CountCrypt: Quantum Cryptography between QCMA and PP

Eli Goldin[1], Tomoyuki Morimae[2], Saachi Mutreja[3], and Takashi Yamakawa[4,2]

[1]New York University (eli.goldin@nyu.edu)
[2]Yukawa Institute for Theoretical Physics, Kyoto University
(tomoyuki.morimae@yukawa.kyoto-u.ac.jp)
[3]Columbia University (saachi@berkeley.edu)
[4]NTT Social Informatics Laboratories (takashi.yamakawa@ntt.com)

October 6, 2025

Abstract

We construct a unitary oracle relative to which **BQP** = **QCMA** but quantum-computation-classical-communication (QCCC) commitments and QCCC multiparty non-interactive key exchange exist. We also construct a unitary oracle relative to which **BQP** = **QMA**, but quantum lightning (a stronger variant of quantum money) exists. This extends previous work by Kretschmer [Kretschmer, TQC22], which showed that there is a quantum oracle relative to which **BQP** = **QMA** but pseudorandom unitaries exist.

We also show that (poly-round) QCCC key exchange, QCCC commitments, and two-round quantum key distribution can all be used to build one-way puzzles. One-way puzzles are a version of "quantum samplable" one-wayness and are an intermediate primitive between pseudorandom state generators and EFI pairs, the minimal quantum primitive. In particular, one-way puzzles cannot exist if **BQP** = **PP**.

Our results together imply that aside from pseudorandom state generators, there is a large class of quantum cryptographic primitives which can exist even if **BQP** = **QCMA**, but are broken if **BQP** = **PP**. Furthermore, one-way puzzles are a minimal primitive for this class. We denote this class "CountCrypt".

Contents

1 Introduction **3**
- 1.1 Our Results . . . 7
- 1.2 Technical Overview . . . 7

2 Preliminaries **14**
- 2.1 Notation . . . 14
- 2.2 Quantum Information . . . 14
- 2.3 The Compressed Oracle Technique/The Recording Method . . . 15
- 2.4 One Way To Hiding . . . 16
- 2.5 Haar Measure And Its Concentration . . . 18
- 2.6 State Tomography . . . 18
- 2.7 Quantum Complexity . . . 18
- 2.8 Quantum Cryptography Definitions . . . 19
- 2.9 State Verification Unitaries . . . 21
- 2.10 Complexity-Theoretic No-Cloning . . . 22

3 Quantum Presampling Lemmas **22**

4 Separating Unitary Oracle **24**
- 4.1 Definition of Oracle . . . 24
- 4.2 Key Lemma . . . 25

5 Quantum Lightning Exists Relative to $(SG, Mix, PSPACE)$. **31**

6 QCCC NIKE Exists Relative To $(SG, Mix, PSPACE)$ **34**

7 QCCC Commitments Exist Relative To $(SG, Mix, PSPACE)$ **38**

8 $\mathbf{BQP}^{SG,Mix,PSPACE} = \mathbf{QCMA}^{SG,Mix,PSPACE}$ **40**

9 OWPuzz Constructions **44**
- 9.1 OWPuzzs From QCCC KE . . . 44
- 9.2 OWPuzzs From QCCC Commitments . . . 44
- 9.3 2 Round QKD Implies State Puzzles . . . 46
- .1 Proof of Lemma 4.3 . . . 55

A Proof of Lemma 2.24 **57**

1 Introduction

Nearly all cryptographic primitives require some form of computational assumption [IL89]. For example, the existence of one-way functions (OWFs) implies that $\mathbf{P} \neq \mathbf{NP}$. However, some primitives seemingly require stronger computational assumptions than others. For instance, it appears that key exchange (KE) requires stronger computational assumptions than OWFs. In particular, OWFs can be built from KE [BCG90], but there exist strong barriers to building KE from OWFs [IR89]. In general, the relationships between classical cryptographic primitives are well-understood. Notably, nearly all primitives can be used to build OWFs, and so the existence of any sort of cryptography necessitates that $\mathbf{P} \neq \mathbf{NP}$.

Everything changes once quantum computers enter the picture. It is not hard to define quantum versions of classical cryptographic primitives, where we allow either the protocol description and/or protocol inputs/outputs to be quantum. For instance, [JLS18] defines pseudorandom state generators (PRSGs), a quantum analogue of pseudorandom generators where the output is a quantum state, and it is indistinguishable from a Haar random quantum state. There exists a classical oracle relative to which single-copy-secure PRSGs[1] exist [KQST23] or quantumly-computable trapdoor functions exist [KQT25] but $\mathbf{P} = \mathbf{NP}$, or there exists a quantum oracle relative to which pseudorandom unitaries (PRUs) exist but $\mathbf{BQP} = \mathbf{QMA}$ [Kre21]. This gives evidence that some quantum primitives require weaker assumptions even than OWFs.

Since [Kre21, MY22, AQY22], a slew of recent work has begun to map out the relationships between different forms of cryptographic hardness. To quickly summarize recent progress, it seems that most quantum cryptographic primitives fall into one of several buckets:

1. "QuantuMania": quantum cryptographic primitives which are broken if $\mathbf{BQP} = \mathbf{QCMA}$. This class includes three types of primitives. The first type is just quantumly-secure classical primitives, such as post-quantum classical OWFs or post-quantum classical public-key encryption (PKE). The second type is primitives with "genuinely quantum" additional functionalities such as PKE with (publicly verifiable) certified deletion [KNY23] and digital signatures with revocable signing keys [MPY23]. The third type is most non-interactive quantum-computation-classical-communication (QCCC) primitives, where locally quantum computing is possible but all communications are classical, such as PKE or digital signatures with classical keys, ciphertexts, or signatures [CGG24].

2. "CountCrypt": quantum cryptographic primitives which may exist if $\mathbf{BQP} = \mathbf{QCMA}$, but are broken by a $\mathbf{PP}$ oracle. That is, CountCrypt primitives do not exist if $\mathbf{BQP} = \mathbf{PP}$. This category includes primitives such as PRSGs [JLS18, Kre21], (pure output) one-way state generators (OWSGs) [MY22, MY24], and one-way puzzles (OWPuzzs) [KT24, CGG+23].[2] There is evidence that these primitives may exist even if $\mathbf{BQP} = \mathbf{QCMA}$ [Kre21]. On the other hand, post-quantum classical OWFs do not exist if $\mathbf{BQP} = \mathbf{QCMA}$. Thus, it seems plausible that these

[1] A single-copy-secure PRSG is a variant of PRSGs where the adversary receives only a single copy of a pseudorandom state.

[2] OWSGs and OWPuzzs are natural quantum analogues of OWFs. A OWSG is a pair $(\mathsf{Gen}, \mathsf{Ver})$ of quantum polynomial-time (QPT) algorithms. $\mathsf{Gen}(k) \to \phi_k$ takes a random classical bit string k as input and outputs a quantum state ϕ_k. The security requires that no QPT adversary, given many copies of ϕ_k, can output k' such that $\mathsf{Ver}(k', \phi_k) \to \top$. A OWPuzz is a pair $(\mathsf{Samp}, \mathsf{Ver})$ of algorithms. Samp is a QPT algorithm that outputs two classical bit strings s and k. The security requires that no QPT adversary, given s, can output k' such that $\mathsf{Ver}(s, k') \to \top$, where Ver is not necessarily QPT.

primitives could be built from weaker assumptions than those necessary for classical cryptography.[3]

3. "NanoCrypt": quantum cryptographic primitives which can exist even if **BQP** = **PP**, and which we do not know how to break using *any* classical oracle. This category contains exclusively (single-copy) "quantum output" primitives. Examples of this category are EFI pairs [BCQ23][4], quantum non-interactive bit commitments [Yan22], multiparty computation, and single-copy-secure PRSGs [MY22]. There exists an oracle relative to which these primitives are secure against one-query to an arbitrary classical oracle [LMW24]. Nevertheless, it is possible that these primitives all can indeed be broken by (multiple queries to) a **PP** oracle, and it is just that no attack has been discovered. There are also some black-box separations between some primitives in NanoCrypt and those in CountCrypt. In particular, there is an oracle relative to which single-copy-secure PRSGs (and therefore EFI pairs) exist, but (multi-copy secure) PRSGs do not [CCS24, AGL24].

Note that each of these categories also has an associated minimal primitive. Nearly every primitive in QuantuMania implies the existence of efficiently verifiable one-way puzzles (EV-OWPuzzs) [CGG24].[5] Like-wise, (inefficiently verifiable) one-way puzzles (OWPuzzs) appear to be minimal for CountCrypt [KT24, KT25], while EFI pairs seem minimal for NanoCrypt [MY24, BCQ23]. From this perspective, we can define QuantuMania as the class of primitives which can be used to build EV-OWPuzzs. We can define CountCrypt as the class of primitives which can be used to build OWPuzzs but not EV-OWPuzzs. And we can define NanoCrypt as the class of primitives which can be used to build EFI pairs but not OWPuzzs. For a figure illustrating the relationships between primitives in these three classes, see Figure 1.

While a great amount of progress has been made in recent years, there are a few primitives which we still do not know how to place within these broad categories.

Interactive QCCC primitives. One interesting class of quantum primitives are those where communication is required to be done over classical channels, namely, QCCC primitives. This class contains variants of most classical cryptographic primitives, where the only difference is that we allow the protocol itself to perform quantum operations. Examples include classical-communication key exchange (KE), classical-communication digital signatures, and EV-OWPuzz. [KT24, CGG24] showed that many QCCC primitives can be used to build EV-OWPuzzs. In particular, QCCC versions of encryption, signatures, and pseudorandomness all imply EV-OWPuzzs and so are broken if **BQP** = **QCMA**.

However, there are a few notable exceptions to this result. In particular, it is not clear how to build EV-OWPuzzs from QCCC primitives whose security game includes multiple rounds of communication. Key examples of such primitives, which we consider in this work, are QCCC KE and QCCC commitments. It is easy to see that QCCC KE and QCCC commitments can be broken with a **PP** oracle. Thus, QCCC KE and QCCC commitments should lie in either QuantuMania or CountCrypt. If we look from the perspective of minimal primitives, even less is known. It is unknown whether or not QCCC KE and QCCC commitments can be used to build any of EV-OWPuzzs, OWPuzzs,

[3] Unlike the classical MicroCrypt, CountCrypt does not collapse to a single primitive. There exists a black-box separation between OWSG and OWPuzz, two CountCrypt primitives [BMM+24, BCN24].

[4] An EFI pair is a two efficiently generatable quantum states that are statistically far but computationally indistinguishable.

[5] An EV-OWPuzz is a variant of OWPuzzs where Ver is quantum polynomial-time (QPT).

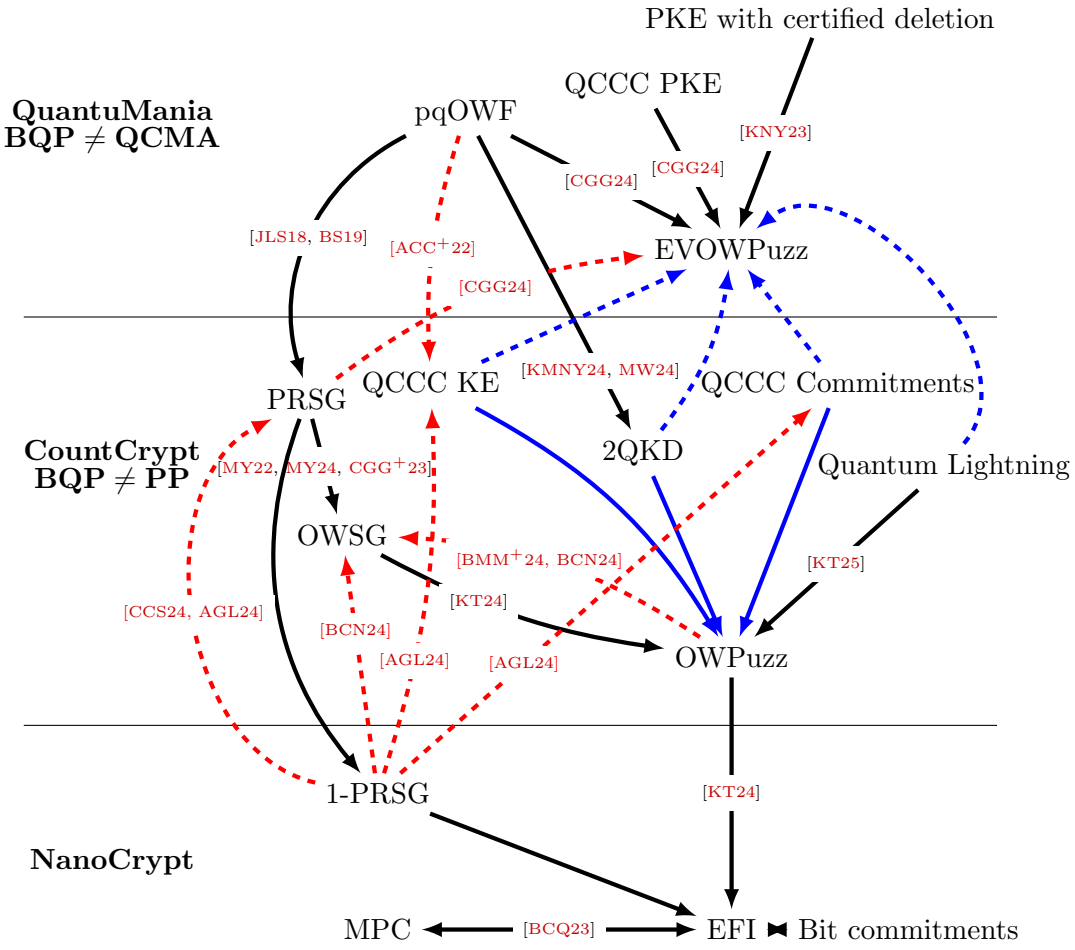


Figure 1: A graph of some known implications between primitives in QuantuMania, CountCrypt, and NanoCrypt. Dashed lines represent black-box separations. Blue lines are new in our work. pqOWF means post-quantum (i.e., quantumly-secure) OWFs. OWSG refer to pure state one-way state generators [MY22].

or EFI pairs[6].

2-round quantum key distribution. One of the most surprising initial results in quantum cryptography is that, with quantum communication, KE can exist *unconditionally* [Wie83, BB14][7]. Such protocols, termed quantum key distribution or QKD, are secure when performed over an authenticated classical channel and an *arbitrary, corrupted* quantum channel. Formally, a QKD protocol is a protocol where Alice and Bob agree on a shared key over a public authenticated classical channel and a public arbitrary quantum channel. In particular, a good QKD protocol should satisfy three properties.

1. (Correctness): If the protocol is executed honestly, then Alice and Bob output the same key.

2. (Validity): An adversarial party Eve, who controls the quantum channel, cannot make Alice and Bob output different keys without either of them noticing and outputting $\perp$.

3. (Security): If either Alice or Bob does not output $\perp$, then Eve can not learn their final key.

A two-round QKD protocol, or 2QKD is simply a QKD protocol where Alice and Bob each send at most one message. It is known that 2QKD protocols can be built from post-quantum OWFs [KMNY24, MW24].

Interestingly, QKD can be performed in three rounds and still maintain information-theoretic security [GYZ17][8]. On the other hand, it is known that 2-round QKD (2QKD) is impossible information-theoretically [MW24]. However, there is no other known lower bound on the computational hardness of 2QKD. The best known upper bound is that 2QKD can be constructed from OWFs [KMNY24, MW24]. It is entirely an open question whether 2QKD lies in QuantuMania, CountCrypt, or NanoCrypt.

Quantum money/quantum lightning. Quantum money [Wie83, JLS18, AC12] is one of the most fundamentally "quantum" cryptographic primitives. Quantum money essentially models digital cash. In a quantum money scheme, a bank has the ability to mint quantum states which represent some currency. Money states should be verifiable: either the bank or other users should be able to tell if a money state is valid. Money states should also be uncloneable: it should not be possible for users to take several copies of a money state and turn it into more copies. Note that quantum money is fundamentally impossible if we require that states are classical. This is because it is easy to copy any classical string.

For the purposes of this work, we will consider a stronger variant of quantum money, known as quantum lightning [Zha21]. A quantum lightning scheme consists of a minter and a verifier. The minter produces a money state along with a corresponding serial number. The verifier, given a money state and a serial number, checks if the serial number matches. Security says that no adversary can construct two money states which verify under the same serial number. This is a stronger definition than standard quantum money, since adversary can choose which state they are trying to clone.

[6]Note that [KT24] constructs OWPuzzs from QCCC commitments with a *non-interactive* opening phase. However, this is not enough for our purposes. In particular, our construction of QCCC commitments in the oracle setting has an *interactive* opening phase.

[7]Assuming the existence of classical authenticated channel.

[8]Note that in this protocol, only one party learns if the execution succeeds.

It is again easy to see that quantum lightning is broken if **BQP** = **PP**, since given the verification algorithm a QPT algorithm querying the **PP** oracle can synthesize a valid state [KT25, CGG+23, INN+22]. Furthermore, recent work has shown that OWPuzzs can be built from quantum lightning [KT25], but given a quantum lightning scheme, it is not known how to build an EV-OWPuzz. Thus, it seems that quantum lightning lies either in CountCrypt or QuantuMania, but it is not clear which is the case. Note that the same reasoning applies to quantum money schemes as well.

1.1 Our Results

The major question of our work is the following.

Do these primitives (QCCC KE, QCCC commitments, 2QKD, quantum lightning) lie in CountCrypt or QuantuMania?

More specifically,

Do these primitives exist even if **BQP** = **QMA** *or* **BQP** = **QCMA***?* Do these primitives imply EV-OWPuzzs or OWPuzzs?

In this paper, we show the following results.

1. There exists a unitary oracle relative to which **BQP** = **QCMA**, but QCCC commitments (with a single-message commit phase and a two-message reveal phase) and multiparty QCCC non-interactive KE (NIKE) (and therefore 2QKD as well) exist.
2. There exists a unitary oracle relative to which **BQP** = **QMA**, but quantum lightning exists.
3. If any of (poly-round) QCCC KE, QCCC commitments, or 2QKD exist, then OWPuzzs exist. (It is already known that quantum lightning can be used to build OWPuzzs [KT24]. A concurrent work also showed that 2QKD implies OWPuzz through similar techniques [QRZ25].)

Together, these results imply that quantum lightning, quantum money, QCCC (NI)KE and QCCC commitments, as well as 2QKD are all CountCrypt primitives.

1.2 Technical Overview

Here we provide a high-level overview of our proofs.

Oracle relative to which QCCC NIKE exist and BQP = QCMA. The key idea behind this oracle is that it functions as an idealized NIKE, but where the internal state is forced to be quantum. As a toy example, consider the simple idealized NIKE protocol defined as follows.[9]

1. Let $\mathsf{RO} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be a random function accessible only to Alice and Bob.
2. Alice chooses x and sends x to Bob.
3. Bob chooses y and sends y to Alice.

[9] Although we can actually construct multiparty NIKE, here for simplicity we explain the two-party case. The multiparty case is similar.

4. Alice and Bob both output $\mathsf{RO}(x, y)$.

Note that this protocol is secure against any adversary who does not have access to RO. This is because just given x and y, the adversary has no way to compute $\mathsf{RO}(x, y)$.

In order to define an oracle relative to which this NIKE protocol is secure, we need an oracle which allows Alice and Bob to query $\mathsf{RO}(x, y)$, while other parties cannot. To do this, we associate to each x and y a quantum key $|\phi_x\rangle$ and $|\phi_y\rangle$. We then guarantee that the only way for a party to compute $|\phi_x\rangle$ is if it chose x itself. We mandate that in order to query $\mathsf{RO}(x, y)$, a party must have access to either $|\phi_x\rangle$ or $|\phi_y\rangle$.

As a first attempt, we will define a CPTP oracle (that is, a quantum oracle which is allowed to make measurements) as follows. Our oracle will be defined respective to a Haar random unitary U and a random function $\mathsf{RO} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$. Our oracle consists of the following three oracles, SG^{CPTP}, Mix^{CPTP}, and $PSPACE$:[10]

1. SG^{CPTP}: samples a bit string x uniformly at random and outputs $(x, |\phi_x\rangle)$. Here $|\phi_x\rangle \coloneqq U|x\rangle$.
2. $Mix^{CPTP}(x, y, |\psi\rangle)$: applies $U^\dagger$ on $|\psi\rangle$, measures the state in the computational basis, and outputs $\mathsf{RO}(x, y)$ if and only if the measurement result is x or y.
3. $PSPACE$: an oracle for a **PSPACE**-complete problem.[11]

This leads to the following natural NIKE protocol.

1. Alice runs $SG^{CPTP} \to (x, |\phi_x\rangle)$ and sends x to Bob.
2. Bob runs $SG^{CPTP} \to (y, |\phi_y\rangle)$ and sends y to Alice.
3. Alice outputs $Mix^{CPTP}(x, y, |\phi_x\rangle) \to \mathsf{RO}(x, y)$.
4. Bob outputs $Mix^{CPTP}(x, y, |\phi_y\rangle) \to \mathsf{RO}(x, y)$.

