a classical verifier protocol such that for all QPT provers $P^*$ it holds that:*

$$\Pr\left[V \text{ accepts} \mid \xi_{\lambda,H} \leq \alpha_\lambda\right] - \Pr\left[V \text{ accepts} \mid \xi_{\lambda,H} \geq \beta_\lambda\right] = \mathsf{poly}(\beta_\lambda - \alpha_\lambda).$$

When $\beta_\lambda - \alpha_\lambda \geq 1/\mathsf{poly}(\lambda)$, then we obtain a protocol with inverse-polynomial completeness-soundness gap, which can be amplified by standard sequential repetition. The proof of this fact follows along the same lines as [NZ23], and we reproduce it here only for completeness.

*Proof.* It is shown in [NZ23] that, if the smallest eigenvalue is at most $\alpha_\lambda$, then there exists a prover that passes the protocol with probability at least:

$$\frac{(1-\kappa_\lambda)(1+\omega_{\mathrm{CHSH}})}{2} + \kappa_\lambda\left(1-\frac{\alpha_\lambda}{4}\right).$$

Thus, all is left to be shown is that the exists a non-trivial (at least inverse-polynomial) completeness-soundness gap.

Assume towards contradiction that the lowest eigenvalue is at least $\beta_\lambda$ and there exists an efficient prover that passes the protocol with probability at least:

$$\frac{(1-\kappa_\lambda)(1+\omega_{\mathrm{CHSH}})}{2} + \kappa_\lambda\left(1-\frac{\beta_\lambda}{4}\right) + \nu_\lambda$$

for some $\nu_\lambda$ to be set later.

Let us denote by $\omega_{\lambda,\mathrm{CHSH}}$, $\omega_{\lambda,\mathrm{Com}}$, $\omega_{\lambda,\mathrm{Tel}}$ the probability that the prover passes the CHSH, commutation, or teleportation subtest, respectively. Since we can upperbound the probability of passing the commutation subtest by 1 and the probability of passing the CHSH subtest by $\omega_{\mathrm{CHSH}}$ plus a negligible function (cf. [NZ23, Lemma 24]), we can bound the probability of passing the teleportation subtest to:

$$\omega_{\lambda,\mathrm{Tel}} \geq 1 - \frac{\beta_\lambda}{4} + \frac{\nu_\lambda}{\kappa_\lambda} - \mathsf{negl}(\lambda). \tag{3}$$

Moreover, we can bound:

$$\frac{(1-\kappa_\lambda)(\omega_{\lambda,\mathrm{Com}}+\omega_{\lambda,\mathrm{CHSH}})}{2} + \kappa_\lambda\cdot\omega_{\lambda,\mathrm{Tel}} \geq \frac{(1-\kappa_\lambda)(1+\omega_{\mathrm{CHSH}})}{2} + \kappa_\lambda\left(1-\frac{\beta_\lambda}{4}\right)$$

bounding $\omega_{\lambda,\mathrm{Tel}} \leq 1$ and rearranging the terms we obtain:

$$\omega_{\lambda,\mathrm{Com}} + \omega_{\lambda,\mathrm{CHSH}} \geq 1 + \omega_{\mathrm{CHSH}} - \frac{\kappa_\lambda}{2(1-\kappa_\lambda)}.$$

Thus, we can conclude that:

$$\omega_{\lambda,\mathrm{Com}} \geq 1 - \frac{\kappa_\lambda}{2(1-\kappa_\lambda)} \quad \text{and} \quad \omega_{\lambda,\mathrm{CHSH}} \geq \omega_{\mathrm{CHSH}} - \frac{\kappa_\lambda}{2(1-\kappa_\lambda)}.$$

Let $\varepsilon := \kappa_\lambda/2(1-\kappa_\lambda)$. Recall that

$$H_\lambda = \sum_{W,i,j} p_{\lambda,i,j} W(e_i+e_j) = \sum_{i,j} p_{\lambda,Z,i,j} \mathop{\mathbb{E}}_{a\leftarrow\$ D_Z^\lambda} Z(a) + \sum_{i,j} p_{\lambda,X,i,j} \mathop{\mathbb{E}}_{b\leftarrow\$ D_X^\lambda} X(b).$$

Therefore, we have for all states $\rho$, it holds

$$\mathrm{tr}(H_\lambda\rho) = \sum_{i,j} p_{\lambda,Z,i,j} \mathop{\mathbb{E}}_{a\leftarrow\$ D_Z^\lambda} \mathrm{tr}(Z(a)\rho) + \sum_{i,j} p_{\lambda,X,i,j} \mathop{\mathbb{E}}_{b\leftarrow\$ D_X^\lambda} \mathrm{tr}(X(b)\rho).$$

Using Lemma 5.14, there exists a state $\rho_\lambda$, such that:

$$\left|\mathrm{tr}(H_\lambda\rho_\lambda) - \left(\sum_{i,j} p_{\lambda,Z,i,j}\,\hat{\mathbb{E}}[H_Z^\lambda] + \sum_{i,j} p_{\lambda,X,i,j}\,\hat{\mathbb{E}}[H_X^\lambda]\right)\right| \leq \mathcal{O}\left(\sqrt{\varepsilon(\lambda)+\mathsf{negl}(\lambda)}\right). \tag{4}$$