Security follows from the fact that given x and y, it is impossible for an eavesdropper to compute $|\phi_x\rangle$ or $|\phi_y\rangle$, since with high probability queries to SG^{CPTP} will never return these values. However, without $|\phi_x\rangle$ or $|\phi_y\rangle$, it is impossible to query $\mathsf{RO}(x, y)$, and therefore the agreed key appears uniformly random. Note that since this security holds information-theoretically, it also holds against adversaries with access to the $PSPACE$ oracle.

Making the oracle unitary. While the oracles $SG^{CPTP}, Mix^{CPTP}, PSPACE$ do indeed give an oracle where QCCC cryptography exists but $\mathbf{BQP} = \mathbf{QCMA}$, the fact that the oracles are CPTP is a major limitation. In particular, CPTP oracle separations only rule out black-box reductions which do not purify the adversary or query its inverse. These are very common techniques in the quantum setting. Thus, the quantum cryptography community has a strong preference for unitary (or even better, classical) oracle separations, and much work has been done to avoid weaker models [GZ25, BMM+24, BCN24].

Our first observation is that the oracles SG^{CPTP} and Mix^{CPTP} have natural unitary "purifications".

1. SG: the unitary which swaps the state $|0\rangle$ with the state $|\psi\rangle \coloneqq \sum_k |k\rangle |\phi_k\rangle$. More formally, it is the unitary, $|\psi\rangle\langle 0| + |0\rangle\langle\psi| + (I - |0\rangle\langle 0| - |\psi\rangle\langle\psi|)$.[12]

[10] Precisely speaking, we define sequences $\{SG_n^{CPTP}\}_{n\in\mathbb{N}}$ and $\{Mix_n^{CPTP}\}_{n\in\mathbb{N}}$ of oracles defined for each $n \in \mathbb{N}$, but for simplicity, here we ignore that point.

[11] We use the normal font for the oracle, and the bold font for the complexity class.

[12] For the notational simplicity, for an n, we often write $|0^n\rangle$ just as $|0\rangle$.

2. Mix: the unitary[13]

$$\sum_{x,y} |x\rangle\langle x| \otimes |y\rangle\langle y| \otimes (|\phi_x\rangle\langle\phi_x| + |\phi_y\rangle\langle\phi_y|) \otimes X^{\mathsf{RO}(x,y)} \otimes X \tag{1}$$

$$+ \sum_{x,y} |x\rangle\langle x| \otimes |y\rangle\langle y| \otimes (I - |\phi_x\rangle\langle\phi_x| - |\phi_y\rangle\langle\phi_y|) \otimes I \otimes I \tag{2}$$

acting on five registers. Intuitively, it works as follows. If the third register is $|\phi_x\rangle$ or $|\phi_y\rangle$ when the first register is x and the second register is y, then it adds $\mathsf{RO}(x,y)$ to the fourth register, and flips the bit of the fifth register:

$$|x\rangle\,|y\rangle\,|\phi_x\rangle\,|z\rangle\,|b\rangle \mapsto |x\rangle\,|y\rangle\,|\phi_x\rangle\,|z \oplus \mathsf{RO}(x,y)\rangle\,|\bar{b}\rangle\,, \tag{3}$$

$$|x\rangle\,|y\rangle\,|\phi_y\rangle\,|z\rangle\,|b\rangle \mapsto |x\rangle\,|y\rangle\,|\phi_y\rangle\,|z \oplus \mathsf{RO}(x,y)\rangle\,|\bar{b}\rangle\,, \tag{4}$$

Otherwise, it acts as the identity.

We will show the following.

Theorem 1.1 (Theorem 6.1 restated)**.** *QCCC NIKE exists relative to* $(SG, Mix, PSPACE)$.

Note that two-party QCCC NIKE is a special case of 2QKD where no quantum channel is used. We therefore obtain the following corollary.

Corollary 1.2. *2QKD exists relative to* $(SG, Mix, PSPACE)$.

Remark 1.3. *Note that by allowing* Mix *to take in an arbitrary number of inputs, we can define new oracles such that multi-party NIKE exists but* $\mathbf{BQP} = \mathbf{QCMA}$. *We will omit this argument, since it follows analogously to Theorem 1.1.*

The key idea to show Theorem 1.1 is that we can simulate SG and Mix by the following CPTP oracles:

1. On initialization, sample a fresh random unitary $\widetilde{U}$. Sample strings $k_1, \ldots, k_t$.
2. SG_{Samp}: on the ith query, output $|k_i\rangle \otimes (\widetilde{U}|k_i\rangle)$.
3. Ver_{Samp}: apply the unitary

$$\sum_{i\in[t]} |k_i\rangle\langle k_i| \otimes \widetilde{U}|k_i\rangle\langle k_i|\widetilde{U}^\dagger \otimes X + \left(I - \sum_{i\in[t]} |k_i\rangle\langle k_i| \otimes \widetilde{U}|k_i\rangle\langle k_i|\widetilde{U}^\dagger\right) \otimes I, \tag{5}$$

which, intuitively, works as follows: If the second register is $\widetilde{U}|k_i\rangle$ when the first register is k_i, then it flips the bit of the third register as

$$|k_i\rangle \otimes \widetilde{U}|k_i\rangle \otimes |b\rangle \mapsto |k_i\rangle \otimes \widetilde{U}|k_i\rangle \otimes |\bar{b}\rangle\,. \tag{6}$$

Otherwise, it acts as the identity.

That is, there exists a simulator $(\mathrm{Sim}_{SG}^{SG_{\mathrm{Samp}}}, \mathrm{Sim}_{Mix}^{Ver_{\mathrm{Samp}}})$ which is indistinguishable from (SG, Mix) under bounded query access, even against adversaries with non-uniform advice that can depend on U and RO.

In particular, to analyze an adversary for our NIKE protocol, we can replace all instances of (SG, Mix) in the security game with $(\mathrm{Sim}_{SG}^{SG_{\mathrm{Samp}}}, \mathrm{Sim}_{Mix}^{Ver_{\mathrm{Samp}}})$. Now, a similar argument to before can be used to show security.

[13] X is the Pauli X operator. For an n-bit string x, $X^x := \bigotimes_{i=1}^n X^{x_i}$.

Constructing the simulator. Our goal is to show the following theorem.

Theorem 1.4. *(Lemma 4.1, restated) With high probability over the oracles U and RO, there exists an efficient simulator such that for all oracle algorithms $\mathcal{A}$ and poly-length advice strings c,*

$$\left|\Pr[\mathcal{A}^{SG,Mix,PSPACE}(1^n,c)\to 1]-\Pr[\mathcal{A}^{Sim_{SG}^{SG_{Samp}},Sim_{Mix}^{Ver_{Samp}},PSPACE}(1^n,c)\to 1]\right|\leq \mathsf{negl}(n). \tag{7}$$

To show the theorem, we first take inspiration from the super-concentration bound for Haar random unitaries from [Kre21], which observes that with all but doubly exponential probability over U, any efficient algorithm querying U acts identically if the algorithm instead samples U itself. Thus, by a union bound over all non-uniform advice strings, $\mathcal{A}^{SG,Mix,PSPACE}(1^n,c)$ acts the same on all advice when we replace U with a freshly sampled Haar random unitary $\widetilde{U}$.

Next, it is known from [Zha24, BMM+24, GZ25] that for a Haar random state $|\phi\rangle$, the map

$$|0\rangle\mapsto|\phi\rangle \tag{8}$$

can be simulated using many copies of $|\phi\rangle$. A straightforward generalization of this approach then shows that the map

$$|0\rangle\mapsto\sum_k|k\rangle|\phi_k\rangle \tag{9}$$

can be simulated using many samples of $(k,|\phi_k\rangle)$. In particular, SG utilizing a fresh random unitary $\widetilde{U}$ can be simulated using SG_{Samp}.

Next we observe that since $\mathcal{A}^{\mathrm{Sim}_{SG}^{SG_{\mathrm{Samp}}},Mix,PSPACE}(1^n,c)$ can never receive $|\phi_k\rangle$ for a k not sampled by $\mathrm{Sim}_{SG}^{SG_{\mathrm{Samp}}}$, we can enforce Mix to always ignore inputs not produced by $\mathrm{Sim}_{SG}^{SG_{\mathrm{Samp}}}$.

The final observation we make is that we can replace $\mathsf{RO}(k_1,k_2)$ for k_1,k_2 sampled by SG_{Samp} with random values. The key intuition is as follows: *if you can only query a fixed random function at random values, then that fixed random function appears like a uniform random function.* We call this a concentration bound for random functions on random inputs. We will explain more details about it in the next paragraph.

Hence we can simulate Mix by using Ver_{Samp} to coherently check whether the inputs are $(x,y,|\phi_x\rangle)$ or $(x,y,|\phi_y\rangle)$ for some x,y sampled by SG_{Samp}, and then if so outputting some fixed $z_{x,y}$ sampled uniformly at random during initialization.

Putting these together, we construct the simulator $(\mathrm{Sim}_{SG}^{SG_{\mathrm{Samp}}},\mathrm{Sim}_{Mix}^{Ver_{\mathrm{Samp}}})$. For full details on the simulator, see Section 4.1.

Concentration bound for random functions. Let us explain here more details about the concentration bound for random functions. That is, with high probability over the choice of some fixed random oracle RO, RO will be indistinguishable under random query access from a freshly sampled random oracle $\widetilde{\mathsf{RO}}$. The intuition here stems from the similarity between a fixed random oracle and the auxiliary-input random oracle model (AI-ROM) [Unr07].

The AI-ROM is a variant of the random oracle model which allows adversaries to have non-uniform advice *depending on the chosen random oracle.* One of the most useful ways

to prove things in the AI-ROM is the so-called presampling technique [Unr07, CDGS18, GLLZ21]. A bit-fixing, or presampling, oracle is a random function where its evaluation on polynomially many fixed points is chosen adversarially. The presampling technique states that any protocol secure with bit-fixing oracles is also secure in the AI-ROM.

When adversaries have quantum query access to the oracle, presampling becomes a bit more complicated. [GLLZ21] shows that in the quantum setting, any protocol secure with a so-called "postselected" oracle is also secure in the AI-ROM. In particular, a "postselected" oracle is an oracle which has been sampled from the conditional distribution satisfying some efficient quantum predicate.

Intuitively, the only way to distinguish a fixed random function RO from a fresh $\widetilde{\mathsf{RO}}$ is by hardcoding some information about RO into the distinguisher. This is morally equivalent to giving the distinguisher non-uniform advice depending on RO, i.e. the AI-ROM. In fact, a straightforward argument shows that presampling/postselecting lemmas also hold for fixed random functions under classical/quantum query access respectively. See Corollary 3.3 for details.

It thus remains to show that for a postselected random oracle, queries to random points look random. This follows from a straightforward application of the compressed oracle framework [Zha19]. In this framework, a random oracle is instead represented by a database containing a superposition of lists of queried points. In particular, after postselection the database can only contain at most a polynomial number of points, so any point chosen at random will necessarily be absent from the database. Queries to points outside of the database will always return uniform random values, and therefore queries to random points on a postselected oracle will return uniform random values.

BQP = QCMA. Since we know that NIKE exists relative to $(SG, Mix, PSPACE)$, it remains to show that

Theorem 1.5 (Theorem 8.1 restated)**.**

$$\mathbf{BQP}^{SG,Mix,PSPACE} = \mathbf{QCMA}^{SG,Mix,PSPACE}. \tag{10}$$

In particular, we show that for any input x and oracle verifier $\mathsf{Ver}^{SG,Mix,PSPACE}$ for a language in $\mathbf{QCMA}^{SG,Mix,PSPACE}$, there exists a verifier $\widetilde{\mathsf{Ver}}$ such that for all x, w,

$$\Pr[\widetilde{\mathsf{Ver}}^{PSPACE}(x, w) \to 1] \approx \Pr[\mathsf{Ver}^{SG,Mix,PSPACE}(x, w) \to 1]. \tag{11}$$

Thus, given x and $\mathsf{Ver}^{SG,Mix,PSPACE}$, we can use the $PSPACE$ oracle to check if $\widetilde{\mathsf{Ver}}^{PSPACE}$ has a witness w accepting x. Since $\widetilde{\mathsf{Ver}}^{PSPACE}$ acts similarly to $\mathsf{Ver}^{SG,Mix,PSPACE}$, this also reveals whether $\mathsf{Ver}^{SG,Mix,PSPACE}$ has a witness accepting x. In other words, we can use the $PSPACE$ oracle to decide languages in $\mathbf{QCMA}^{SG,Mix,PSPACE}$.

The proof of this result is a straightforward application of Theorem 1.4. However, some care must be taken to simulate SG, Mix on small domains. Following the approach of [Kre21], for $\ell = O(\log n)$ we construct simulators to emulate SG, Mix efficiently by applying a learning algorithm to learn their behavior up to minimal error.

Existence of QCCC commitments. We can observe that under the same oracle, we can also build QCCC commitments. Our protocol goes as follows.

1. The committer queries $|0\rangle$ to the unitary oracle SG to generate $SG\,|0\rangle = \sum_k |k\rangle|\phi_k\rangle$ and measures the first register to obtain the pair $(x, |\phi_x\rangle)$ of the measurement result

x and the post-measurement state $|\phi_x\rangle$. It repeats this process until the first bit of x is the bit that it is trying to commit to. The committer then sends all but the first bit of x to the receiver, which we will denote $x_{>1}$.

2. To open to a bit b, the receiver queries $|0\rangle$ to SG to generate $SG|0\rangle = \sum_k |k\rangle|\phi_k\rangle$, measures the first register, obtains $(y, |\phi_y\rangle)$, and sends y to the committer.[14]

3. Upon receiving y, the committer queries $|x\rangle\,|y\rangle\,|\phi_x\rangle\,|0\rangle\,|0\rangle$ to the unitary oracle Mix to generate $Mix\,|x\rangle\,|y\rangle\,|\phi_x\rangle\,|0\rangle\,|0\rangle$, measures the fourth register to get the result c, and sends c to the receiver.

4. The receiver sets $x = (b, x_{>1})$, queries $|x\rangle\,|y\rangle\,|\phi_y\rangle\,|0\rangle\,|0\rangle$ to the unitary oracle Mix to generate $Mix\,|x\rangle\,|y\rangle\,|\phi_y\rangle\,|0\rangle\,|0\rangle$, measures the fourth register, and accepts if the result is c.

Hiding of this protocol is trivial, since the committer just sends a random string in the commit phase. Binding is slightly more difficult, but the proof follows very similar lines to the security of NIKE. In particular, the only way that the committer can open to a bit b is if it knows $|\phi_{b,x_{>1}}\rangle$. However, it is impossible for the receiver to know both $|\phi_{0,x_{>1}}\rangle$ and $|\phi_{1,x_{>1}}\rangle$, since it would have needed to receive both of these separately from SG. Thus, once the committer knows $|\phi_{b,x_{>1}}\rangle$, it can not change its mind and decide to open to $\bar{b}$.

We therefore oftain the following theorem.

Theorem 1.6 (Theorem 7.3, restated)**.** *QCCC commitments exist relative to* $(SG, Mix, PSPACE)$.

Existence of quantum lightning. Note that it is easy to see that, for any key k, no adversary can obtain two copies of $|\phi_k\rangle$. This follows immediately from the fact that the oracle SG_{Samp} only ever returns at most one copy of $\widetilde{U}\,|k\rangle$ for each k sampled during initialization. This leads to the natural quantum lightning protocol as follows.

1. **Mint**: Query $|0\rangle$ to SG to generate $SG\,|0\rangle = \sum_k |k\rangle|\phi_k\rangle$, measure the first register to obtain the pair $(k, |\phi_k\rangle)$ of the measurement result k and the post-measurement state $|\phi_k\rangle$. Output k as the serial number and $|\phi_k\rangle$ as the bolt state.

2. **Verify**$(k, |\psi\rangle)$: Query $|k\rangle\,|0\rangle\,|\psi\rangle\,|0\rangle\,|0\rangle$ to Mix to obtain $Mix\,|k\rangle\,|0\rangle\,|\psi\rangle\,|0\rangle\,|0\rangle$, and measure the last register. Accept if the result is 1, and reject otherwise.

The security goes via a reduction to the complexity-theoretic no-cloning theorem defined by [AC12]. In particular, this theorem states that with only polynomially many queries to a verification oracle, it is impossible to clone a Haar random state with non-negligible success probability. To clone a Haar random state $|\psi\rangle$ using a quantum lightning breaker, we will run the breaker on $(\text{Sim}_{SG}^{SG_{\text{Samp}}}, \text{Sim}_{Mix}^{Ver_{\text{Samp}}})$, with a random $U\,|k_i\rangle$ replaced by $|\psi\rangle$. With probability approximately $\frac{1}{\#\text{ queries made by the lightning breaker}}$, the quantum lightning breaker will clone $|\psi\rangle$, violating complexity-theoretic no-cloning.

Theorem 1.7 (Theorem 5.1, restated)**.** *Quantum lightning exists relative to* $(SG, Mix, PSPACE)$.

Remark 1.8. *Note that for the setting of quantum lightning, the random oracle is entirely redundant. Thus, we can replace* Mix *with an oracle* $Ver\,|k\rangle\,|\phi_k\rangle\,|b\rangle \mapsto |k\rangle\,|\phi_k\rangle\,|\bar{b}\rangle$. *Note that relative to this oracle, the proof of [Kre21] can be easily adapted to show that* $\mathbf{BQP}^{SG,Ver,PSPACE} = \mathbf{QMA}^{SG,Ver,PSPACE}$.

[14] If we wish, this step can be done in the commit phase instead of the opening phase. One advantage of doing it in the opening phase is that the receiver does not need to store state long term and the commit phase becomes non-interactive.

Building OWPuzzs from QCCC KE. To construct OWPuzzs from (poly-round) QCCC KE, we look to the construction of OWFs from classical KE [IL89]. In particular, this paper constructs distributional OWFs from KE, and shows that distributional OWFs and OWFs are equivalent. A distributional OWF is a OWF which is hard to *distributionally* invert. That is, given $y = f(x)$, it is hard to sample from the set $f^{-1}(y)$.

Recent work [CGG24] analogously defines a distributional variant of OWPuzzs. In particular, a distributional OWPuzzs consists of a sampler producing puzzle-key pairs (s, k) such that given a puzzle s, it is hard to sample from the conditional distribution over keys k. It turns out that distributional OWPuzzs and OWPuzzs are equivalent [CGG24].

It is then not hard to build a distributional OWPuzzs from a QCCC KE. In particular, the puzzle will be the transcript (i.e., the sequence of messages exchanged) between Alice and Bob while the OWPuzz key will be the corresponding shared key of the KE produced by Alice. For a random transcript, the conditional distribution over Alice's key will be close to the constant distribution, as otherwise Bob will disagree with Alice on the shared key. Distributionally inverting this puzzle is enough to recover the shared key and thus break the KE.

Theorem 1.9 (Theorem 9.1, restated)**.** *If there exists (poly-round) QCCC KE, then there exist OWPuzzs.*

Building OWPuzzs from QCCC commitments. To build OWPuzzs from (general poly-round) QCCC commitments, we again take advantage of distributional OWPuzzs. In particular, we observe that if distributional OWPuzzs do not exist, then it is possible to do conditional sampling for any efficiently sampleable distribution. This gives an approach to break hiding for QCCC commitment schemes.

In particular, given the transcript τ of the commit phase for a commitment to a bit b, a corrupted receiver can sample the committer's first message in the opening phase conditioned on τ.[15] The receiver can then repeat this process to sample the committer's second and third messages, up until the entire opening phase has been simulated. Since the receiver can run the entire opening phase on its own, it can learn the output b. By binding, this should be the same as the committed bit, and therefore this attack breaks hiding.

Theorem 1.10 (Theorem 9.2, restated)**.** *If there exist QCCC commitments, then there exist OWPuzzs.*

Building OWPuzzs from 2QKD. A concurrent work [KT25] defines a quantum version of OWPuzzs, state puzzles, and shows that state puzzles are equivalent to OWPuzzs. A state puzzle is essentially a OWPuzz where the key is a quantum state. Since the key is quantum, there is a canonical verification algorithm which just projects onto the target state.[16]

We observe that the existence of a 2QKD protocol immediately gives a state puzzle. In particular, the puzzle will consist of Alice's first classical message to Bob, and the key will be Alice's internal state as well as her first quantum message.[17] In particular, if this is not a state puzzle, then we can attack the 2QKD protocol by replacing Alice's quantum

[15] Remember that we are now considering general commitments where the opening phase is not necessarily non-interactive.

[16] Note that we can always assume that the key is a pure state, since adding the extra qubits from its purification into the key only improves security.

[17] Remember that in QKD we have authenticated classical channel and non-authenticated quantum channel. We therefore assume that Alice's first message consists of a quantum part and a classical part.

message with the result from the state puzzle breaker. This will allow an adversary to impersonate Alice and obtain a corresponding private state. Since Alice does not send any more messages beyond this, Bob will not be able to detect this impersonation and so the adversary will be able to predict Bob's output.

Theorem 1.11 (Theorem 9.8, restated)**.** *If there exist 2QKD, then there exist OWPuzzs.*

Paper organization We begin with preliminaries and definitions in Section 2. In Section 3, we outline and modify the quantum version of the presampling lemma from [GLLZ21]. In Section 4, we introduce our oracles $SG, Mix, PSPACE$ and prove the key lemma we will use to simulate this oracle by a stateful efficient procedure. This argument contains the novel key ideas needed to show all of our results relative to $SG, Mix, PSPACE$. We include the proofs of security for quantum lightning, key exchange, and commitments relative to $SG, Mix, PSPACE$ in Sections 5 to 7. Section 8 contains the proof that $\mathbf{BQP}^{SG,Mix,PSPACE} = \mathbf{QCMA}^{SG,Mix,PSPACE}$. We conclude with the constructions of one-way puzzles from CountCrypt primitives in Section 9.

2 Preliminaries

In this section, we provide a refresher on basics of quantum information, Haar measure and its concentration, as well as definitions of primitives in quantum cryptography.

2.1 Notation

Throughout the paper, $[n]$ denotes the set of integers $\{1, 2, \ldots, n\}$. If X is a set, we use $x \leftarrow X$ to denote that x is sampled uniformly at random from X. Similarly, if A is an algorithm and x is an input, we use $y \leftarrow A(x)$ to say that y is the output of A on input x. For a distribution D, $x \leftarrow D$ means that x is sampled according to D. A function f is *negligible* if for every constant $c > 0$, $f(n) \leq \frac{1}{n^c}$ for all sufficiently large $n \in \mathbb{N}$. We will use $\mathsf{negl}(n)$ to denote an arbitrary negligible function, and $\mathsf{poly}(n)$ to denote an arbitrary polynomially bounded function. For two interactive algorithms A and B that interact over a classical or quantum channel, $A \rightleftarrows B$ means the execution of the interactive protocol, or the distribution of the transcript, i.e. the sequence of messages exchanged between A and B. $A \leftrightarrows B \to y$ means that the execution of the interactive protocol produces output y. We use the abbreviation QPT for a uniform quantum polynomial time algorithm. We will also use non-uniform QPT to refer to quantum polynomial time algorithms with polynomial *classical* advice. We use the notation $A^{(\cdot)}$ to refer to a (classical or quantum) algorithm that makes queries to an oracle. For an algorithm $\mathcal{A}$, we use $|\mathcal{A}|$ to refer to the length of some canonical description of $\mathcal{A}$. We use $\mathsf{TD}(\rho, \sigma) = \frac{1}{2}\|\rho - \sigma\|_1$ to denote the trace distance between density matrices ρ and σ. For two distributions D_1, D_2, we use $\Delta(D_1, D_2) = \frac{1}{2}\sum_x |\Pr[x \leftarrow D_1] - \Pr[x \leftarrow D_2]|$ to denote the total variation distance (aka statistical distance) between D_1 and D_2. For algorithms which take in a unary security parameter 1^n, we will sometimes omit 1^n as input when the value is implicit. We will use ω_T to denote a (canonical) Tth root of unity. That is, $\omega_T^T = \omega_T^0 = 1$, and for all $0 < i < T$, $\omega_T^i \neq 1$. For a projector Π, we will say "apply the measurement $\{\Pi, I - \Pi\}$" to mean apply the map $\rho \mapsto \Pi\rho\Pi + (I - \Pi)\rho(I - \Pi)$.

2.2 Quantum Information

A quantum channel $\mathcal{A}$ is any completely-positive trace-preserving (CPTP) map. Intuitively, quantum channels represent all "physical" processes. That is, the set of CPTP

maps is the quantum analogue of the set of all functions.

For a quantum channel $\mathcal{A}$, we let $\|A\|_\diamond$ denote its *diamond norm*. The diamond norm and trace distance satisfy the following relation:

Theorem 2.1 ([NC11])**.** *Let $\mathcal{A}$ and $\mathcal{B}$ be quantum channels and ρ be a density matrix. Then,* $\mathsf{TD}(\mathcal{A}(\rho), \mathcal{B}(\rho)) \leq ||\mathcal{A} - \mathcal{B}||_\diamond$.

Hence if two quantum channels are close in diamond distance, triangle inequality implies that they are indistinguishable via bounded query access.

We note the following relationship between trace distance and fidelity of pure states.

Lemma 2.2. *Let $|\phi\rangle, |\psi\rangle$ be two pure states. Then,*

$$\mathsf{TD}(|\phi\rangle\langle\phi|, |\psi\rangle\langle\psi|) = \sqrt{1 - |\langle\phi|\psi\rangle|^2}. \tag{12}$$

In this work, we will often consider quantum channels corresponding to performing some projective measurement, which we call measurement channels.

Definition 2.3 (Measurement Channels)**.** *Let $\mathcal{M} = \{\Pi, I - \Pi\}$ be a binary measurement corresponding to a projector Π. The measurement channel corresponding to $\mathcal{M}$ is the following channel:*

$$\rho \mapsto \Pi\rho\Pi + (I - \Pi)\rho(I - \Pi). \tag{13}$$

We make use of the following fact about measurement channels.

Lemma 2.4 ([Kuk18])**.** *Let $\mathcal{A} = \{\Pi_A, I - \Pi_A\}$, $\mathcal{B} = \{\Pi_B, I - \Pi_B\}$ be two measurement channels. Then*

$$||\mathcal{A} - \mathcal{B}||_\diamond = 2 \max_\rho |\mathrm{Tr}(\Pi_A\rho) - \mathrm{Tr}(\Pi_B\rho)|. \tag{14}$$

2.3 The Compressed Oracle Technique/The Recording Method

Here we explain the compressed oracle technique and the recording method [Zha19]. First, for the random oracle $H : \{0,1\}^m \to \{0,1\}^n$, we denote the usual quantum random oracle, or the standard oracle by Sto. In this model, H is represented as a truth table: a vector of size 2^m where each component is an n bit string. The compressed oracle for $H : \{0,1\}^m \to \{0,1\}^n$, CStO, acts on the query and database registers $(\mathbf{Q}, \mathbf{D})$, where the database register $\mathbf{D}$ will be a collection D of input output pairs (x, y), and $(x, y) \in D$ denotes that the random oracle has been specified to have value y on input x. We will write $D(x) = y$ in this case. If, for an input x, there is no $(x, y) \in D$, we will write $D(x) = \perp$, indicating that the function has not been specified on x.

$$\mathsf{CStO} \coloneqq \mathsf{Decomp} \circ \mathsf{CStO}' \circ \mathsf{Decomp} \circ \mathsf{Increase},$$

We will refer to the unitary CStO as the compressed standard oracle. where $\mathsf{Increase}$ is the procedure which initializes a new register $|(\perp, 0^n)\rangle$ and appends it to the end: $\mathsf{Increase}\,|x, y\rangle_{\mathbf{Q}} \otimes |D\rangle_{\mathbf{D}} = |x, y\rangle \otimes |D\rangle\,|(\perp, 0^n)\rangle$. Decomp and CStO' are defined as follows:

- Decomp is defined by $|x,u\rangle_{\mathbf{Q}} \otimes |D\rangle_{\mathbf{D}} \mapsto |x,u\rangle_{\mathbf{Q}} \otimes \mathsf{Decomp}_x |D\rangle_{\mathbf{D}}$, where Decomp_x is defined by its action on basis states as follows. Let D be such that $D(x) = \perp$:

$$|D\rangle \mapsto \frac{1}{\sqrt{|\mathcal{Y}|}} \sum_{y\in\mathcal{Y}} |D \cup \{(x,y)\}\rangle \tag{15}$$

$$\frac{1}{\sqrt{|\mathcal{Y}|}} \sum_{y\in\mathcal{Y}} |D \cup \{(x,y)\}\rangle \mapsto |D\rangle \tag{16}$$

$$\frac{1}{\sqrt{|\mathcal{Y}|}} \sum_{y\in\mathcal{Y}} (-1)^{y\cdot u} |D \cup \{(x,y)\}\rangle \mapsto \frac{1}{\sqrt{|\mathcal{Y}|}} \sum_{y\in\mathcal{Y}} (-1)^{y\cdot u} |D \cup \{(x,y)\}\rangle \qquad \text{for } u \neq 0 \tag{17}$$

  In words, Decomp_x swaps the states $|D\rangle$ and $\frac{1}{\sqrt{|\mathcal{Y}|}} \sum_{y\in\mathcal{Y}} |D \cup \{(x,y)\}\rangle$, and acts as the identity on the space orthogonal to these two states.

- CStO' maps $|x,u\rangle_{\mathbf{Q}} \otimes |D\rangle_{\mathbf{D}} \mapsto |x, u \oplus D(x)\rangle_{\mathbf{Q}} \otimes |D\rangle_{\mathbf{D}}$.

Lemma 2.5 (Compressed oracle method [Zha19])**.** Ctso *and* Sto *are perfectly indistinguishable. That is, for any adversary* $\mathcal{A}$*, we have* $\Pr[A^{\mathsf{Ctso}}(\cdot) = 1] = \Pr[A^{\mathsf{Sto}}(\cdot) = 1]$.

Next, we will state the following theorem, which gives a general formulation of the overall state of $\mathcal{A}$ and the compressed standard oracle after $\mathcal{A}$ makes t queries, even conditioned on arbitrary measurement results.

Theorem 2.6 ([Zha19, GLLZ21])**.** *Let* $M, N \in \mathbb{N}$*, and let* $f : [M] \to [N]$*. For any algorithm* $\mathcal{A}$ *making* t *queries to a compressed standard oracle, assuming the overall state of* $\mathcal{A}$ *and the compressed standard oracleis* $|A^{\mathsf{Ctso}}\rangle = \sum_{z,D} \alpha_{z,D} |z\rangle_{\boldsymbol{Q}} |D\rangle_{\boldsymbol{D}}$ *where* $\boldsymbol{Q}$ *is the query register and* $\boldsymbol{D}$ *is the database register of the oracle,*

$$\mathop{\mathbb{E}}_{f:[M]\to[N]} \left[|A^f\rangle\langle A^f| \right] = \mathrm{Tr}_{\mathbf{D}} \left[\left| \mathcal{A}^{\mathsf{Ctso}} \right\rangle \left\langle \mathcal{A}^{\mathsf{Ctso}} \right| \right]. \tag{18}$$

where $|\mathcal{A}^f\rangle$ *is the state of* $\mathcal{A}$ *after making* t *queries to* f*. Furthermore, define*

$$\Pi_{\leq t} := \textit{projection onto the set of databases } D : |D| \leq t. \tag{19}$$

Then the $\boldsymbol{D}$ *register will only have support on all databases* D *which lie within the subspace specified by* $\Pi_{\leq t}$*. That is,*

$$\mathrm{Tr}\left((I \otimes \Pi_{\leq t}) \left| \mathcal{A}^{\mathsf{Ctso}} \right\rangle \left\langle \mathcal{A}^{\mathsf{Ctso}} \right| \right) = 1. \tag{20}$$

Moreover, this is true even if the state is conditioned on arbitrary outcomes (with non zero probability) of $\mathcal{A}$*'s intermediate measurements.*

2.4 One Way To Hiding

We use the following lemma which combines [AHU18, Theorem 3] and [AHU18, Lemma 8], where the latter is rooted in Vazirani's *Swapping Lemma* [Vaz98].

Lemma 2.7 (One-Way-to-Hiding Lemma, [AHU18])**.** *Let* $\mathcal{X}, \mathcal{Y}$ *be arbitrary sets and let* $\mathcal{S} \subseteq \mathcal{X}$ *be a (possibly random) subset. Let* $G, H : \mathcal{X} \to \mathcal{Y}$ *be arbitrary (possibly random) functions such that* $H(x) = G(x)$*, for all* $x \notin \mathcal{S}$*. Let* z *be a classical bit string or a (possibly mixed) quantum state. (Note that* G, H, S, z *may have arbitrary joint distribution.) Let* $\mathcal{A}$

be an oracle-aided quantum algorithm that makes at most q quantum queries. Let $\mathcal{B}$ be an algorithm that on input z chooses a random query index $i \leftarrow [q]$, runs $\mathcal{A}^H(z)$, measures $\mathcal{A}$'s i-th query and outputs the measurement outcome. Then, we have

$$\left|\Pr[\mathcal{A}^G(z)=1] - \Pr[\mathcal{A}^H(z)=1]\right| \leq 2q\sqrt{\Pr[\mathcal{B}^H(z)\in\mathcal{S}]}. \tag{21}$$

Moreover, for any fixed choice of G, H, S and z (when z is a classical string or a pure state), we get

$$\left\| \left|\psi_q^H\right\rangle - \left|\psi_q^G\right\rangle \right\| \leq 2q\sqrt{\frac{1}{q}\sum_{i=0}^{q-1}\left\|\Pi_{\mathcal{S}}\left|\psi_i^H\right\rangle\right\|^2}, \tag{22}$$

where $\left|\psi_i^H\right\rangle$ denotes the intermediate state of $\mathcal{A}$ just before the $(i+1)$-st query, where the initial state at $i=0$ corresponds to z, and $\Pi_{\mathcal{S}}$ is a projector onto $\mathcal{S}$.

In particular, we will actually make use of the slightly more generalized version.

Lemma 2.8. *Let U, V be any unitaries acting on a Hilbert space $\mathcal{H}$, with the guarantee that, $U\left|\psi\right\rangle = V\left|\psi\right\rangle$, for all $\left|\psi\right\rangle \notin S$, which is a (possibly random) subspace of $\mathcal{H}$. Let z be a classical bit string or a (possibly mixed) quantum state. Let $\mathcal{A}$ be an oracle-aided quantum algorithm that makes at most q quantum queries. Then, we have*

$$\left\| \left|\psi_q^V\right\rangle - \left|\psi_q^U\right\rangle \right\| \leq 2\sum_{i=0}^{q-1}\sqrt{\left\|\Pi_{\mathcal{S}}\left|\psi_i^U\right\rangle\right\|^2}, \tag{23}$$

where $\left|\psi_i^U\right\rangle$ denotes the intermediate state of $\mathcal{A}$ just before the $(i+1)$-st query, where the initial state at $i=0$ corresponds to z, and $\Pi_{\mathcal{S}}$ is a projector onto $\mathcal{S}$.

Proof. This follows directly from gentle measurement. Let Π_S be the projector onto the subspace S. Define $\left|\psi_i^{Int,U}\right\rangle$ to be the state generated as follows:

- Run $\mathcal{A}^U$ up until just before the $(i+1)$-st query, but before every query $j \in [i]$, apply the measurement $\{\Pi_S, I-\Pi_S\}$ onto $\left|\psi_j\right\rangle$.
- If the result is the first outcome, output $\perp$.
- Otherwise, continue.

Note that since U and V agree on $\mathsf{Im}(I-S)$, $\left|\psi_i^{Int,U}\right\rangle = \left|\psi_i^{Int,V}\right\rangle$.

We know that $\left|\psi_1^U\right\rangle = \left|\psi_1^{Int,U}\right\rangle$. Gentle measurement immediately gives that $\left\|\left|\psi_2^U\right\rangle - \left|\psi_2^{Int,U}\right\rangle\right\| \leq 2\sqrt{\left\|\Pi_S\left|\psi_1^U\right\rangle\right\|^2}$. Now for our induction hypothesis, assume that, $\left\|\left|\psi_{q-1}^U\right\rangle - \left|\psi_{q-1}^{Int,U}\right\rangle\right\| \leq 2\sum_{i=0}^{q-2}\sqrt{\left\|\Pi_S\left|\psi_i^U\right\rangle\right\|^2}$.
By gentle measurement and triangle inequality

$$\begin{aligned}\left\|\left|\psi_q^U\right\rangle - \left|\psi_q^{Int,U}\right\rangle\right\| &= \left\|\left|\psi_{q-1}^U\right\rangle - \left|\psi_{q-1}^{Int,U}\right\rangle\right\| + 2\sqrt{\left\|\Pi_S\left|\psi_{q-1}^U\right\rangle\right\|^2} \\ &\leq 2\sum_{i=0}^{q-1}\sqrt{\left\|\Pi_S\left|\psi_i^U\right\rangle\right\|^2}\end{aligned}$$

A symmetric argument gives

$$\left\|\left|\psi_q^V\right\rangle - \left|\psi_q^{Int,V}\right\rangle\right\| \leq 2\sum_{i=0}^{q-1}\sqrt{\left\|\Pi_S\left|\psi_i^U\right\rangle\right\|^2}$$

and the result follows from triangle inequality. □

2.5 Haar Measure And Its Concentration

We use $\mathbb{U}(N)$ to denote the group of $N \times N$ unitary matrices, and μ_N to denote the Haar measure on $\mathbb{U}(N)$. We use $\mathbb{C}^N$ to denote the set of quantum pure states of dimension N, and μ_N^s to denote the Haar measure on $\mathbb{C}^N$. Given a metric space $(\mathcal{M}, d)$ where d denotes the metric on the set $\mathcal{M}$, a function $f : \mathcal{M} \to \mathbb{R}$ is L-lipschitz if for all $x, y \in \mathcal{M}, |f(x) - f(y)| \leq L \cdot d(x, y)$. For d, we will in particular be concerned about the Frobenius norm. For a complex matrix M, its Frobenius norm is $\sqrt{\mathrm{Tr}(M^\dagger M)}$.

The following inequality involving Lipschitz continuous functions captures the strong concentration of Haar measure.

Theorem 2.9 ([Mec19])**.** *Let $k \geq 1$ be an integer. Given $N_1, N_2, \ldots, N_k \in \mathbb{N}$, let $X = \mathbb{U}(N_1) \bigoplus \cdots \bigoplus \mathbb{U}(N_k)$ be the space of block diagonal unitary matrices with blocks of size $N_1, N_2, \ldots, N_k$. Let $\mu = \mu_{N_1} \times \cdots \times \mu_{N_k}$ be the product of Haar measures on X, where, for each $i \in [k]$, μ_{N_i} is the Haar measure on $\mathbb{U}(N_i)$. Suppose that $f : X \to \mathbb{R}$ is L-Lipshitz with respect to the Frobenius norm. Then for every $t > 0$,*

$$\Pr_{U \leftarrow \mu}[|f(U) - \mathbb{E}_{V \leftarrow \nu}[f(V)]| \geq t] \leq 2 \exp\left(-\frac{(N-2)t^2}{24L^2}\right), \tag{24}$$

where $N = \min(N_1, \ldots, N_k)$.

We will use the following lemma.

Lemma 2.10 ([Kre21])**.** *Let $A^{(\cdot)}$ be a quantum algorithm that makes T queries to a unitary oracle operating on a D-dimensional state space, and let $|\psi\rangle$ be any input to $A^{(\cdot)}$. For $U \in \mathbb{U}(D)$, define $f(U) = \Pr[A^U(|\psi\rangle) = 1]$. Then, f is $2T$-Lipshitz with respect to the Frobenius norm.*

2.6 State Tomography

Theorem 2.11 (Quantum State Tomography, Proposition 2.2 from [HKOT23])**.** *There is a pure state tomography procedure with the following behavior. Let $d \in \mathbb{N}$. For any $n > d, \epsilon > 0$, given $O(n/\epsilon)$ copies of a pure state $|\phi\rangle \in \mathbb{C}^d$, it outputs a classical description of an estimate pure state $|\phi'\rangle$ such that with probability $\geq 1 - e^{-5n}$,*

$$\left|\langle\phi|\phi'\rangle\right|^2 \geq 1 - \epsilon. \tag{25}$$

Remark 2.12. *The statement from [HKOT23] specifies an error depending on the dimension of the state. However, this stronger statement is immediate upon observing the proof, and indeed the predecessor of this result from [GKKT20] makes the dependence on n explicit.*

2.7 Quantum Complexity

We will consider promise problem versions of **BQP** and **QCMA**.

A promise problem Π is a pair of disjoint sets $\Pi_{yes}, \Pi_{no} \subseteq \{0,1\}^*$ with $\Pi_{yes} \cap \Pi_{no} = \emptyset$.

Definition 2.13. *Let $\mathcal{O}$ be an oracle. We say that a promise problem $\Pi = (\Pi_{yes}, \Pi_{no})$ is in* **BQP** *if there exists constants $c_1 > c_2$ and a randomized polynomial-time quantum oracle algorithm $\mathcal{A}^{\mathcal{O}}(x)$ such that*

1. *If $x \in \Pi_{yes}$, then $\Pr[\mathcal{A}^{\mathcal{O}}(x) = 1] \geq c_1$*

2. *If* $x \in \Pi_{no}$*, then* $\Pr[\mathcal{A}^{\mathcal{O}}(x) = 1] \leq c_2$

Definition 2.14. *Let* $\mathcal{O}$ *be an oracle. We say that a promise problem* $\Pi = (\Pi_{yes}, \Pi_{no})$ *is in* **QCMA** *if there exists constants* $c_1 > c_2$*, a randomized polynomial-time quantum oracle algorithm* $\mathsf{Ver}^{\mathcal{O}}(x)$*, and a polynomial* $p(\cdot)$ *such that*

1. *If* $x \in \Pi_{yes}$*, then there exists a witness* $w \in \{0,1\}^{p(|x|)}$ *such that*

$$\Pr[\mathsf{Ver}^{\mathcal{O}}(x, c) = 1] \geq c_1$$

2. *If* $x \in \Pi_{no}$*, then for all witnesses* $w \in \{0,1\}^{p(|x|)}$*,*

$$\Pr[\mathsf{Ver}^{\mathcal{O}}(x, w) = 1] \leq c_2$$

2.8 Quantum Cryptography Definitions

In this subsection, we provide definitions of quantum cryptographic primitives we consider in this paper.