By Lemma 5.15 and Eqs. (3) and (4) we deduce that:

$$\mathrm{tr}[H_\lambda \rho_\lambda] \leq \beta_\lambda - \frac{4\nu_\lambda}{\kappa_\lambda} + \mathsf{negl}(\lambda) + \mathcal{O}\left(\sqrt{\varepsilon(\lambda) + \mathsf{negl}(\lambda)}\right).$$

Thus, we can derive a contradiction if $\mathcal{O}(\sqrt{\varepsilon(\lambda) + \mathsf{negl}(\lambda)}) < 4\nu_\lambda/\kappa_\lambda - \mathsf{negl}(\lambda)$. Setting $\nu_\lambda = \kappa_\lambda(\beta_\lambda - \alpha_\lambda)/8$, we obtain that:

$$\mathcal{O}\left(\sqrt{\frac{\kappa_\lambda}{1-\kappa_\lambda} + \mathsf{negl}(\lambda)}\right) < \frac{\beta_\lambda - \alpha_\lambda}{2} - \mathsf{negl}(\lambda)$$

which implies that

$$\mathcal{O}\left(\sqrt{\frac{\kappa_\lambda}{1-\kappa_\lambda}}\right) < \frac{\beta_\lambda - \alpha_\lambda}{2}$$

for $\kappa_\lambda \in \Theta((\beta_\lambda - \alpha_\lambda)^2)$, as desired. □


### Acknowledgements

A.K., G.M., S.S., and M.W. are supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2092 CASA – 390781972.

G.M. acknowledges support by the European Research Council through an ERC Starting Grant (Grant agreement No. 101077455, ObfusQation).

M.W. acknowledges support by the European Research Council through an ERC Starting Grant (Grant agreement No. 101040907, SYMOPTIC) and by the BMBF through project Quantum Methods and Benchmarks for Resource Allocation (QuBRA).


## References


[ABCC24] Atul Singh Arora, Kishor Bharti, Alexandru Cojocaru, and Andrea Coladangelo. A computational test of quantum contextuality, and even simpler proofs of quantumness, 2024.

[AC01] Mark Adcock and Richard Cleve. A quantum goldreich-levin theorem with cryptographic applications, 2001.

[AMR22] Navid Alamati, Giulio Malavolta, and Ahmadreza Rahimi. Candidate trapdoor claw-free functions from group actions with applications to quantum protocols. In Eike Kiltz and Vinod Vaikuntanathan, editors, *TCC 2022, Part I*, volume 13747 of *LNCS*, pages 266–293, Chicago, IL, USA, November 7–10, 2022. Springer, Heidelberg, Germany.

[BCM+18] Zvika Brakerski, Paul Christiano, Urmila Mahadev, Umesh V. Vazirani, and Thomas Vidick. A cryptographic test of quantumness and certifiable randomness from a single quantum device. In Mikkel Thorup, editor, *59th FOCS*, pages 320–331, Paris, France, October 7–9, 2018. IEEE Computer Society Press.

[Bel64] John S Bell. On the Einstein Podolsky Rosen paradox. *Physics Physique Fizika*, 1(3):195, 1964.

[BFK09] Anne Broadbent, Joseph Fitzsimons, and Elham Kashefi. Universal blind quantum computation. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science*, pages 517–526, 2009.

[BGKM+23] Zvika Brakerski, Alexandru Gheorghiu, Gregory D. Kahanamoku-Meyer, Eitan Porat, and Thomas Vidick. Simple tests of quantumness also certify qubits. In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part V*, volume 14085 of *LNCS*, pages 162–191, Santa Barbara, CA, USA, August 20–24, 2023. Springer, Heidelberg, Germany.

[BKL+22] James Bartusek, Yael Tauman Kalai, Alex Lombardi, Fermi Ma, Giulio Malavolta, Vinod Vaikuntanathan, Thomas Vidick, and Lisa Yang. Succinct classical verification of quantum computation. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part II*, volume 13508 of *LNCS*, pages 195–211, Santa Barbara, CA, USA, August 15–18, 2022. Springer, Heidelberg, Germany.

[BKVV20] Zvika Brakerski, Venkata Koppula, Umesh Vazirani, and Thomas Vidick. Simpler proofs of quantumness, 2020.

[Bra18] Zvika Brakerski. Quantum FHE (almost) as secure as classical. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part III*, volume 10993 of *LNCS*, pages 67–95, Santa Barbara, CA, USA, August 19–23, 2018. Springer, Heidelberg, Germany.

[BVB+24] Matilde Baroni, Quoc-Huy Vu, Boris Bourdoncle, Eleni Diamanti, Damian Markham, and Ivan Šupić. Quantum bounds for compiled XOR games and $d$-outcome CHSH games. *preprint arXiv:2403.05502*, 2024.