Definition 2.15 (QCCC Key Exchange (KE))**.** *A QCCC key exchange (KE) protocol is an interactive two-party protocol between two parties* A *and* B*. Both* A *and* B *are QPT, but all communications are classical. At the end of the protocol,* $\mathsf{A} \leftrightarrows \mathsf{B}$ *produces a classical transcript* τ*,* A*'s output bit* a*, and* B*'s output bit* b*. A KE protocol must satisfy the following properties.*

- *Correctness: At the conclusion of the protocol, both parties agree on the output. Formally,*

$$\Pr[a = b : \mathsf{A}(1^n) \leftrightarrows \mathsf{B}(1^n) \to (\tau, a, b)] \geq 1 - \mathsf{negl}(n). \tag{26}$$

- *Security: An adversary with the ability to see all communication should not be able to learn the output. That is, for all non-uniform QPT adversaries* $\mathcal{E}$*,*

$$\Pr[a \leftarrow \mathcal{E}(\tau) : \mathsf{A}(1^n) \leftrightarrows \mathsf{B}(1^n) \to (\tau, a, b)] \leq \mathsf{negl}(n). \tag{27}$$

Definition 2.16 (QCCC Multiparty Non-Interactive Key Exchange (NIKE))**.** *An* N*-party QCCC multiparty non-interactive key exchange (NIKE) is a pair* $(\mathsf{Publish}, \mathsf{KeyGen})$ *of QPT algorithms with the following syntax:*

- $\mathsf{Publish}(1^n, i) \to (x_i, \rho_i)$*: It is a QPT algorithm that, on input the security parameter* 1^n *and an integer* $i \in [N]$*, outputs a classical bit string* x_i *and a quantum state* ρ_i*.*
- $\mathsf{KeyGen}(i, x_1, ..., x_N, \rho_i) \to k_i$*: It is a QPT algorithm that, on input* i*,* $(x_1, ..., x_N)$ *and* ρ_i*, outputs a classical key* k_i*.*

We require the following two properties:

- *Correctness:*

$$\Pr[\forall i, j \in [N], k_i = k_j : (x_i, \rho_i) \leftarrow \mathsf{Publish}(1^n, i), k_i \leftarrow \mathsf{KeyGen}(i, x_1, ..., x_N, \rho_i)] \geq 1 - \mathsf{negl}(n). \tag{28}$$

- *Security: For any non-uniform QPT adversary* $\mathcal{E}$*,*

$$\Pr\left[\exists i \in [N] \text{ s.t. } e = k_i : \begin{array}{l} (x_i, \rho_i) \leftarrow \mathsf{Publish}(1^n, i) \\ k_i \leftarrow \mathsf{KeyGen}(i, x_1, ..., x_N, \rho_i) \\ e \leftarrow \mathcal{E}(x_1, ..., x_N) \end{array}\right] \leq \mathsf{negl}(n). \tag{29}$$

Definition 2.17 (Quantum Lightning [Zha21])**.** *A quantum lightning protocol is a tuple* $(\mathsf{Mint}, \mathsf{Ver})$ *of uniform QPT algorithms with the following syntax.*

- $\mathsf{Mint}(1^n) \to (\sigma, |\$\rangle)$*: takes in the security parameter* 1^n *and outputs a serial number* σ *and a bolt state* $|\$\rangle$*.*
- $\mathsf{Ver}(1^n, \sigma, |\$\rangle) \to \top/\bot$*: takes in the security parameter* 1^n *as well as a serial number* σ *and a money state* $|\$\rangle$*, and outputs* $\top/\bot$*.*

We require the following properties.

- *Correctness: Honestly generated bolt states verify under their serial number. That is,*

$$\Pr[\top \leftarrow \mathsf{Ver}(1^n, \mathsf{Mint}(1^n))] \geq 1 - \mathsf{negl}(n). \tag{30}$$

- *Security: No attacker can generate two bolt states for the same serial number. That is, for all non-uniform QPT attackers* $\mathcal{A}$*,*

$$\Pr\left[\begin{array}{c} \top \leftarrow \mathsf{Ver}(1^n, \sigma, \rho_{\mathbf{A}}) \\ \top \leftarrow \mathsf{Ver}(1^n, \sigma, \rho_{\mathbf{B}}) \end{array} \;:\; (\sigma, \rho_{\mathbf{AB}}) \leftarrow \mathcal{A}(1^n) \right] \leq \mathsf{negl}(n). \tag{31}$$

  Here, $\rho_{\mathbf{AB}}$ *is a state over two registers* $\mathbf{A}$ *an* $\mathbf{B}$*, and* $\rho_{\mathbf{A}}$ *(*$\rho_{\mathbf{B}}$*) is the state on the register* $\mathbf{A}$ *(*$\mathbf{B}$*).*

Definition 2.18 (Distributional One-Way Puzzles [CGG24])**.** *A* β*-distributional one-way puzzle (OWPuzz) is a uniform QPT algorithm* $\mathsf{Samp}(1^n) \to (k, s)$ *which takes in a security parameter* 1^n *and produces two classical bit strings, a key* k *and a puzzle* s*, such that given a puzzle* s*, it is computationally infeasible to sample from the conditional distribution over keys* k*. More formally, we require that for all non-uniform QPT algorithms* $\mathcal{A}$*, for all sufficiently large* $n \in \mathbb{N}$*,*

$$\Delta((k, s), (\mathcal{A}(1^n, s), s)) \geq \beta(n), \tag{32}$$

where $(k, s) \leftarrow \mathsf{Samp}(1^n)$*.*

Remark 2.19. *Distributional OWPuzzs are equivalent to OWPuzzs [CGG24], defined in [KT24]. A OWPuzz is a QPT sampler paired with an inefficient verifier such that given a puzzle* s*, it is hard to find a key* k *which verifies. Since we will not work directly with OWPuzzs, we omit the formal definition.*

Definition 2.20 (QCCC Commitments [KT24])**.** *A QCCC commitment scheme is a two-party protocol between a QPT committer* Com *and a QPT receiver* Rec *over a classical channel consisting of a commit stage and an opening stage operating on a private input* m *described as follows.*

1. *Commit stage: Both parties receive the security parameter* 1^n*. The committer* Com *receives a private input* m*.* Com *interacts with the receiver* Rec *using only classical messages, and together they produce a transcript* z*. At the end of the stage, both parties hold a private quantum state* ρ_{Com} *and* ρ_{Rec}*, respectively.*
2. *Opening stage: Both parties receive the transcript* z *as well as their private quantum states* ρ_{Com} *and* ρ_{Rec}*, respectively. The committer* Com *interacts with the receiver* Rec *using only classical messages. At the end of the stage,* Rec *either outputs a message or the reject symbol* $\bot$*.*

We require the following two properties.

1. *Correctness: For all messages m, when* Com *and* Rec *interact honestly, the probability that* Rec *outputs m at the end of the opening stage is at least $1 - \mathsf{negl}(n)$.*

2. *(Computational) hiding: For all $m \neq m'$ and for all non-uniform QPT adversarial receivers* Rec'*, the view to* Rec' *interacting with* Com *with input m is computationally indistinguishable from the view of* Rec' *interacting with* Com *with input m'. That is, for any non-uniform QPT adversary* Rec' *that interacts with the honest* Com *in the commit stage and outputs a bit after the end of the commit stage,*

$$|\Pr[1 \leftarrow \mathsf{Rec}'_{1^n,m}] - \Pr[1 \leftarrow \mathsf{Rec}'_{1^n,m'}]| \leq \mathsf{negl}(n). \tag{33}$$

   Here, $\mathsf{Rec}'_{1^n,m}$ *is a non-uniform QPT adversary that interacts with the honest* Com *whose input is $(1^n, m)$.*

3. *(Computational weak honest) binding: For all m and for any non-uniform QPT adversarial committer* Com'*, the probability that* Com' *wins the following game is $\leq \mathsf{negl}(n)$.*

   (a) *In the commit stage, an honest receiver* Rec *interacts with the honest committer* Com *to produce a transcript z and the output states ρ_{Com} and ρ_{Rec}.*

   (b) *In the opening stage, the honest receiver* Rec *is given ρ_{Rec} and z, while* Com' *is given z (but not ρ_{Com}). They then proceed to run the opening stage with the committer replaced by* Com'*, and* Rec *produces a final output m'.* Com' *wins if $m' \neq m$ and $m' \neq \bot$.*

Remark 2.21. *Here we use a very weak security definition for QCCC commitments. This only serves to make our construction of OWPuzz from QCCC commitments stronger. For our separation, we use a stronger security notion detailed in Section 7.*

2.9 State Verification Unitaries

Definition 2.22. *For a state $|\phi\rangle$, we will define two unitaries representing "verification of the state" as*

$$\begin{aligned} R_{|\phi\rangle} |\phi\rangle &\mapsto (-1) |\phi\rangle \\ P_{|\phi\rangle} |\phi\rangle |b\rangle &\mapsto |\phi\rangle |\bar{b}\rangle \end{aligned} \tag{34}$$

and on all orthogonal inputs both maps will act as the identity. It is well known that $R_{|\phi\rangle}$ can be used to compute $P_{|\phi\rangle}$ and vice-versa.

Lemma 2.23 ([JLS18], Theorem 4)**.** *For any $\epsilon > 0$, for any $t \geq \frac{2}{\epsilon^2}$ and an efficient algorithm $\mathcal{B}$ such that for all $|\phi\rangle$,*

$$\|R_{|\phi\rangle}(\cdot)R^\dagger_{|\phi\rangle} - \mathcal{B}(|\phi\rangle^{\otimes t}, \cdot)\|_\diamond \leq \epsilon. \tag{35}$$

In particular, let Π^{t+1}_{sym} be the projection onto the symmetric subspace $Span(|\phi\rangle^{\otimes t+1})$. Then $\mathcal{B}(|\phi\rangle^{\otimes t}, |\psi\rangle)$ will apply the measurement $\{\Pi^{t+1}_{sym}, I - \Pi^{t+1}_{sym}\}$ on $|\phi\rangle^{\otimes t} |\psi\rangle$.

The following corollary follows immediately from the fact that $R_{|\phi\rangle}$ can be used to implement $P_{|\phi\rangle}$.

Lemma 2.24 (Variant of Lemma 4.1 from [AGKL24]). *Let $N, M > 0$ be integers with $N \leq M$. Let ρ be a quantum state of dimension NM defined by sampling N states $|\phi_k\rangle \leftarrow \mu_M^s$ for $k \in [N]$. That is,*

$$\rho := \mathop{\mathbb{E}}_{|\phi_k\rangle \leftarrow \mu_M^s, k\in[N]} \left[\bigotimes_{k=1}^{N} |\phi_k\rangle\langle\phi_k| \right]. \tag{36}$$

Let ρ' be the same state but where we require that each $|\phi_k\rangle, |\phi_{k'}\rangle$ are orthogonal. Formally, ρ' is the mixed state of dimension NM defined as

$$\rho' := \mathop{\mathbb{E}}_{U \leftarrow \mu_M} \left[\bigotimes_{k=1}^{N} U |k\rangle \langle k| U^\dagger \right]. \tag{37}$$

Then $\mathrm{TD}(\rho, \rho') \leq \frac{N(N+1)}{2M}$.

A formal proof of this lemma is included in Appendix A.

Corollary 2.25. *Let $t = \mathsf{poly}(n)$ be a polynomial. Let $\mathcal{A}^{(\cdot)}$ be any t-query quantum algorithm. Define*

$$p_1 := \Pr_{|\phi_k\rangle \leftarrow \mu_{2^n}^s, k\in[t]} \left[\mathcal{A}^{R_{|\phi_1\rangle}, \ldots, R_{|\phi_t\rangle}} (|\phi_1\rangle\langle\phi_1| \otimes \cdots \otimes |\phi_t\rangle\langle\phi_t|) \right], \tag{38}$$

$$p_2 := \Pr_{U \leftarrow \mu_{2^n}} \left[\mathcal{A}^{R_{U|1\rangle}, \ldots, R_{U|t\rangle}} (U|1\rangle\langle 1| U^\dagger \otimes \cdots \otimes U|t\rangle\langle t| U^\dagger) \right]. \tag{39}$$

Then $|p_1 - p_2| = \mathsf{negl}(n)$.

Proof. This follows from the fact that $R_{|\phi\rangle}$ can be simulated using $2^{n/8}$ copies of $|\phi\rangle$ up to negligible error by Lemma 2.23. Therefore, plugging in $M = 2^n$ and $N = 2^{n/8} \cdot t$, by Lemma 2.24 each query can perturb the state by at most $O(N^2/M) = \mathsf{negl}(n)$. Thus, by induction, $\mathcal{A}$'s internal state when acting on orthogonal $U|i\rangle$ is $t \cdot \mathsf{negl}(n) = \mathsf{negl}(n)$ close in trace distance to $\mathcal{A}$'s internal state when acting on random $|\phi_i\rangle$. □

2.10 Complexity-Theoretic No-Cloning

Theorem 2.26 (Complexity-Theoretic No-Cloning [AC12]). *Let $n \in \mathbb{N}$ and let $\epsilon \in (0, 1)$ such that $1/\epsilon = o(2^n)$. Let $|\psi\rangle$ be an n-qubit state chosen uniformly at random from the Haar measure. Given one copy of $|\psi\rangle$, as well as oracle access to $P_{|\psi\rangle}$, a counterfeiter C needs $\Omega(\sqrt{\epsilon} 2^{n/2})$ queries to prepare a $2n$-qubit state ρ such that a projector $V_{|\psi\rangle}^{\otimes 2}$ onto $|\psi\rangle\langle\psi|^{\otimes 2}$ accepts ρ with probability at least ϵ. Here the probability is taken over the choice of $|\psi\rangle$ and the behavior of $V_{|\psi\rangle}^{\otimes 2}$ and C.*

3 Quantum Presampling Lemmas

In this subsection, we outline a technique for proving security of indistinguishability style games relative to fixed random functions. In particular, we adapt the postselected random oracle model from [GLLZ21]. This model will operate on functions over arbitrary domains, and so for the following definitions let $N, M \in \mathbb{N}$ be arbitrary, and let $\mathcal{F} := \{f : [N] \to [M]\}$.

Definition 3.1 (Postselected random oracle model [GLLZ21]). *Let $\mathcal{O}_0^f$ and $\mathcal{O}_1^f$ be two (potentially stateful) oracle algorithms querying f. We say that $\mathcal{O}_0^f, \mathcal{O}_1^f$ are (s,t,ϵ)-indistinguishable in the postselected random oracle model if the following holds:*

Let $\mathcal{A}^f = (\mathcal{A}_s^f, \mathcal{A}_t^f)$ be any pair of (possibly inefficient) quantum oracle algorithms making at most s,t queries respectively. For $b \in \{0,1\}$, define the game $G_b(\mathcal{A})$ as follows.

1. *Sample a random function f using the following process.*
   (a) *Sample f uniformly at random from $\mathcal{F}$.*
   (b) *Run $\mathcal{A}_s^f \to b$.*
   (c) *If $b = 0$, restart this process from (a).*
2. *Run $\mathcal{A}_t^{\mathcal{O}_b^f} \to b'$. The output of the game is b'.*

Then

$$\max_{\mathcal{A}} |\Pr[G_0(\mathcal{A}) \to 1] - \Pr[G_1(\mathcal{A}) \to 1]| \leq \epsilon. \tag{40}$$

Theorem 3.2 (Theorem 1 from [GLLZ21]). *Let $\mathcal{O}_1^f, \mathcal{O}_2^f$ be two oracle algorithms making at most $t_{\mathcal{O}}$ queries to $f \in \mathcal{F}$. Let $S, T \in \mathbb{N}$. Let $\epsilon, \gamma \in (0,1)$. If $\mathcal{O}_1^f, \mathcal{O}_2^f$ are $((S + \log \gamma^{-1}) \cdot t \cdot t_{\mathcal{O}}, t, (\epsilon - \gamma)/2)$-indistinguishable in the postselected random oracle model, then for all inefficient maps ADV from functions to strings of length S, for all T-query quantum oracle algorithms $\mathcal{A}^{(\cdot)}$,*

$$\left| \Pr_{f \leftarrow \mathcal{F}}[\mathcal{A}^{\mathcal{O}_1^f}(ADV(f)) \to 1] - \Pr_{f \leftarrow \mathcal{F}}[\mathcal{A}^{\mathcal{O}_2^f}(ADV(f)) \to 1] \right| \leq \epsilon. \tag{41}$$

We adapt this technique to give the main tool we will use for arguing indistinguishability relative to a fixed random oracle.

Corollary 3.3. *Let $\mathcal{O}_1^f, \mathcal{O}_2^f$ be two oracle algorithms making at most $t_{\mathcal{O}}$ queries to f. Let $S, T \in \mathbb{N}$. Let $\epsilon, \gamma \in (0,1)$. If $\mathcal{O}_1^f, \mathcal{O}_2^f$ are $((S + \log \gamma^{-1}) \cdot t \cdot t_{\mathcal{O}}, t, (\epsilon^2 - \gamma)/2)$-indistinguishable in the postselected random oracle model, then for a random function f, with probability $\geq 1 - \epsilon$ over $f \leftarrow \mathcal{F}$, for all T-query quantum oracle algorithms $\mathcal{A}^{(\cdot)}$ and advice strings c such that $|\mathcal{A}| + |c| + 1 \leq S$,*

$$\left| \Pr[\mathcal{A}^{\mathcal{O}_1^f}(c) \to 1] - \Pr[\mathcal{A}^{\mathcal{O}_2^f}(c) \to 1] \right| \leq \epsilon. \tag{42}$$

Proof. Assume towards contradiction that for a random function f, with probability $> \epsilon$ over f, there exists an adversary $\mathcal{A}_f$ and an advice string c_f such that $|\mathcal{A}_f| + |c_f| \leq S$ and

$$\left| \Pr[\mathcal{A}_f^{\mathcal{O}_1^f}(c_f) \to 1] - \Pr[\mathcal{A}_f^{\mathcal{O}_2^f}(c_f) \to 1] \right| > \epsilon. \tag{43}$$

We will call the set of such f: *GOOD*.

We will define b_f to be 1 if and only if

$$\Pr[\mathcal{A}_f^{\mathcal{O}_1^f}(c_f) \to 1] > \Pr[\mathcal{A}_f^{\mathcal{O}_2^f}(c_f) \to 1]. \tag{44}$$

In particular,

$$\Pr[\mathcal{A}_f^{\mathcal{O}_1^f}(c_f) \to b_f] - \Pr[\mathcal{A}_f^{\mathcal{O}_2^f}(c_f) \to b_f] > \epsilon$$

We can define an advice function $ADV(f) = \mathcal{A}_f.c_f.b_f$ for $f \in GOOD$ and $ADV(f) = \perp$ for $f \notin GOOD$. We will define $\mathcal{B}(ADV(f)) = \mathcal{B}(\mathcal{A}_f, c_f, b_f)$ to output $\mathcal{A}_f(c_f) \oplus b_f \oplus 1$. We will define $\mathcal{B}(\perp) = 0$. We have that for all $f \in GOOD$,

$$\begin{aligned}\Pr[\mathcal{B}^{\mathcal{O}_1^f}(ADV(f)) \to 1] &= \Pr[\mathcal{A}_f^{\mathcal{O}_1^f}(c_f) \to b_f] \\ \Pr[\mathcal{B}^{\mathcal{O}_2^f}(ADV(f)) \to b_f \oplus 1] &= \Pr[\mathcal{A}_f^{\mathcal{O}_2^f}(c_f) \to b_f].\end{aligned} \tag{45}$$

We can then compute

$$\begin{aligned}&\Pr_f[\mathcal{B}^{\mathcal{O}_1^f}(ADV(f)) \to 1] - \Pr_f[\mathcal{B}_f^{\mathcal{O}_2^f}(ADV(f)) \to 1] \\ &= \Pr_f[f \in GOOD] \cdot \left(\Pr_f[\mathcal{B}^{\mathcal{O}_1^f}(ADV(f)) \to 1 | f \in GOOD] - \Pr[\mathcal{B}_f^{\mathcal{O}_2^f}(ADV(f)) \to 1 | f \in GOOD]\right) \\ &+ \Pr_f[f \notin GOOD] \cdot \left(\Pr_f[\mathcal{B}^{\mathcal{O}_1^f}(ADV(f)) \to 1 | f \notin GOOD] - \Pr[\mathcal{B}_f^{\mathcal{O}_2^f}(ADV(f)) \to 1 | f \notin GOOD]\right) \\ &\geq \epsilon \cdot \left(\Pr_f[\mathcal{A}^{\mathcal{O}_1^f}(c) \to b_f | f \in GOOD] - \Pr[\mathcal{A}_f^{\mathcal{O}_2^f}(c) \to b_f | f \in GOOD]\right) + 0 \\ &> \epsilon^2,\end{aligned} \tag{46}$$

which contradicts Theorem 3.2 and so we are done. □

4 Separating Unitary Oracle

In this section, we introduce our oracle, and show a key lemma, Lemma 4.1, that will be used later.

4.1 Definition of Oracle

Our oracle will be parameterized by a sequence of unitaries $\{U_m\}_{m\in\mathbb{N}}$ and a sequence of random oracles $\{\mathsf{RO}_\ell\}_{\ell\in\mathbb{N}}$, where $\mathsf{RO}_\ell : \{0,1\}^\ell \times \{0,1\}^\ell \to \{0,1\}^\ell$. Each U_ℓ will be sampled from the Haar distribution over ℓ-qubit unitaries and each RO_ℓ will be uniformly sampled from the set of functions $\{0,1\}^\ell \times \{0,1\}^\ell \to \{0,1\}^\ell$. Our oracle consists of two sequences of unitary oracles, $SG := \{SG_\ell\}_{\ell\in\mathbb{N}}$ and $Mix := \{Mix_\ell\}_{\ell\in\mathbb{N}}$, and the classical oracle $PSPACE$:

1. SG_ℓ: It is the unitary which swaps $|0\rangle$ [18] and $\frac{1}{\sqrt{2^\ell}}\sum_{x\in\{0,1\}^\ell\setminus\{0\}}|x\rangle|\phi_x\rangle$, where $|\phi_x\rangle := U_\ell|x\rangle$, and acts as identity everywhere else. That is,

$$SG_\ell := |0\rangle\langle\psi_m| + |\psi_\ell\rangle\langle 0| + (I - |0\rangle\langle 0| - |\psi_\ell\rangle\langle\psi_\ell|), \tag{47}$$

   where $|\psi_\ell\rangle := \frac{1}{\sqrt{2^\ell}}\sum_{x\in\{0,1\}^\ell\setminus\{0\}}|x\rangle|\phi_x\rangle$.

2. Mix_ℓ: It is the following unitary.

$$\sum_{x,y\in\{0,1\}^\ell} |x\rangle\langle x| \otimes |y\rangle\langle y| \otimes (|\phi_x\rangle\langle\phi_x| + |\phi_y\rangle\langle\phi_y|) \otimes X^{\mathsf{RO}(x,y)} \otimes X \tag{48}$$

$$+ \sum_{x,y\in\{0,1\}^\ell} |x\rangle\langle x| \otimes |y\rangle\langle y| \otimes (I - |\phi_x\rangle\langle\phi_x| - |\phi_y\rangle\langle\phi_y|) \otimes I \otimes I. \tag{49}$$

[18] $|0\rangle$ is a shorthand for $|0...0\rangle$ with the appropriate length.

Here, X is the Pauli X operator and $X^r := \bigotimes_{i=1}^n X_i^{r_i}$ for any n-bit string r. In other words, Mix_ℓ performs the following operation:

$$|x\rangle |y\rangle |\phi_x\rangle |z\rangle |b\rangle \to |x\rangle |y\rangle |\phi_x\rangle |z \oplus \mathsf{RO}(x,y)\rangle |\bar{b}\rangle \tag{50}$$
$$|x\rangle |y\rangle |\phi_y\rangle |z\rangle |b\rangle \to |x\rangle |y\rangle |\phi_y\rangle |z \oplus \mathsf{RO}(x,y)\rangle |\bar{b}\rangle \tag{51}$$

and acts as identity for other orthogonal states.

3. $PSPACE$: an oracle for a **PSPACE**-complete problem.[19]

4.2 Key Lemma

We will define the following oracles $SG_{\text{Samp}} := \{SG_{\text{Samp},\ell}\}_{\ell\in\mathbb{N}}$ and $Ver_{\text{Samp}} := \{Ver_{\text{Samp},\ell}\}_{\ell\in\mathbb{N}}$ which we will use to simulate SG and Mix. These oracles will be further parameterized by some function t, which will represent the maximum number of times these oracles can be queried.

Initialization:

- For each $\ell \in \mathbb{N}$, sample an ℓ-qubit Haar random unitary $\widetilde{U}_\ell$. (Note that $\widetilde{U}_\ell$ are sampled independently from U_ℓ).
- For each $\ell \in \mathbb{N}$, sample $\widetilde{\mathsf{RO}_\ell} : \{0,1\}^\ell \times \{0,1\}^\ell \to \{0,1\}^\ell$. (Note that $\widetilde{\mathsf{RO}_\ell}$ are sampled independently from RO_ℓ).
- Sample $k_1, \dots k_t \leftarrow \{0,1\}^\ell \setminus \{0\}$, where t is some sufficiently large polynomial in ℓ, and set $\widetilde{|\phi_{k_1}\rangle}, ..., \widetilde{|\phi_{k_t}\rangle}$, where $\widetilde{|\phi_{k_i}\rangle} := \widetilde{U}_\ell |k_i\rangle$.

1. $SG_{\text{Samp},\ell}$:
   - On the i^{th} query for $i \le t$, output $|k_i\rangle \widetilde{|\phi_{k_i}\rangle}$.
   - On the i^{th} query for $i > t$, output $|\perp\rangle$.
2. $Ver_{\text{Samp},\ell}$:
   - For all $i \in [t]$ perform the mapping $|k_i\rangle \widetilde{|\phi_{k_i}\rangle} |b\rangle \to |k_i\rangle \widetilde{|\phi_{k_i}\rangle} |\bar{b}\rangle$ and acts as identity on all other orthogonal states.

Next, we will describe the oracle algorithms $\text{Sim}_{Mix}^{Ver_{\text{Samp}}} := \{\text{Sim}_{Mix_\ell}^{Ver_{\text{Samp},\ell}}\}_{\ell\in\mathbb{N}}$ and $\text{Sim}_{SG}^{SG_{\text{Samp}}} := \{\text{Sim}_{SG_\ell}^{SG_{\text{Samp},\ell}}\}_{\ell\in\mathbb{N}}$. These algorithms will be parameterized by some function t', which will be specified as required.

$\mathbf{Sim}_{Mix_\ell}^{Ver_{\mathbf{Samp},\ell}}$

1. Initialize 3 ancilla registers $|0\rangle_{\mathbf{F}} |0\rangle_{\mathbf{G}} |0\rangle_{\mathbf{H}}$.
2. On input $|x\rangle_{\mathbf{A}} |y\rangle_{\mathbf{B}} |\psi\rangle_{\mathbf{C}} |z\rangle_{\mathbf{D}} |b\rangle_{\mathbf{E}}$, do the following:
   - Query $Ver_{\text{Samp},\ell}$ on input $|x\rangle_{\mathbf{A}} |\psi\rangle_{\mathbf{C}} |0\rangle_{\mathbf{F}}$.
   - Query $Ver_{\text{Samp},\ell}$ on input $|y\rangle_{\mathbf{B}} |\psi\rangle_{\mathbf{C}} |0\rangle_{\mathbf{G}}$.

[19] For the oracle, we use the normal font, and for the complexity class, we use the bold font.

- Compute the OR of the contents of registers $\mathbf{F}$ and $\mathbf{G}$, and XOR the result into the register $\mathbf{H}$.

3. Controlled on the register $\mathbf{H}$, add $\widetilde{\mathsf{RO}_\ell}(x,y)$ to the register $\mathbf{D}$, i.e., $|x\rangle_{\mathbf{A}}|y\rangle_{\mathbf{B}}|z\rangle_{\mathbf{D}} \to |x\rangle_{\mathbf{A}}|y\rangle_{\mathbf{B}}|z \oplus \widetilde{\mathsf{RO}_\ell}(x,y)\rangle_{\mathbf{D}}$.
4. Controlled on the register $\mathbf{H}$, flip the bit of the register $\mathbf{E}$.
5. Uncompute ancilla registers $\mathbf{F}, \mathbf{G}, \mathbf{H}$ by querying $Ver_{\mathrm{Samp},\ell}$ again.

$\mathbf{Sim}_{SG_\ell}^{SG_{\mathbf{Samp},\ell}}$

1. Sample $\eta \leftarrow Bin(t,1/2)$ where $Bin(t,1/2)$ is the binomial distribution. That is,
$$\Pr[Bin(t,1/2)=\eta] = \frac{\binom{t}{\eta}}{2^t}. \tag{52}$$
2. Query $SG_{\mathrm{Samp},\ell}$ a total of η times, receiving output states $|k_1\rangle\widetilde{|\phi_{k_1}\rangle}\ldots|k_\eta\rangle\widetilde{|\phi_{k_\eta}\rangle}$.
3. For $i>\eta$, set $k_i=0$. For ease of notation, we will define $\widetilde{|\phi_0\rangle}=|0\rangle$.
4. Generate the state
$$\left(\sum_{\pi\in Sym(t)}\bigotimes_{i=1}^{t}|k_{\pi(i)}\rangle_{\mathbf{A_i}}|0\rangle_{\mathbf{B_i}}\right)\otimes\left(\bigotimes_{j=1}^{t}|k_j\rangle_{\mathbf{C_j}}\widetilde{|\phi_{k_j}\rangle}_{\mathbf{D_j}}\right). \tag{53}$$
5. For all i,j, apply $SWAP_{\mathbf{B_i},\mathbf{D_j}}$ controlled on registers $\mathbf{A_i},\mathbf{C_j}$ containing the same value in the standard basis. This results in the state
$$\left(\sum_{\pi\in Sym(t)}\bigotimes_{i=1}^{t}|k_{\pi(i)}\rangle_{\mathbf{A_i}}\widetilde{|\phi_{k_{\pi(i)}}\rangle}_{\mathbf{B_i}}\right)\otimes\left(\bigotimes_{j=1}^{t}|k_j\rangle_{\mathbf{C_j}}|0\rangle_{\mathbf{D_j}}\right). \tag{54}$$
6. Next, trace out registers $\mathbf{C_1D_1}\ldots\mathbf{C_tD_t}$. This will contain the state
$$\sum_{\pi\in Sym(t)}\bigotimes_{i=1}^{t}|k_{\pi(i)}\rangle_{\mathbf{A_i}}\widetilde{|\phi_{k_{\pi(i)}}\rangle}_{\mathbf{B_i}}. \tag{55}$$
7. Finally, let $\mathbf{In}$ be the input register. Apply the measurement $\{\Pi_{sym}^{t'+1}, I-\Pi_{sym}^{t'+1}\}$ on $\mathbf{In},\mathbf{A_1B_1},\ldots,\mathbf{A_tB_t}$, where $\Pi_{sym}^{t'+1}$ is projection onto the symmetric subspace.

Lemma 4.1. *Let n be the security parameter, and let $S(n),T(n)$ be any functions such that $T\le 2^{n/10}$. With probability at least $1-O(S\cdot 2^{-n/4})$ over U_n and RO_n, for all T query algorithms $\mathcal{A}(1^n,c)$ with $|\mathcal{A}|+|c|\le S$, where c is the advice string, for all ϵ, there exist simulators $Sim_{Mix_n}^{Ver_{Samp,n}}, Sim_{SG_n}^{SG_{Samp,n}}$ running in time $O(n,1/\epsilon)$ such that*
$$\left|\Pr[\mathcal{A}^{Mix_n,SG_n}(1^n,c)=1]-\Pr[\mathcal{A}^{Sim_{Mix_n}^{Ver_{Samp,n}},Sim_{SG_n}^{SG_{Samp,n}}}(1^n,c)=1]\right|\le\epsilon+O\left(\frac{T^{2.5}+S}{2^{n/8}}\right). \tag{56}$$

If both S and T are polynomials, we obtain the following corollary.

Corollary 4.2. *Let n be the security parameter. Let $S(n), T(n)$ be two polynomials. With probability at least $1 - \mathsf{negl}(n)$ over U_n and RO_n, for all T query algorithms $\mathcal{A}(1^n, c)$ with $|\mathcal{A}| + |c| \leq S$, where c is the advice string, for all polynomials $1/\epsilon$, there exist simulators $Sim_{Mix_n}^{Ver_{Samp,n}}, Sim_{SG_n}^{SG_{Samp,n}}$ running in time $\mathsf{poly}(n, 1/\epsilon)$ such that for all sufficiently large n,*

$$\left| \Pr[\mathcal{A}^{Mix_n, SG_n}(1^n, c) = 1] - \Pr[\mathcal{A}^{Sim_{Mix_n}^{Ver_{Samp,n}}, Sim_{SG_n}^{SG_{Samp,n}}}(1^n, c) = 1] \right| \leq \epsilon. \tag{57}$$

Before jumping into the proof of Lemma 4.1, we will formally state an important lemma

Lemma 4.3. *There exists an efficient oracle algorithm $\mathrm{Sim}^{(\cdot)}(1^t)$ such that the following is true: Let n be any natural number. Let $\{|\phi_k\rangle\}_{k \in \{0,1\}^n}$ be any collection of states. For a function $f : \{0,1\}^n \to \{0,1\}^n = [2^n]$, define the state*

$$|\phi^f\rangle := \frac{1}{\sqrt{2}} |0\rangle - \frac{1}{\sqrt{2}} \frac{1}{\sqrt{2^n - 1}} \sum_{k \in \{0,1\}^n \setminus \{0\}} \omega_{2^n}^{f(k)} |k\rangle |\phi_k\rangle . \tag{58}$$

Let $Gen^{\{|\phi_k\rangle\}_k}$ be the CPTP oracle which acts as follows

1. *Sample $k \leftarrow \{0,1\}^n \setminus \{0\}$.*
2. *Output $|k\rangle |\phi_k\rangle$.*

Then for all t, we have

$$\mathop{\mathbb{E}}_f \left[|\phi^f\rangle\langle\phi^f|^{\otimes t} \right] = \mathrm{Sim}^{Gen^{\{|\phi_k\rangle\}_k}}(1^t). \tag{59}$$

Lemma 4.3 is proven and explained in more detail in Appendix .1.

We then proceed to our proof of Lemma 4.1.

Proof. We will let $Ver_{\mathrm{Samp},n}, SG_{\mathrm{Samp},n}, \mathrm{Sim}_{Mix,n}^{Ver_{\mathrm{Samp},n}}, \mathrm{Sim}_{SG,n}^{SG_{\mathrm{Samp},n}}$ be as in the previous section with parameters $t = T$ and $t' = \frac{c_\epsilon}{\epsilon}$ for some sufficiently large c_ϵ.

We will establish the proof of Lemma 4.1 through the following sequence of hybrids:

Hybrid 0: $\mathcal{A}^{Mix_n, SG_n}(1^n, c)$.
Hybrid 1: Same as **Hybrid 0**, but all calls to U_n in Mix_n and SG_n are replaced with a freshly sampled Haar random unitary $\widetilde{U}_n$ over n qubits. Define $\widetilde{SG}_n$ and $\widetilde{Mix}_n$ to be the same as SG_n and Mix_n, except that queries to U_n are replaced by queries to $\widetilde{U}_n$.

Claim 4.4. *With probability $\geq 1 - 2^S \exp\left(-2^{n/2}\right)$ over U_n, and with probability 1 over RO_n, for all sufficiently large n,*

$$\max_{\mathcal{A},c} |\Pr[1 \leftarrow \textbf{Hybrid 1}(1^n)] - \Pr[1 \leftarrow \textbf{Hybrid 0}(1^n)]| \leq \frac{1}{2^{n/8}}. \tag{60}$$

Proof. Let $f(U_n) = \Pr[1 \leftarrow \textbf{Hybrid 0}(1^n)]$. Then, Lemma 2.10 implies that f is $2T$ Lipshitz. Now observe that $\Pr[1 \leftarrow \textbf{Hybrid 1}(1^n)] = \mathbb{E}_{U_n} \Pr[1 \leftarrow \textbf{Hybrid 1}(1^n)]$.

Invoking the strong concentration of the Haar measure in Theorem 2.9 with $t = \delta$ and $L = 2T$ gives us

$$\Pr_{U_n} \left[\left| \Pr_{\widetilde{U}_n}[1 \leftarrow \textbf{Hybrid 1}(1^n)] - \Pr[1 \leftarrow \textbf{Hybrid 0}(1^n)] \right| \geq \delta \right] \tag{61}$$

$$\leq 2 \exp\left(-\frac{(2^n - 2)\delta^2}{96T^2} \right) \leq \exp\left(-2^{n/2}\right). \tag{62}$$

for all sufficiently large n. Setting $\delta = 2^{-n/8}$ and taking the union bound over all adversaries $\mathcal{A}$ and all c gives us that, with probability at least $1 - 2^S \cdot \exp\left(-2^{n/2}\right)$ over the choice of U_n and with probability 1 over the choice of RO_n, for all sufficiently large n

$$\max_{\mathcal{A},c} |\Pr[1 \leftarrow \textbf{Hybrid } \mathbf{1}(1^n)] - \Pr[1 \leftarrow \textbf{Hybrid } \mathbf{0}(1^n)]| \leq \frac{1}{2^{n/8}} + \exp\left(-2^{(n/2)+1}\right). \quad (63)$$

□

Hybrid 2: Same as **Hybrid 1**, except that the oracle $\widetilde{SG}_n$ is replaced with the following oracle SG'_n:

- SG'_n : Let $\widetilde{|\phi_x\rangle} = \widetilde{U}_n |x\rangle$. Then swap $|0\rangle$ and $\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} \omega_{2^n}^{\mathsf{RO}_n^*(x)} |x\rangle \widetilde{|\phi_x\rangle}$, and act as identity everywhere else. Here, RO_n^* is a freshly sampled random oracle that's independent of RO_n. [20]

Claim 4.5. *For all n and for all U_n, RO_n,*

$$\Pr[1 \leftarrow \textbf{Hybrid } \mathbf{1}(1^n)] = \Pr[1 \leftarrow \textbf{Hybrid } \mathbf{2}(1^n)]. \quad (64)$$

Proof. We observe that **Hybrid** $-$ **2** is actually identical to **Hybrid** $-$ **1**, except instead of sampling $\widetilde{U}_n$ as a Haar random unitary over n qubits, we instead sample U'_n as a Haar random unitary over n qubits and set $\widetilde{U}_n = U'_n \cdot S^{\mathsf{RO}^*}$ where S^{RO^*} is the unitary which maps $|x\rangle \to \omega_{2^n}^{\mathsf{RO}^*(x)} |x\rangle$. In particular, since the behavior $\widetilde{Mix}_n$ is independent of the phase of $\widetilde{U} |x\rangle$ for all x, $\widetilde{Mix}_n$ acts the same when instantiated with either $\widetilde{U}_n$ or U'_n. □

Hybrid 3: Same as **Hybrid 2:**, except SG'_n is simulated by the simulator $\mathrm{Sim}_{SG_n}^{SG_{\mathrm{Samp},n}}$.

Claim 4.6. *For all n, with probability 1 over U_n and RO_n,*

$$|\Pr[1 \leftarrow \textbf{Hybrid } \mathbf{3}(1^n)] - \Pr[1 \leftarrow \textbf{Hybrid } \mathbf{2}(1^n)]| \leq \epsilon. \quad (65)$$

Proof. First, observe that swapping between $|0\rangle$ and $\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n \setminus \{0\}} \omega_{2^n}^{\mathsf{RO}_n^*(x)} |x\rangle \widetilde{|\phi_x\rangle}$ is the same as applying $R_{|\phi-\rangle}$, where $|\phi-\rangle$ is $|0\rangle - \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n \setminus \{0\}} \omega_{2^n}^{\mathsf{RO}_n^*(x)} |x\rangle \widetilde{|\phi_x\rangle}$. Let $\mathcal{B}$ be the algorithm from Lemma 2.23. We know that

$$\|R_{|\phi-\rangle}(\cdot)R^\dagger_{|\phi-\rangle} - \mathcal{B}(|\phi-\rangle^{\otimes t'}, \cdot)\|_\diamond \leq O(1/t'), \quad (66)$$

where $t = \mathsf{poly}(n)$. Now, from Lemma 4.3, we can conclude that the final state prepared by $\mathrm{Sim}_{SG_n}^{SG_{\mathrm{Samp},n}}$, i.e.,

$$\left(\sum_{\pi \in Sym(t)} \bigotimes_{i=1}^{t} |k_{\pi(i)}\rangle_{\mathbf{A_i}} \widetilde{|\phi_{k_{\pi(i)}}\rangle}_{\mathbf{B_i}}\right)\left(\sum_{\pi \in Sym(t)} \bigotimes_{i=1}^{t} \langle k_{\pi(i)}|_{\mathbf{A_i}} \widetilde{\langle \phi_{k_{\pi(i)}}|}_{\mathbf{B_i}}\right) \quad (67)$$

is identical as a mixed state to $|\phi-\rangle\langle\phi-|^{\otimes t}$. By the construction of $\mathrm{Sim}_{SG,n}^{SG_{\mathrm{Samp},n}}$,

$$\|SG'_n - \mathrm{Sim}_{SG_n}^{SG_{\mathrm{Samp},n}}\|_\diamond \leq O(1/t'). \quad (68)$$

Therefore, by setting t' sufficiently large and applying Theorem 2.1, the claim follows. □

[20] Here ω_{2^n} is the 2^n-th root of unity. Moreover, we interpret $\mathsf{RO}_n(x) \in \{0,1\}^n$ as an element of $[2^n]$.