[CGJV19] Andrea Coladangelo, Alex B Grilo, Stacey Jeffery, and Thomas Vidick. Verifier-on-a-leash: new schemes for verifiable delegated quantum computation, with quasilinear resources. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 247–277. Springer, 2019.

[CHSH69] John F Clauser, Michael A Horne, Abner Shimony, and Richard A Holt. Proposed experiment to test local hidden-variable theories. *Physical Review Letters*, 23(15):880, 1969.

[CHTW04] R. Cleve, P. Hoyer, B. Toner, and J. Watrous. Consequences and limits of nonlocal strategies. In *Proceedings. 19th IEEE Annual Conference on Computational Complexity, 2004.*, pages 236–249, 2004.

[CLLZ21] Andrea Coladangelo, Jiahui Liu, Qipeng Liu, and Mark Zhandry. Hidden cosets and applications to unclonable cryptography. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part I*, volume 12825 of *LNCS*, pages 556–584, Virtual Event, August 16–20, 2021. Springer, Heidelberg, Germany.

[CMM+24] David Cui, Giulio Malavolta, Arthur Mehta, Anand Natarajan, Connor Paddock, Simon Schmidt, Michael Walter, and Tina Zhang. A computational tsirelson's theorem for the value of compiled xor games, 2024.

[DK06] Vincent Danos and Elham Kashefi. Determinism in the one-way model. *Phys. Rev. A*, 74:052310, Nov 2006.

[Gri19] Alex B. Grilo. A simple protocol for verifiable delegation of quantum computation in one round. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, volume 132 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 28:1–28:13, Dagstuhl, Germany, 2019. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[GV19] Alexandru Gheorghiu and Thomas Vidick. Computationally-secure and composable remote state preparation. In David Zuckerman, editor, *60th FOCS*, pages 1024–1033, Baltimore, MD, USA, November 9–12, 2019. IEEE Computer Society Press.

[GV24] Aparna Gupte and Vinod Vaikuntanathan. How to construct quantum fhe, generically. In *Advances in Cryptology – CRYPTO 2024: 44th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18–22, 2024, Proceedings, Part III*, page 246–279, Berlin, Heidelberg, 2024. Springer-Verlag.

[JNV+21] Zhengfeng Ji, Anand Natarajan, Thomas Vidick, John Wright, and Henry Yuen. MIP* = RE. *Communications of the ACM*, 64(11):131–138, 2021.

[KCVY22] Gregory D. Kahanamoku-Meyer, Soonwon Choi, Umesh V. Vazirani, and Norman Y. Yao. Classically verifiable quantum advantage from a computational bell test. *Nature Physics*, 18(8):918–924, August 2022.

[KLVY23] Yael Kalai, Alex Lombardi, Vinod Vaikuntanathan, and Lisa Yang. Quantum advantage from any non-local game. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, STOC 2023, page 1617–1628, New York, NY, USA, 2023. Association for Computing Machinery.

[KMP+24] Alexander Kulpe, Giulio Malavolta, Connor Paddock, Simon Schmidt, and Michael Walter. A bound on the quantum value of all compiled nonlocal games, 2024.

[Mah18a] Urmila Mahadev. Classical homomorphic encryption for quantum circuits. In Mikkel Thorup, editor, *59th FOCS*, pages 332–338, Paris, France, October 7–9, 2018. IEEE Computer Society Press.

[Mah18b] Urmila Mahadev. Classical verification of quantum computations. In Mikkel Thorup, editor, *59th FOCS*, pages 259–267, Paris, France, October 7–9, 2018. IEEE Computer Society Press.

[MDF17] Atul Mantri, Tommaso F. Demarie, and Joseph F. Fitzsimons. Universality of quantum computation with cluster states and (x, y)-plane measurements. *Scientific reports*, 7(1):42861, 2017.

[MNZ24] Tony Metger, Anand Natarajan, and Tina Zhang. Succinct arguments for qma from standard assumptions via compiled nonlocal games, 2024.

[MPW24] Arthur Mehta, Connor Paddock, and Lewis Wooltorton. Self-testing in the compiled setting via tilted-CHSH inequalities. *preprint arXiv:2406.04986*, 2024.

[NZ23] Anand Natarajan and Tina Zhang. Bounding the quantum value of compiled nonlocal games: From chsh to bqp verification. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1342–1348, 2023.

[RB01] Robert Raussendorf and Hans J. Briegel. A one-way quantum computer. *Phys. Rev. Lett.*, 86:5188–5191, May 2001.

[RUV13] Ben W Reichardt, Falk Unger, and Umesh Vazirani. Classical command of quantum systems. *Nature*, 496(7446):456–460, 2013.

[Slo17] William Slofstra. The set of quantum correlations is not closed. *Forum of Mathematics, Pi*, 7, 2017.

[Vid20] Thomas Vidick. Course fsmp, fall'20: Interactions with quantum devices. http://users.cms.caltech.edu/~vidick/teaching/fsmp/fsmp.pdf, 2020.