Hybrid 4: Same as **Hybrid 3** except $\widetilde{Mix}_n$ is replaced with the simulator $\mathrm{Sim}'^{Ver_{\mathrm{Samp},n}}_{Mix_n}$, which is exactly the same as $\mathrm{Sim}^{Ver_{\mathrm{Samp},n}}_{Mix_n}$, except that controlled on register $\mathbf{H}$, $\mathsf{RO}_n(x,y)$ is added to the register $\mathbf{D}$ instead of $\widetilde{\mathsf{RO}}_n(x,y)$, i.e., $|x\rangle_{\mathbf{A}}|y\rangle_{\mathbf{B}}|z\rangle_{\mathbf{D}} \to |x\rangle_{\mathbf{A}}|y\rangle_{\mathbf{B}}|z\oplus\mathsf{RO}_n(x,y)\rangle_{\mathbf{D}}$. $\mathrm{Sim}'^{Ver_{\mathrm{Samp},n}}_{Mix_n}$ is precisely defined as follows:

1. Initialize 3 ancilla registers $|0\rangle_{\mathbf{F}}|0\rangle_{\mathbf{G}}|0\rangle_{\mathbf{H}}$.
2. On input $|x\rangle_{\mathbf{A}}|y\rangle_{\mathbf{B}}|\psi\rangle_{\mathbf{C}}|z\rangle_{\mathbf{D}}|b\rangle_{\mathbf{E}}$, do the following:
   - Query $Ver_{\mathrm{Samp},n}$ on input $|x\rangle_{\mathbf{A}}|\psi\rangle_{\mathbf{C}}|0\rangle_{\mathbf{F}}$.
   - Query $Ver_{\mathrm{Samp},n}$ on input $|y\rangle_{\mathbf{B}}|\psi\rangle_{\mathbf{C}}|0\rangle_{\mathbf{G}}$.
   - Compute the OR of the contents of registers $\mathbf{F}$ and $\mathbf{G}$, and XOR the result into the register $\mathbf{H}$.
3. Controlled on the register $\mathbf{H}$, add $\mathsf{RO}_n(x,y)$ to the register $\mathbf{D}$, i.e., $|x\rangle_{\mathbf{A}}|y\rangle_{\mathbf{B}}|z\rangle_{\mathbf{D}} \to |x\rangle_{\mathbf{A}}|y\rangle_{\mathbf{B}}|z\oplus\mathsf{RO}_n(x,y)\rangle_{\mathbf{D}}$.
4. Controlled on the register $\mathbf{H}$, flip the bit of the register $\mathbf{E}$.
5. Uncompute ancilla registers $\mathbf{F},\mathbf{G},\mathbf{H}$ by querying $Ver_{\mathrm{Samp},n}$ again.

Claim 4.7. *With probability* $\geq 1-\exp(-2^n)$ *over* U_n, RO_n,

$$\max_{\mathcal{A},c} |\Pr[1 \leftarrow \textbf{Hybrid 3}(1^n)] - \Pr[1 \leftarrow \textbf{Hybrid 4}(1^n)]| \leq O\left(\frac{T^2}{2^n}\right). \tag{69}$$

Proof. Let $\{k_1,\dots k_t\}$ be the set of all inputs to $SG_{\mathrm{Samp,n}}$ chosen during **Initialization**. Since $Ver_{\mathrm{Samp},n}$ only performs the mapping $|k_i\rangle\,\widetilde{|\phi_{k_i}\rangle}\,|b\rangle \to |k_i\rangle\,\widetilde{|\phi_{k_i}\rangle}\,|\bar{b}\rangle$ for $k_i \in \{k_1,\dots k_t\}$, by construction of $\mathrm{Sim}'^{Ver_{\mathrm{Samp},n}}_{Mix_n}$, the oracles $\widetilde{Mix}_n$ and $\mathrm{Sim}'^{Ver_{\mathrm{Samp},n}}_{Mix_n}$ are identical on all inputs of the following form:

- $|x\rangle\,|y\rangle\,\widetilde{|\phi_x\rangle}\,|z\rangle\,|b\rangle\,, \forall x \in \{k_1,\dots k_t\}$.
- $|x\rangle\,|y\rangle\,\widetilde{|\phi_y\rangle}\,|z\rangle\,|b\rangle\,, \forall y \in \{k_1,\dots k_t\}$.

which implies that $\widetilde{Mix}_n$ and $\mathrm{Sim}'^{Ver_{\mathrm{Samp},n}}_{Mix_n}$ only differ on all inputs of the form $|x\rangle\,|y\rangle\,\widetilde{|\phi_x\rangle}\,|z\rangle\,|b\rangle\,, \forall x \notin \{k_1,\dots k_t\}$ and $|x\rangle\,|y\rangle\,\widetilde{|\phi_y\rangle}\,|z\rangle\,|b\rangle\,, \forall y \notin \{k_1,\dots k_t\}$. Let the subspace spanned by all these differing inputs be S. Invoking Lemma 2.8, we can conclude that

$$\left|\Pr[\mathcal{A}^{\widetilde{Mix}_n}=1] - \Pr[\mathcal{A}^{\mathrm{Sim}'^{Ver_{\mathrm{Samp},n}}_{Mix_n}}=1]\right| \leq 2\sum_{i=0}^{T-1}\sqrt{||\Pi_{\mathcal{S}}\,|\psi_i^{\widetilde{U_n}}\rangle\,||^2}$$

where Π_S is the projector onto S and $|\psi_i^{\widetilde{U_n}}\rangle$ is as in Lemma 2.8. Since $\widetilde{\mathsf{RO}}$ is queryable only through $\widetilde{Mix}_n$, for each i, $||\Pi_{\mathcal{S}}\,|\psi_i^{\widetilde{\mathsf{RO}_n}}\rangle\,||$ is at most the probability that the n qubit state $\widetilde{|\phi_k\rangle}$ for $k \notin \{k_1,\dots k_t\}$ is cloned by $\mathcal{A}$. From Theorem 2.26, we know that for any adversary, this probability is upper bounded by $O\left(\frac{T^2}{2^n}\right)$. Therefore,

$$\left|\Pr[\mathcal{A}^{\widetilde{Mix}_n}=1] - \Pr[\mathcal{A}^{\mathrm{Sim}'^{Ver_{\mathrm{Samp},n}}_{Mix_n}}=1]\right| \leq 2\sqrt{T}\cdot O\left(\frac{T^2}{2^n}\right).$$

□

Thus, so far we have established that, with probability $\geq 1 - 2^S \exp(-2^n)$ over U_n, RO_n,

$$\max_{\mathcal{A},c} |\Pr[1 \leftarrow \textbf{Hybrid } \mathbf{4}(1^n)] - \Pr[1 \leftarrow \textbf{Hybrid } \mathbf{0}(1^n)]| \leq \epsilon + O\left(\frac{T^{2.5}}{2^{n/8}}\right). \tag{70}$$

Now we will introduce additional hybrids to switch to the post selected random oracle model in order to complete the proof.

Hybrid 5: In this hybrid, we move to the post selected random oracle model (Definition 3.1) to remove the advice c. In particular, we will let s, t be functions to be set later. In **Hybrid 5** the adversary $\mathcal{A}$ is modelled as $(\mathcal{A}_s, \mathcal{A}_t)$, and has access to the oracle $(\mathrm{Sim}'^{Ver_{\mathrm{Samp}}}_{Mix}, \mathrm{Sim}^{SG_{\mathrm{Samp}}}_{SG})$ as defined above. Here $\mathcal{A}_s$ makes at most s queries to its oracle and $\mathcal{A}_t$ makes at most t queries to its oracle. Formally, **Hybrid 5** is defined as follows:

1. Sample RO_n as follows:
   (a) Sample RO_n uniformly at random.
   (b) Run $\mathcal{A}_s^{\mathsf{RO}_n} \to b$.
   (c) If $b = 0$, restart this process from item (a).
2. Run $\mathcal{A}_t^{\left(\mathrm{Sim}'^{Ver^{\mathsf{RO}_n}_{\mathrm{Samp}}}_{Mix}, \mathrm{Sim}^{SG^{\mathsf{RO}_n}_{\mathrm{Samp}}}_{SG}\right)} \to b'$.

Hybrid 6: This is the same as **Hybrid 5**, but RO is modeled as a compressed oracle with database register $|D\rangle$.

Claim 4.8.

$$\Pr[1 \leftarrow \textbf{Hybrid } \mathbf{6}(1^n)] = \Pr[1 \leftarrow \textbf{Hybrid } \mathbf{5}(1^n)]. \tag{71}$$

Proof. This follows directly from Lemma 2.5. □

Hybrid 7: This is the same as **Hybrid 6**, but $(\mathcal{A}_s, \mathcal{A}_t)$, has access to the oracle $(\mathrm{Sim}^{Ver_{\mathrm{Samp},n}}_{Mix_n}, \mathrm{Sim}^{SG_{\mathrm{Samp},n}}_{SG_n})$.

Claim 4.9.

$$\max_{(\mathcal{A}_s,\mathcal{A}_t)} |\Pr[1 \leftarrow \textbf{Hybrid } \mathbf{6}(1^n)] - \Pr[1 \leftarrow \textbf{Hybrid } \mathbf{7}(1^n)]| \leq O\left(\frac{s}{2^{n/2}}\right). \tag{72}$$

Proof. First, from Theorem 2.6, we can conclude that, after $\mathcal{A}_s$ postselects on the RO such that $\mathcal{A}_s^{\mathsf{RO}} \to 1$, the state on the database register ρ_D will always satisfy $Tr(\Pi_{\leq s}\rho_D) = 1$. Now, define the projection $\Pi_{\{k_1,\ldots,k_t\}\cap D=\emptyset}$ onto databases which contain no elements that are in $\{k_1, \ldots k_t\}$. Since $k_1, \ldots k_t \leftarrow \{0,1\}^n \setminus \{0\}$ is a set of random elements, for all D such that $|D| \leq s$,

$$\Pr[D \cap \{k_1, \ldots k_t\} = \emptyset] = 1 - \Pr[\exists x \in D \cap \{k_1, \ldots k_t\}] \tag{73}$$

$$\geq 1 - \sum_{x \in \{k_1,\ldots k_t\}} \Pr[x \in D] \tag{74}$$

$$\geq 1 - \frac{s^2}{2^n}. \tag{75}$$

Note that, if this projector is applied after initialization, then **Hybrid 6** and **Hybrid 7** are identically distributed, since in **Hybrid 7** the fresh random oracle will only be queried on (x, y) such that either $x \in \{k_1, \ldots k_t\}$ or $y \in \{k_1, \ldots k_t\}$. And therefore the lemma follows by gentle measurement. □

Then, combining **Hybrid 5**, **Hybrid 6** and **Hybrid 7** and applying triangle inequality implies that

$$\max_{\mathcal{A}_s,\mathcal{A}_t} |\Pr[1 \leftarrow \textbf{Hybrid 5}(1^n)] - \Pr[1 \leftarrow \textbf{Hybrid 7}(1^n)]| \leq O(s \cdot 2^{-n/2}), \tag{76}$$

which is to say that the oracles $(\mathrm{Sim}_{Mix_n}^{Ver_{\mathrm{Samp},n}}, \mathrm{Sim}_{SG_n}^{SG_{\mathrm{Samp},n}})$, $(\mathrm{Sim}'^{Ver_{\mathrm{Samp},n}}_{Mix_n}, \mathrm{Sim}_{SG_n}^{SG_{\mathrm{Samp},n}})$ are $(s, t, O(s \cdot 2^{-n/2}))$ indistinguishable in the post selected random oracle model.

Setting $\gamma = 2^{-n}, \epsilon = O\left(\frac{S}{2^{n/4}}\right)$, from Corollary 3.3, this implies that, with probability $\geq 1 - O\left(\frac{S}{2^{n/4}}\right)$ over RO_n, for all T query algorithms $\mathcal{A}$ and advice strings c such that $|A| + |c| \leq S$,

$$\left|\Pr[\mathcal{A}^{(\mathrm{Sim}_{Mix_n}^{Ver_{\mathrm{Samp},n}}, \mathrm{Sim}_{SG_n}^{SG_{\mathrm{Samp},n}})} = 1] - \Pr[\mathcal{A}^{(\mathrm{Sim}'^{Ver_{\mathrm{Samp},n}}_{Mix_n}, \mathrm{Sim}_{SG_n}^{SG_{\mathrm{Samp},n}})} = 1]\right| \leq O\left(\frac{S}{2^{n/4}}\right). \tag{77}$$

Now, recall that we have already established that with probability $\geq 1 - 2^S \exp(-2^{n/2})$ over U_n, RO_n,

$$\max_{\mathcal{A},c} \left|\Pr[1 \leftarrow \mathcal{A}^{Mix,SG}(1^n, c)] - \Pr[\mathcal{A}^{(\mathrm{Sim}'^{Ver_{\mathrm{Samp},n}}_{Mix_n}, \mathrm{Sim}_{SG_n}^{SG_{\mathrm{Samp},n}})}(1^n, c) = 1]\right| \leq \epsilon + O\left(\frac{T^{2.5}}{2^{n/8}}\right), \tag{78}$$

which implies that with probability $\geq 1 - 2^S \exp(-2^{n/2}) - O\left(\frac{S}{2^{n/4}}\right) \geq 1 - O\left(\frac{S}{2^{n/4}}\right)$ (using the fact that $2^S \exp(-2^{n/2}) \leq S \cdot 2^{-n/2}$) over U_n, RO_n, for all T query algorithms $\mathcal{A}$ and advice strings c

$$\left|\Pr[1 \leftarrow \mathcal{A}^{Mix_n, SG_n}(1^n, c)] - \Pr[1 \leftarrow \mathcal{A}^{(\mathrm{Sim}_{Mix_n}^{Ver_{\mathrm{Samp},n}}, \mathrm{Sim}_{SG_n}^{SG_{\mathrm{Samp},n}})}(1^n, c)]\right| \tag{79}$$

$$\leq \epsilon + O\left(\frac{T^{2.5}}{2^{n/8}}\right) + O\left(\frac{S}{2^{n/4}}\right) \leq \epsilon + O\left(\frac{T^{2.5} + S}{2^{n/8}}\right), \tag{80}$$

□

5 Quantum Lightning Exists Relative to $(SG, Mix, PSPACE)$.

In this section, we show quantum lightning exists relative to the oracle $(SG, Mix, PSPACE)$.

Theorem 5.1. *With probability* 1 *over* $\{U_n\}_{n\in\mathbb{N}}$ *and* $\{\mathsf{RO}_n\}_{n\in\mathbb{N}}$*, quantum lightning exists relative to* $(SG, Mix, PSPACE)$*.*

Proof. Our construction is as follows.

1. $\mathsf{Mint}(1^n)$: Query $|0\rangle$ to SG_n to get , where $|\phi_x\rangle := U_n|x\rangle$, and measure the first register to get the pair $(k, |\phi_k\rangle)$ of the measurement result k and the post-measurement state $|\phi_k\rangle$. Output the serial number $\sigma := k$ and the bold state $|\$\rangle := |\phi_k\rangle$.

2. $\mathsf{Ver}(1^n, \sigma, |\$\rangle)$: Query $|\sigma\rangle\,|0\rangle\,|\$\rangle\,|0\rangle\,|0\rangle$ to Mix_n, and measure the last register. Output the measurement result.

Correctness follows by construction. Next, we show security. Let $\mathcal{A}$ be an attacker against the quantum lightning security of $(\mathsf{Mint}, \mathsf{Ver})$. In particular, we will take $\mathcal{A}^{SG_n, Mix_n, PSPACE}(1^n, c)$ to be a uniform algorithm with non-uniform advice c and making T queries to its oracles, where $|\mathcal{A}| + |c| \leq \mathsf{poly}(n)$. We will first show that with all but negligible probability over U_n, RO_n (and for any fixed $\{U_\ell\}_{\ell \in \mathbb{N} \setminus \{n\}}, \{\mathsf{RO}_\ell\}_{\ell \in \mathbb{N} \setminus \{n\}}$), for all c,

$$\Pr[\mathcal{A}(1^n, c) \text{ wins the quantum lightning game}] \leq \mathsf{negl}(n). \tag{81}$$

In particular, let $p(n)$ be any polynomial. We will show that for all sufficiently large n,

$$\Pr[\mathcal{A}(1^n, c) \text{ wins the quantum lightning game}] \leq 1/p(n). \tag{82}$$

To show it, we define the following sequence of games.

1. $\mathbf{G}_1$: It is the original security game of quantum lightning.
   (a) Run $\mathcal{A}^{SG, Mix, PSPACE}(1^n, c) \to (x, \rho_{\mathbf{AB}})$.
   (b) Apply the measurement $\{|\phi_x, \phi_x\rangle\langle\phi_x, \phi_x|, I - |\phi_x, \phi_x\rangle\langle\phi_x, \phi_x|\}$ to $\rho_{\mathbf{AB}}$. Output $\top$ if measurement results in the first outcome. Otherwise, output $\bot$.

2. $\mathbf{G}_2$: It is the same as $\mathbf{G}_1$ except that it replaces all calls to SG_n and Mix_n with queries to $\mathrm{Sim}_{SG_n}^{SG_{\mathrm{Samp},n}}$ and $\mathrm{Sim}_{Mix_n}^{Ver_{\mathrm{Samp},n}}$ respectively, from Corollary 4.2 instantiated with error $\epsilon = \frac{1}{6p(n)}$.

3. $\mathbf{G}_3$: It is the same as $\mathbf{G}_2$ except that it outputs $\bot$ if the initialization for $SG_{\mathrm{Samp},n}$ samples the same k twice.

4. $\mathbf{G}_4$: It is the same as $\mathbf{G}_3$ except that it samples each $|\phi_k\rangle$ independently at random instead of guaranteeing they are all orthogonal. In particular, $\mathbf{G}_4$ is the quantum lightning game where the oracles SG_n, Mix_n are replaced by $\mathrm{Sim}_{SG_n}^{SG'_{\mathrm{Samp},n}}, \mathrm{Sim}_{Mix_n}^{Ver'_{\mathrm{Samp},n}}$ with $SG'_{\mathrm{Samp},n}$ and $Ver'_{\mathrm{Samp},n}$ defined as follows.
   (a) On initialization, sample $k_1 \neq \ldots \neq k_t$ uniformly at random from $\{0,1\}^n \setminus \{0\}$, and sample $|\phi_{k_i}\rangle \leftarrow \mu_{2^n}^s$.
   (b) On the ith query, $SG'_{\mathrm{Samp},n}$ will output $|k_i\rangle |\phi_{k_i}\rangle$.
   (c) $Ver'_{\mathrm{Samp},n}$ is the map such that, for all $i \in [t]$,

$$|k_i\rangle |\phi_{k_i}\rangle |b\rangle \mapsto |k_i\rangle |\phi_{k_i}\rangle |\bar{b}\rangle \tag{83}$$

   and acts as the identity for all orthogonal states.

Now we show each hybrids are close with each other.

1. By Corollary 4.2, with probability $1 - \mathsf{negl}(n)$ over the choice of oracle SG_n, Mix_n, for all $\mathcal{A}, c$,

$$|\Pr[1 \leftarrow \mathbf{G}_1(1^n)] - \Pr[1 \leftarrow \mathbf{G}_2(1^n)]| \leq \frac{1}{6p(n)}. \tag{84}$$

2. We know that $|\Pr[1 \leftarrow \mathbf{G}_2(1^n)] - \Pr[1 \leftarrow \mathbf{G}_3(1^n)]| \leq \Pr[\bot \leftarrow \mathbf{G}_3(1^n)]$. But this is exactly the probability that there is a repeat among $\leq T = \mathsf{poly}(n)$ samples of a random value from $\{0,1\}^n$. By the birthday bound, we get that with probability 1 over the choice of SG_n, Mix_n, for all $\mathcal{A}, c$,

$$|\Pr[1 \leftarrow \mathbf{G}_2(1^n)] - \Pr[1 \leftarrow \mathbf{G}_3(1^n)]| \leq \frac{T(n)^2}{2^n} \leq \frac{1}{6p(n)}. \tag{85}$$

3. Corollary 2.25 immediately gives with probability 1 over the choice of SG_n, Mix_n, for all $\mathcal{A}, c$,

$$|\Pr[1 \leftarrow \mathbf{G}_3(1^n)] - \Pr[1 \leftarrow \mathbf{G}_4(1^n)]| \leq \mathsf{negl}(n). \tag{86}$$

Therefore by triangle inequality, with probability $1-\mathsf{negl}(n)$ over the choice of SG, Mix, for all $\mathcal{A}, c$,

$$|\Pr[1 \leftarrow \mathbf{G}_1(1^n)] - \Pr[1 \leftarrow \mathbf{G}_4(1^n)]| \leq \frac{1}{2p(n)}. \tag{87}$$

So it is sufficient to establish that $\Pr[1 \leftarrow \mathbf{G}_4(1^n)] \leq \mathrm{negl}(n)$. Let $\mathcal{A}^{SG,Mix,PSPACE}$ be any adversary such that $\Pr[\mathbf{G}_4 \to 1] = \epsilon$. We will construct an inefficient adversary $\mathcal{A}'^{P_{|\psi\rangle}}$ cloning a single Haar-random state $|\psi\rangle$ with access to a verification oracle $P_{|\psi\rangle}$.

$\mathcal{A}'$ behaves as follows.

1. On input $|\psi\rangle$.
2. Sample $k_1 \neq \ldots \neq k_t \leftarrow \{0,1\}^n \setminus \{0\}$.
3. Sample $i^* \leftarrow [t]$. Set $|\phi_{k_{i^*}}\rangle = |\psi\rangle$.
4. For $i \neq i^*$, sample $|\phi_{k_i}\rangle$ from the Haar distribution.
5. Define $\widetilde{SG_{\mathrm{Samp},n}}, \widetilde{Ver_{\mathrm{Samp},n}}$ as follows:
   (a) On the ith query, $\widetilde{SG_{\mathrm{Samp},n}}$ outputs $|k_i\rangle |\phi_{k_i}\rangle$.
   (b) $Ver'_{\mathrm{Samp},n}$ is the map such that, for all $i \in [t]$,
   $$|k_i\rangle |\phi_{k_i}\rangle |b\rangle \mapsto |k_i\rangle |\phi_{k_i}\rangle |\bar{b}\rangle \tag{88}$$
   and acts as the identity for other orthogonal states.
6. Run $(y, \rho) \leftarrow \mathcal{A}^{\mathrm{Sim}_{SG_n}^{\widetilde{SG_{\mathrm{Samp},n}}}, \mathrm{Sim}_{Mix_n}^{\widetilde{Ver_{\mathrm{Samp},n}}}, PSPACE}(1^n, c)$.
7. If $y = k_{i^*}$, output ρ.

Note that the view of $\mathcal{A}$ is exactly the same view as in $\mathbf{G}_4$. Therefore as long as $\mathcal{A}'$ correctly guesses the index $\mathcal{A}$ wins at, $\mathcal{A}'$ will also win. That is,

$$\Pr[\mathcal{A}' \text{ wins cloning game}] \geq \Pr[1 \leftarrow \mathbf{G}_4(1^n)] \cdot \Pr[k_{i^*} = y] \geq \frac{\epsilon}{t}. \tag{89}$$

By Theorem 2.26, we know that any adversary making at most $2^{n/4}$ queries cannot win the cloning game for a single Haar random state with probability above $2^{-n/2}$. Since $\mathcal{A}'$ makes at most polynomially many queries, we have that $\Pr[\mathcal{A}'(1^n) \text{ wins cloning game}] \leq 2^{-n/2}$. And so $\epsilon = \Pr[\mathbf{G}_4 \to 1] \leq t \cdot 2^{-n/2} = \mathsf{negl}(n)$.

Thus, by Equation (89), $\Pr[1 \leftarrow \mathbf{G}_4(1^n)] \leq \mathsf{negl}(n)$, and therefore, by Equation (87) for all advice c, with probability $\geq 1-\mathsf{negl}(n)$ over SG_n, Mix_n, for all $\mathcal{A}, c$, for all polynomials $p(n)$,

$$\Pr[\mathcal{A}(1^n, c) \text{ wins the quantum lightning game}] \leq \frac{1}{p(n)}. \tag{90}$$

Therefore with probability $\geq 1 - \mathsf{negl}(n)$ over SG_n, Mix_n, for all $\mathcal{A}, c$,

$$\Pr[\mathcal{A}(1^n, c) \text{ wins the quantum lightning game}] \leq \mathsf{negl}(n). \tag{91}$$

But note that since $\sum_{n=1}^{\infty} \mathsf{negl}(n)$ converges, by the Borel-Cantelli lemma [ÉB09, Can17] $\mathcal{A}$ achieves negligible advantage for all but finitely many input lengths $n \in \mathbb{N}$ with probability 1 over $\{U_\ell\}_{\ell\in\mathbb{N}}, \{\mathsf{RO}_\ell\}_{\ell\in\mathbb{N}}$. In particular, $(\mathsf{Mint}, \mathsf{Ver})$ is a quantum lightning scheme. □

Establishing BQP = QMA The main barrier to showing **BQP** = **QMA** relative to our choice of oracles $Mix, SG, PSPACE$ is that Mix relies on the random oracle. However, consider the following oracle $\{Ver_\ell\}_{\ell\in\mathbb{N}}$:

- Ver_ℓ: denotes the unitary which takes as input $(x, |\psi\rangle)$, and performs the following mapping:

$$|k\rangle |\phi_k\rangle |b\rangle \mapsto |k\rangle |\phi_k\rangle |\bar{b}\rangle, \tag{92}$$

  where $|\phi_k\rangle = U_\ell |k\rangle$ and acts as identity on all orthogonal states.

Now, it is easy to see that quantum lightning still exists relative to SG, Ver even against inefficient adversaries (although the same doesn't apply for the other primitives we consider). We have the following construction:

1. $\mathsf{Mint}(1^n)$: Query $|0\rangle$ to SG_n to get , where $|\phi_x\rangle := U_n|x\rangle$, and measure the first register to get the pair $(k, |\phi_k\rangle)$ of the measurement result k and the post-measurement state $|\phi_k\rangle$. Output the serial number $\sigma := k$ and the bold state $|\$\rangle := |\phi_k\rangle$.

2. $\mathsf{Ver}(1^n, \sigma, |\$\rangle)$: Query $|\sigma\rangle |\$\rangle |0\rangle$ to Ver_n, and measure the last register. Output the measurement result.

The proof of security is essentially identical to the one relative to SG, Mix. Unfortunately, it is not clear that $\mathbf{BQP}^{SG,Ver,PSPACE} = \mathbf{QMA}^{SG,Ver,PSPACE}$ directly. However, following the techniques of [Kre21], it is not difficult to define a recursive oracle $\mathcal{C}$ (essentially $PSPACE^{SG_{\text{Samp}}, Ver_{\text{Samp}}}$) such that $\mathbf{BQP}^{SG,Ver,\mathcal{C}} = \mathbf{QMA}^{SG,Ver,\mathcal{C}}$.

6 QCCC NIKE Exists Relative To $(SG, Mix, PSPACE)$

In the following, we construct two-party QCCC NIKE. Using a modified Mix_n oracle which takes in an arbitrary number of inputs, it is not difficult to give an oracle relative to which multiparty NIKE exists but **BQP** = **QCMA**.

Theorem 6.1. *With probability* 1 *over* $\{U_n\}_{n\in\mathbb{N}}$, $\{\mathsf{RO}_n\}_{n\in\mathbb{N}}$, *relative to the oracle* $(SG, Mix, PSPACE)$, *QCCC NIKE exist.*

Our NIKE protocol operates as follows.

1. On input 1^n, Alice queries $|0\rangle$ to SG_n to obtain $\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle |\phi_x\rangle$, and measures the first register to obtain the measurement result x and the post-measurement state $|\phi_x\rangle$. Alice sends x to Bob.

2. On input 1^n, Bob queries $|0\rangle$ to SG_n to obtain $\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle |\phi_x\rangle$, and measures the first register to obtain the measurement result y and the post-measurement state $|\phi_y\rangle$. Bob sends y to Alice.

3. Alice queries $|x\rangle|y\rangle|\phi_x\rangle|0\rangle|0\rangle$ to Mix_n, and measures the fourth register to get $\mathsf{RO}_n(x,y)$. Alice outputs $\mathsf{RO}_n(x,y)$.

4. Bob queries $|x\rangle|y\rangle|\phi_y\rangle|0\rangle|0\rangle$ to Mix_n, and measures the fourth register to obtain $\mathsf{RO}_n(x,y)$. Bob outputs $\mathsf{RO}_n(x,y)$.

Agreement follows by construction. It only remains to show security. Without loss of generality we assume the adversary is a T-query uniform quantum algorithm $\mathcal{A}^{SG_n,Mix_n,PSPACE}(1^n,c,x,y)$ where (x,y) is the transcript and $c\in\{0,1\}^{\mathsf{poly}(n)}$ is the advice string. We will first show that with all but negligible probability over U_n,RO_n (and for any fixed $\{U_\ell\}_{\ell\in\mathbb{N}\setminus\{n\}},\{\mathsf{RO}_\ell\}_{\ell\in\mathbb{N}\setminus\{n\}}$), for all c, and all polynomials p,

$$\Pr_{x,y}[\mathcal{A}(1^n,c,x,y)=\mathsf{RO}_n(x,y)]\leq\frac{1}{p(n)}. \tag{93}$$

We will use hybrids to switch our NIKE security game to one where we can more easily argue that the adversary has negligible advantage.

1. $\mathbf{G}_1$ is the original security game of the NIKE played with $\mathcal{A}^{SG_n,Mix_n,PSPACE}$. More formally, it runs as follows:
   (a) Query $|0\rangle$ to SG_n to get $\frac{1}{\sqrt{2^n}}\sum_x|x\rangle|\phi_x\rangle$, and measure the first register to obtain the pair $(x,|\phi_x\rangle)$ of the measurement result x and the post-measurement state $|\phi_x\rangle$.
   (b) Query $|0\rangle$ to SG_n to get $\frac{1}{\sqrt{2^n}}\sum_k|k\rangle|\phi_k\rangle$, and measure the first register to obtain the pair $(y,|\phi_y\rangle)$ of the measurement result y and the post-measurement state $|\phi_y\rangle$.
   (c) Run $\mathcal{A}^{SG_n,Mix_n,PSPACE}(1^n,c,x,y)\to guess$.
   (d) Output 1 if and only if $guess=\mathsf{RO}_n(x,y)$.

2. $\mathbf{G}_2$ is the same as $\mathbf{G}_1$, except that we will prevent $\mathcal{A}$ from querying Mix_n on $|\phi_x\rangle$. In particular, every time $\mathcal{A}$ queries the oracle Mix_n on $\rho_{\mathbf{ABCDE}}$, we will first apply the measurement $\{|\phi_x\rangle\langle\phi_x|,I-|\phi_x\rangle\langle\phi_x|\}$ on register $\mathbf{C}$, and if the result is the first option the game will output 0. Otherwise, we apply Mix_n on the residual state.

3. $\mathbf{G}_3$ is the same as $\mathbf{G}_2$, except that (analogously to the previous hybrid) we prevent $\mathcal{A}$ from querying Mix_n on $|\phi_y\rangle$. Formally, define $\Pi_{bad}=I_{\mathbf{AB}}\otimes(|\phi_x\rangle\langle\phi_x|_{\mathbf{C}}+|\phi_y\rangle\langle\phi_y|_{\mathbf{C}})\otimes I_{\mathbf{DE}}$. $\mathbf{G}_3$ will be the same as $\mathbf{G}_1$, but whenever $\mathcal{A}$ queries Mix_n on $\rho_{\mathbf{ABCDE}}$, it will apply the measurement $\{\Pi_{bad},I-\Pi_{bad}\}$. If the result is the first option, the game will output 0. Otherwise, we apply Mix_n on the residual state. Note that this measurement can be exactly implemented by making a query to Mix_n on $|x\rangle\langle x|_{\mathbf{A}'}\otimes|y\rangle\langle y|_{\mathbf{B}'}\otimes\rho_{\mathbf{C}}\otimes|0\rangle\langle 0|_{\mathbf{D}'\mathbf{E}'}$ measuring if register $\mathbf{E}'$ contains 1, and then applying Mix_n again to clear registers $\mathbf{D}'\mathbf{E}'$.

4. $\mathbf{G}_4$ is the same as $\mathbf{G}_3$, except that we replace SG_n,Mix_n with $\mathrm{Sim}_{SG_n}^{SG_{Samp,n}},\mathrm{Sim}_{Mix_n}^{Ver_{Samp,n}}$ with error $\frac{1}{2p(n)}$. In detail, this will be identical to the following game
   (a) Sample a fresh random unitary $\widetilde{U}_n$ and a random oracle $\widetilde{\mathsf{RO}}_n$ for $SG_{\mathrm{Samp},n},Ver_{\mathrm{Samp},n}$.
   (b) Sample $x,y\leftarrow\{0,1\}^n$.
   (c) Define $\widetilde{\Pi}_{bad}=I_{\mathbf{AB}}\otimes(\widetilde{U}_n|x\rangle\langle x|\widetilde{U}_n^\dagger+\widetilde{U}_n|y\rangle\langle y|\widetilde{U}_n^\dagger)_{\mathbf{C}}\otimes I_{\mathbf{DE}}$.

(d) Run $\mathcal{A}^{\mathrm{Sim}_{SG_n}^{SG_{Samp,n}},\mathrm{Sim}_{Mix_n}^{Ver_{Samp,n}}}(1^n, c, x, y) \to guess$, but every time $\mathcal{A}$ makes a $\mathrm{Sim}_{Mix_n}^{Ver_{Samp,n}}$ query on a state $\rho_{\mathbf{ABCDE}}$, apply the measurement $\{\widetilde{\Pi}_{bad}, I - \widetilde{\Pi}_{bad}\}$ on registers $\mathbf{ABC}$ and if the result is the first outcome, the game will output 0.

(e) Output 1 if and only if $guess = \widetilde{\mathsf{RO}}_n(x, y)$.

Lemma 6.2. *With probability* 1 *over* SG_n, Mix_n*, for all* $\mathcal{A}, c$

$$|\Pr[1 \leftarrow \mathbf{G}_1(1^n)] - \Pr[1 \leftarrow \mathbf{G}_2(1^n)]| \leq \mathsf{negl}(n). \tag{94}$$

Proof. We will show that for all t, if $\rho_{\mathbf{ABCDE}}$ is the state input to Mix_n in the t-th query, then $\mathrm{Tr}((I_{\mathbf{ABDE}} \otimes |\phi_x\rangle\langle\phi_x|_{\mathbf{C}})\rho_{\mathbf{ABCDE}}) \leq \mathsf{negl}(n)$. The lemma then follows immediately from gentle measurement and induction.

In fact, this claim follows immediately from the security our quantum lightning construction. In particular, let $\mathcal{A}(\cdot, c)$ be any adversary with advice c such that the state $\rho_{\mathbf{ABCDE}}$ input to Mix_n in the t-th query of $\mathbf{G}_1$ satisfies

$$\mathrm{Tr}((I_{\mathbf{ABDE}} \otimes |\phi_x\rangle\langle\phi_x|_{\mathbf{C}})\rho_{\mathbf{ABCDE}}) \geq \frac{1}{p(n)} \tag{95}$$

for some polynomial p. Then we get an adversary $\mathcal{A}'(1^n, c)$ for quantum lightning defined as follows.

1. Run $\mathsf{Mint}(1^n) \to (x, |\phi_x\rangle)$ and $\mathsf{Mint}(1^n) \to (y, |\phi_y\rangle)$.
2. Run $\mathcal{A}^{SG_n, Mix_n, PSPACE}(1^n, c, x, y)$ up until the t-th query to Mix_n, generating a query state $\rho_{\mathbf{ABCDE}}$.
3. Output $|\phi_x\rangle\langle\phi_x| \otimes \rho_{\mathbf{C}}$.

Note that the view of $\mathcal{A}$ generated by $\mathcal{A}'$ is identical to its view in $\mathbf{G}_1$. And therefore $\rho_{\mathbf{C}}$ measures to $|\phi_x\rangle$ with probability $\frac{1}{p(n)}$. But so does $|\phi_x\rangle$, and therefore $\mathcal{A}'$ breaks the lightning protocol with probability $\frac{1}{p(n)}$. □

Analogously, we also have

Lemma 6.3. *With probability* 1 *over* SG_n, Mix_n*, for all* $\mathcal{A}, c$

$$|\Pr[1 \leftarrow \mathbf{G}_2(1^n)] - \Pr[1 \leftarrow \mathbf{G}_3(1^n)]| \leq \mathsf{negl}(n). \tag{96}$$

Corollary 4.2 immediately implies

Lemma 6.4. *With probability* $1 - \mathsf{negl}(n)$ *over* U_n, RO_n*, for all* $\mathcal{A}, c$

$$|\Pr[1 \leftarrow \mathbf{G}_4(1^n)] - \Pr[1 \leftarrow \mathbf{G}_3(1^n)]| \leq \frac{1}{2p(n)}. \tag{97}$$

Finally, it is clear that

Lemma 6.5. *For all* SG_n, Mix_n*, for all* $\mathcal{A}, c$

$$\Pr[1 \leftarrow \mathbf{G}_4(1^n)] \leq \frac{1}{2^n}. \tag{98}$$

In particular, $\mathcal{A}(1^n, c, x, y)$ must guess $\mathsf{RO}_n(x, y)$, but its oracles are independent of $\mathsf{RO}_n(x, y)$. Thus, by lazy sampling, it is equivalent if $\mathsf{RO}_n(x, y)$ is sampled after $\mathcal{A}$ produces its output. The lemma follows.

Combining these lemmas, we get that with probability $1 - \mathsf{negl}(n)$ over U_n, RO_n, for all $\mathcal{A}, c$,

$$\Pr_{x,y}[\mathcal{A}^{SG_n, Mix_n, PSPACE}(1^n, c, x, y) = \mathsf{RO}_n(x, y)] \leq \mathsf{negl}(n). \tag{99}$$

But note that since $\sum_{n=1}^{\infty} \mathsf{negl}(n)$ converges, by the Borel-Cantelli lemma [ÉB09, Can17] $\mathcal{A}$ achieves negligible advantage for all but finitely many input lengths $n \in \mathbb{N}$ with probability 1 over $\{U_\ell\}_{\ell\in\mathbb{N}}$ and $\{\mathsf{RO}_\ell\}_{\ell\in\mathbb{N}}$. In particular, with probability 1, this protocol is a secure NIKE scheme relative to $(SG, Mix, PSPACE)$.

Generalizing to QCCC Multiparty NIKE We can generalize our 2 party protocol for non interactive key exchange to non interactive key exchange with polynomially many parties under a slightly modified (but still unitary) version of $Mix = \{Mix_\ell\}_{\ell\in\mathbb{N}}$.

- Let n be the security parameter, and let the number of parties in the multiparty NIKE protocol be $p(n)$, where p is some polynomial. The oracle $Mix^* = \{Mix^*_\ell\}_{\ell\in\mathbb{N}}$ is the following unitary:

$$\sum_{x_1,\dots x_{p(n)}\in\{0,1\}^\ell} |x_1\rangle\langle x_1| \otimes \cdots \otimes |x_{p(n)}\rangle\langle x_{p(n)}| \otimes (|\phi_{x_1}\rangle\langle\phi_{x_1}| \cdots + |\phi_{x_{p(n)}}\rangle\langle\phi_{x_{p(n)}}|) \tag{100}$$

$$\otimes X^{\mathsf{RO}(x_1,\dots x_{p(n)})} \otimes X \tag{101}$$

$$+ \sum_{x_1,\dots x_{p(n)}\in\{0,1\}^\ell} |x_1\rangle\langle x_1| \otimes \cdots \otimes |x_{p(n)}\rangle\langle x_{p(n)}| \otimes (I - |\phi_{x_1}\rangle\langle\phi_{x_1}| - \cdots - |\phi_{x_{p(n)}}\rangle\langle\phi_{x_{p(n)}}|) \tag{102}$$

$$\otimes I \otimes I. \tag{103}$$

  Here, X is the Pauli X operator and $X^r := \bigotimes_{i=1}^n X_i^{r_i}$ for any n-bit string r. In other words, Mix_ℓ performs the following operation:

$$|x_1\rangle \dots |x_{p(n)}\rangle |\phi_{x_1}\rangle |z\rangle |b\rangle \to |x_1\rangle \dots |x_{p(n)}\rangle |\phi_x\rangle |z \oplus \mathsf{RO}(x_1, \dots x_{p(n)})\rangle |\bar{b}\rangle \tag{104}$$

$$\vdots \tag{105}$$

$$|x_1\rangle \dots |x_{p(n)}\rangle |\phi_{x_{p(n)}}\rangle |z\rangle |b\rangle \to |x_1\rangle \dots |x_{p(n)}\rangle |\phi_x\rangle |z \oplus \mathsf{RO}(x_1, \dots x_{p(n)})\rangle |\bar{b}\rangle \tag{106}$$

  and acts as identity for other orthogonal states.

SG is defined identically to before.
The 2 party NIKE protocol can be generalized to $p(n)$ parties $P_1, \dots P_{p(n)}$ trivially:

1. On input 1^n, $\forall i \in [p(n)], P_i$ queries $|0\rangle$ to SG_n to obtain $\frac{1}{\sqrt{2^n}} \sum_{x_i\in\{0,1\}^n} |x_i\rangle |\phi_{x_i}\rangle$, and measures the first register to obtain the measurement result x_i and the post-measurement state $|\phi_{x_i}\rangle$. P_i sends x_i to $\{P_j\}_{j\neq i}$

2. $\forall i \in [p(n)]$, P_i queries $|x_1\rangle \dots |x_{p(n)}\rangle |\phi_{x_i}\rangle |z\rangle |b\rangle$ to Mix^*_n, and measures the fourth register to get $\mathsf{RO}_n(x_1, \dots x_{p(n)})$. $\forall i \in [p(n)], P_i$ outputs $\mathsf{RO}_n(x_1, \dots x_{p(n)})$.

7 QCCC Commitments Exist Relative To $(SG, Mix, PSPACE)$

Since this section details a construction, we will show that our construction satisfies stronger security requirements than given in Definition 2.20. Note that if a bit commitment scheme satisfies statistical hiding and sum-binding, then it trivially satisfies computational hiding and weak computational binding.

Definition 7.1 (Statistical hiding [HR07])**.** *Let* $(\mathsf{Com}, \mathsf{Rec})$ *be a bit-commitment scheme. We say that* $(\mathsf{Com}, \mathsf{Rec})$ *satisfies statistical hiding if for all (inefficient) adversarial receivers* Rec'*, the transcript of the commit stage between* Rec' *and* Com *with input* m *is statistically close to the transcript of the commit stage between* Rec' *and* Com *with input* m'*. That is,*

$$\Delta(\mathsf{Com}(m) \rightleftarrows \mathsf{Rec}', \mathsf{Com}(m') \rightleftarrows \mathsf{Rec}') \leq \mathsf{negl}(n). \tag{107}$$

Here $\mathsf{Com}(m) \rightleftarrows \mathsf{Rec}'$ *is the distribution of the transcript of the commit stage between* Rec' *and* Com *with input* m*.*

Definition 7.2 (Sum-Binding [Unr16])**.** *Let* $(\mathsf{Com}, \mathsf{Rec})$ *be a bit-commitment scheme. We say that* $(\mathsf{Com}, \mathsf{Rec})$ *satisfies sum-binding if for all QPT adversarial senders* Com'*, the probability that* Com' *wins the following game is* $\leq \frac{1}{2} + \mathsf{negl}(n)$*.*

1. *In the commit stage, an honest receiver* Rec *interacts with the adversarial committer* Com' *to produce a transcript* z *and the receiver state* ρ_{Rec}*.* Com' *produces the adversarial committer state* $\rho_{\mathsf{Com}'}$*.*
2. *The challenger samples a bit* $b \leftarrow \{0,1\}$*.*
3. *In the opening stage, the honest receiver* Rec *is given* ρ_{Rec} *and* z*, while* Com' *is given* $z, \rho_{\mathsf{Com}'}$*, and* b*. They then proceed to run the opening stage with the committer replaced by* Com'*, and* Rec *produces a final output* b'*.* Com' *wins if* $b' = b$*.*

Theorem 7.3. *With probability* 1 *over* $\{U_m\}_{m\in\mathbb{N}}$ *and* $\{\mathsf{RO}_m\}_{m\in\mathbb{N}}$*, there exists a bit commitment scheme satisfying statistical hiding and sum-binding relative to* $SG, Mix, PSPACE$*.*

Proof. We construct a commitment scheme $(\mathsf{Com}^{SG,Mix,PSPACE}, \mathsf{Rec}^{SG,Mix,PSPACE})$ as follows. Note that our commitment scheme has a non-interactive commitment stage and an interactive opening stage.

1. To commit to b, Com queries $|0\rangle$ to SG_n to generate $SG_n|0\rangle$ and measures the first register to get $(x, |\phi_x\rangle)$. Let x_1 be the first bit of x and let $x_{>1}$ be the last $n-1$ bits of x. Com repeats this process until $x_1 = b$. Com sends $x_{>1}$ to Rec.
2. To open to a bit b, Rec queries $|0\rangle$ to SG_n to generate $SG_n|0\rangle$ and measures the first register to get $(y, |\phi_y\rangle)$. Rec sends y to Com.
3. Upon receiving y, Com computes $Mix_n|x\rangle|y\rangle|\phi_x\rangle|0\rangle|0\rangle$, measures the fourth register to get the measurement result r, and sends r to Rec.
4. Rec computes $Mix_n|y\rangle|b, x_{>1}\rangle|\phi_y\rangle|0\rangle|0\rangle$ and measures the fourth register to get the result r'. Rec accepts if $r = r'$. Otherwise, Rec rejects.

Correctness is clear by construction. Note that the commitment $x_{>1}$ is a uniformly random string, and therefore $(\mathsf{Com}, \mathsf{Rec})$ trivially satisfies statistical hiding.

It remains to show computational binding. We will define a sequence of hybrids $\mathbf{G}_1, \ldots, \mathbf{G}_5$ where each hybrid is defined relative to some t-query oracle algorithm $\mathcal{A}^{SG,Mix,PSPACE}(1^n, c)$.

1. $\mathbf{G}_1$ will be the original sum-binding security game. We will rewrite it for simplicity as follows
   (a) Sample $b \leftarrow \{0,1\}$.
   (b) Query $|0\rangle$ to SG_n to generate $SG_n|0\rangle$, and measure the first register to get $(y, |\phi_y\rangle)$.
   (c) Run $(x, \rho_{\mathcal{A}}) \leftarrow \mathcal{A}^{SG,Mix,PSPACE}(1^n, c)$.
   (d) Run $z \leftarrow \mathcal{A}^{SG,Mix,PSPACE}(b, x, y, \rho_{\mathcal{A}})$.
   (e) Generate $Mix_n|y\rangle|b,x\rangle|\phi_y\rangle|0\rangle|0\rangle$ and measure the fourth register. If the result is z, output 1. Otherwise, output 0.

2. $\mathbf{G}_2$ is the same as $\mathbf{G}_1$, except that we will prevent $\mathcal{A}$ from querying Mix_n on $|\phi_y\rangle$. In particular, every time $\mathcal{A}$ queries the oracle Mix_n on $\rho_{\mathbf{ABCDE}}$, we will first apply the measurement $\{|\phi_y\rangle\langle\phi_y|, I - |\phi_y\rangle\langle\phi_y|\}$ on register $\mathbf{C}$, and if the result is the first option the game will output 0. Otherwise, we apply Mix_n on the residual state. Note that this measurement can be performed by querying Mix_n on $|y\rangle\langle y|_{\mathbf{A}'} \otimes |0\rangle\langle 0|_{\mathbf{B}'} \otimes \rho_C \otimes |0\rangle\langle 0|_{\mathbf{C}'\mathbf{D}'}$, measuring whether register D' contains $|1\rangle$, and then applying Mix_n again to clear out registers C' and D'.

3. $\mathbf{G}_3$ is the same as $\mathbf{G}_2$, except that we replace all calls to SG_n and Mix_n (including the ones used to evaluate $\{|\phi_x\rangle\langle\phi_x|, I - |\phi_x\rangle\langle\phi_x|\}$) with $\mathrm{Sim}_{SG,n}^{SG_{\mathrm{Samp},n}}, \mathrm{Sim}_{Mix,n}^{Ver_{\mathrm{Samp},n}}$ for $\epsilon = \frac{1}{2p(n)}$.

4. $\mathbf{G}_4$ is the same as $\mathbf{G}_3$, except that we resample all random oracle outputs $\mathsf{RO}(w,y)$ for all inputs w for which $\mathcal{A}$ does not have access to $|\phi_z\rangle$. In particular, let $k_1, \ldots, k_t$ be the inputs for $SG_{\mathrm{Samp},n}$ chosen during initialization. After $\mathcal{A}$ outputs the decommitment z, for all $w \notin \{k_1, \ldots, k_t\}$, we will replace $\mathsf{RO}'_n(w,y)$ with a fresh random value.

5. $\mathbf{G}_5$ will be the same as $\mathbf{G}_4$, except that we require for all $i \neq j \in [t]$, k_i, k_j sampled during initialization must differ in the last $n-1$ bits. That is, $(k_i)_{>1} \neq (k_j)_{>1}$.

Claim 7.4. *With probability* 1 *over* SG, Mix, *and for all* $\mathcal{A}, c$

$$|\Pr[1 \leftarrow \mathbf{G}_1(1^n)] - \Pr[1 \leftarrow \mathbf{G}_2(1^n)]| \leq \mathsf{negl}(n). \tag{108}$$

Proof. This follows the exact same proof structure as Lemma 6.2. Intuitively, if $\mathcal{A}$ ever queries on $|\phi_y\rangle$, since the game itself has another copy of $|\phi_y\rangle$, this would violate security of our quantum lightning scheme. □

Claim 7.5. *For all sufficiently large* n, *with probability* $\geq 1 - \mathsf{negl}(n)$ *over* U_n, RO_n, *for all* $\mathcal{A}, c$

$$|\Pr[1 \leftarrow \mathbf{G}_3(1^n)] - \Pr[1 \leftarrow \mathbf{G}_2(1^n)]| \leq \frac{1}{2p(n)}. \tag{109}$$

This follows immediately from Corollary 4.2

Claim 7.6. *For all* $\{U_\ell\}_{\ell\in\mathbb{N}}, \{\mathsf{RO}_\ell\}_{\ell\in\mathbb{N}}$, *for all* $n \in \mathbb{N}$, *and for all* $\mathcal{A}, c$,

$$\Pr[1 \leftarrow \mathbf{G}_4(1^n)] = \Pr[1 \leftarrow \mathbf{G}_3(1^n)]. \tag{110}$$

Proof. This follows immediately from the fact that $\mathcal{A}$'s oracle is independent of $\mathsf{RO}'_n(w,y)$ for $w \notin \{k_1, \dots k_t\}$. This is because by construction any query to $\mathrm{Sim}_{Mix,n}^{Ver_{\mathrm{Samp},n}}$ on input $\rho_{\mathbf{ABCDE}}$ is independent of $\mathsf{RO}(w,y)$ unless either $\rho_{\mathbf{C}}$ has some overlap with $|\phi_y\rangle\langle\phi_y|$ or $w \in \{k_1, \dots, k_t\}$. But note that by construction neither game ever queries $\mathrm{Sim}_{Mix,n}^{Ver_{\mathrm{Samp},n}}$ on $\rho_{\mathbf{ABCDE}}$ where $\rho_{\mathbf{C}}$ has any overlap with $|\phi_y\rangle\langle\phi_y|$. □

Claim 7.7. *With probability* 1 *over* $\{U_\ell\}_{\ell\in\mathbb{N}}$ *and* $\{\mathsf{RO}_\ell\}_{\ell\in\mathbb{N}}$*, for all* $\mathcal{A}, c$*,*

$$|\Pr[1 \leftarrow \mathbf{G}_5(1^n)] - \Pr[1 \leftarrow \mathbf{G}_4(1^n)]| \leq \mathsf{negl}(n). \tag{111}$$

Proof. The only way to distinguish these two games is if in $\mathbf{G}_4$ there is some $i \neq j \in [t]$ such that $(k_i)_{>1} = (k_j)_{>1}$. By collision-bound, this occurs with probability $\leq \frac{t^2}{2^{n-1}} \leq \mathsf{negl}(n)$. □

Claim 7.8. $\Pr[1 \leftarrow \mathbf{G}_5(1^n)] \leq \frac{1}{2} + \frac{1}{2^n}$.

Proof. If $b.x$ (i.e. the concatenation of b and x) is not in $\{k_1, \dots, k_t\}$, then $\mathsf{RO}'_n(b.x, y)$ will be resampled after the decommitment z is chosen, and so $\mathcal{A}$ will succeed with probability $\frac{1}{2^n}$. However, at the point when x is chosen, $k_1, \dots, k_t$ have all been chosen. If $x \neq (k_i)_{>1}$ for some i, then it is necessarily the case that $b.x \notin \{k_1, \dots, k_t\}$. If $x = (k_i)_{>1}$ for some i, then unless $b = (k_i)_1$, it will be the case that $b.x \notin \{k_1, \dots, k_t\}$. And so in particular, with probability $\geq \frac{1}{2}$, it will be the case that $b.x \notin \{k_1, \dots, k_t\}$. And so the adversaries success probability is bounded by $\frac{1}{2} + \frac{1}{2^n}$. □

Combining these lemmas, we get that with probability $1 - \mathsf{negl}(n)$ over SG_n, Mix_n, and for all $\mathcal{A}, c$,

$$\Pr[\mathcal{A}(1^n, c, \cdot) \text{ breaks sum-binding}] \leq \frac{1}{2} + \mathsf{negl}(n). \tag{112}$$

But note that since $\sum_{n=1}^{\infty} \mathsf{negl}(n)$ converges, by the Borel-Cantelli lemma [ÉB09, Can17], $\mathcal{A}$ achieves negligible advantage for all but finitely many input lengths $n \in \mathbb{N}$ with probability 1 over $\{U_\ell\}_{\ell\in\mathbb{N}}$ and $\{\mathsf{RO}_\ell\}_{\ell\in\mathbb{N}}$.

In particular, with probability 1, this protocol is a computationally sum-binding commitment scheme relative to $(SG, Mix, PSPACE)$. □

8 $\mathbf{BQP}^{SG,Mix,PSPACE} = \mathbf{QCMA}^{SG,Mix,PSPACE}$

In this section, we show the equivalence of **BQP** and **QCMA** relative to $(SG, Mix, PSPACE)$.

Theorem 8.1. *With probability at least* $1 - \mathsf{negl}$ *over* $\{U_\ell\}_{\ell\in\mathbb{N}}$ *and* $\{\mathsf{RO}_\ell\}_{\ell\in\mathbb{N}}$*,*

$$\mathbf{BQP}^{SG,Mix,PSPACE} = \mathbf{QCMA}^{SG,Mix,PSPACE}. \tag{113}$$

To show it, we use the following lemma.

Lemma 8.2 (Path Recording [MH25])**.** *Let* $n, t \in \mathbb{N}$*. There exists an efficiently implementable isometry* PR *operating on an input register* $\mathbf{A}$ *and a state register* $\mathbf{St}$ *indistinguishable from a Haar random unitary by oracle access. Formally, let* $\mathcal{A}^{(\cdot)}$ *be any* t*-query oracle algorithm. Then*

$$\mathsf{TD}\left(\mathop{\mathbb{E}}_{U\leftarrow\mu_{2^n}}\left[\mathcal{A}^{U,U^\dagger}\right], \mathrm{Tr}_{\mathbf{St}}\left(\mathcal{A}^{PR,PR^\dagger}\right)\right) \leq O\left(\frac{t^2}{2^{n/8}}\right). \tag{114}$$

Proof of Theorem 8.1. Let $\Pi \in \mathbf{QCMA}^{SG,Mix,PSPACE}$, which implies that there exists a $\mathbf{QCMA}$ verifier $\mathsf{Ver}^{SG,Mix,PSPACE}$ which takes as input $x \in \{0,1\}^*$ and a classical witness $w \in \{0,1\}^{q(|x|)}$ where q is a polynomial, with completeness $\frac{99}{100}$ and soundness $\frac{1}{100}$. Without loss of generality, we will assume $q(|x|) \geq |x|$. Let $|x| = n$, and let $T(n)$ be a polynomial upper bound on the running time of Ver on inputs of length n.

We will now describe a QPT algorithm $\mathcal{B}^{SG,Mix,PSPACE}(x)$ such that, with probability $1 - \mathsf{negl}$ over $\{U_\ell\}_{\ell\in\mathbb{N}}$ and $\{\mathsf{RO}_\ell\}_{\ell\in\mathbb{N}}$, $\mathcal{B}$ decides Π on all but finitely many inputs x. In particular, we will show that for all but finitely many $x \in \Pi_{yes}$, $\Pr[\mathcal{B}^{SG,Mix,PSPACE}(x) = 1] \geq \frac{2}{3}$ and for all but finitely many $x \in \Pi_{no}$, $\Pr[\mathcal{B}^{SG,Mix,PSPACE}(x) = 1] \leq \frac{1}{3}$. Let $\delta = \frac{1}{600}$ and let $d = \log(c_d q(|x|)^2 T(|x|)^4 + 2)$ for some sufficiently large constant c_d.

- For all $\ell \in [d]$, $\mathcal{B}$ performs tomography on each U_ℓ, producing estimates U'_ℓ. The estimates are produced as follows:
  - $\mathcal{B}$ runs SG_ℓ on $|0\rangle$ and measures the first register of $\frac{1}{\sqrt{2^\ell}} \sum_{x\in\{0,1\}^\ell\setminus\{0\}} |x\rangle |\phi_x\rangle$. $\mathcal{B}$ repeats this until, $\forall k \in [2^\ell]$, there exist $t > 2^d \cdot \frac{T}{\delta}$ copies of $|\phi_k\rangle$.
  - For every $k \in [2^\ell]$, $\mathcal{B}$ runs the state tomography algorithm from Theorem 2.11 on inputs $n = q(n)^2 \cdot T(n)$ and $\epsilon = \frac{\delta^2}{2^{2\ell}\cdot T(n)^2}$. This produces an estimate $|\phi'_k\rangle$ such that $|\langle\phi_k|\phi'_k\rangle|^2 \geq 1 - \epsilon$ with probability at least $1 - e^{-5q(n)^2T(n)}$.
  - $\mathcal{B}$ defines the map $U'_\ell |k\rangle = |\phi'_k\rangle$ for all $k \in [2^\ell]$.
- Further, for all $\ell \in [d]$, $\mathcal{B}$ learns all functions RO_ℓ by querying SG_ℓ, and produces estimates RO'_ℓ. The estimates are produced as follows:
  - $\mathcal{B}$ runs SG_ℓ on $|0\rangle$ and measures the first register. $\mathcal{B}$ repeats it until, $\forall k \in [2^\ell]$, there exist 2^d copies of $|\phi_k\rangle$. With probability $\geq 1 - e^{-q(n)^2}$, this terminates in time polynomial in n for all ℓ.
  - For all $x, y \in \{0,1\}^\ell$, $\mathcal{B}$ queries $|x\rangle|y\rangle|\phi_x\rangle|0\rangle|0\rangle$ to Mix_ℓ and measures the fourth register to get $\mathsf{RO}_\ell(x, y)$. $\mathcal{B}$ stores the result as $\mathsf{RO}'_\ell(x, y)$.
- Next, $\mathcal{B}$ constructs the following $\mathbf{QCMA}^{PSPACE}$ verifier $\widetilde{\mathsf{Ver}}^{PSPACE}$ which will simulate $\mathsf{Ver}^{SG,Mix,PSPACE}$ by replacing queries to $\{SG_\ell\}_{\ell\in\mathbb{N}}$ and $\{\mathsf{RO}_\ell\}_{\ell\in\mathbb{N}}$ as follows:
  - For all $\ell \in [d]$, every query SG_ℓ and Mix_ℓ make to U_ℓ, RO_ℓ are replaced by queries to U'_ℓ and RO'_ℓ.
  - For all $\ell \in [d+1, T(n)]$ simulate all queries to SG_ℓ, Mix_ℓ with queries to $\mathrm{Sim}^{SG_{\mathrm{Samp},\ell}}_{SG,\ell}, \mathrm{Sim}^{Ver_{\mathrm{Samp},\ell}}_{Mix,\ell}$ with error parameter $\epsilon = \frac{\delta^2}{T(n)}$ with the following modification. Instead of sampling fresh random $\widetilde{U}_\ell$ and $\widetilde{\mathsf{RO}}_\ell$ during initialization, $\widetilde{\mathsf{Ver}}$ will instead simulate these for SG_{Samp} and Ver_{Samp} using path-recording oracles and compressed oracles respectively.
- Finally, consider the $\mathbf{QCMA}^{PSPACE}$ problem $\widetilde{\Pi}$ corresponding to $\widetilde{\mathsf{Ver}}$ with completeness $\frac{2}{3}$ and soundness $\frac{1}{3}$. Since $\mathbf{QCMA}^A \subseteq \mathbf{PSPACE}^A$ for all classical oracles A, and since $\widetilde{\Pi} \in \mathbf{QCMA}^{PSPACE} \subseteq \mathbf{PSPACE}^{PSPACE} = \mathbf{PSPACE}$, $\mathcal{B}$ will then use the $PSPACE$ oracle to decide $\widetilde{\Pi}$, and will output 1 if $x \in \widetilde{\Pi}_{yes}$ and 0 if $x \in \widetilde{\Pi}_{no}$.

We will show via a sequence of hybrids that with probability $1 - n^3$ over $\{U_\ell\}_{\ell\in\mathbb{N}}$ and $\{\mathsf{RO}_\ell\}_{\ell\in\mathbb{N}}$, for all (x, w),

$$\left|\Pr[\widetilde{\mathsf{Ver}}^{PSPACE}(x, w) \to 1] - \Pr[\mathsf{Ver}^{SG,Mix,PSPACE}(x, w) \to 1]\right| \leq \frac{1}{100}. \tag{115}$$

For a verification algorithm V_i, Define $acc(V_i) := \Pr[V_i(w,x) \to 1]$, and let $d = \log(3456|x|T(|x|)^4 + 2)$.

1. Let $V_1 = \mathsf{Ver}^{SG,Mix,PSPACE}$.

2. V_2 : This is the same as V_1 except, for all $\ell \in [d]$, all queries to RO_ℓ are replaced with queries to RO'_ℓ generated by $\mathcal{B}$.

   Claim 8.3. *For all SG, Mix, for all Ver, with probability $\geq 1 - e^{-q(n)^2}$ over the randomness of $\mathcal{B}$, $acc(V_2) = acc(V_1)$.*

   Proof. For all $\ell \in [d]$, $\mathcal{B}$ learns all functions RO_ℓ with no error with probability at least $1 - e^{-q(n)^2}$, so $acc(V_2) = acc(V_1)$. □

3. V_3: This is the same as V_2 except, for all $\ell \in [d]$, all queries to U_ℓ are replaced with queries to U'_ℓ generated by $\mathcal{B}$.

   Claim 8.4. *For all SG, Mix, for all Ver, with probability $\geq 1 - e^{-q(n)^2}$ over the randomness of $\mathcal{B}$, $|acc(V_3) - acc(V_2)| \leq 4\delta$.*

   Proof. From Theorem 2.11, we have that for each $\ell \in [d]$, for each $k \in \{0,1\}^\ell$, with probability $\geq 1 - e^{-5q(n)T(n)}$, $|\langle \phi_x | \phi'_x \rangle|^2 \geq 1 - \epsilon$, with $\epsilon = \frac{\delta^2}{16 \cdot 2^{2\ell} \cdot T^2}$. Taking the union bound over all d, k, we get that with probability $\geq 1 - d \cdot 2^d \cdot e^{-5q(n)T(n)} \geq 1 - e^{-q(n)}$, for all $\ell \in [d], k \in \{0,1\}^d$, $|\langle \phi_x | \phi'_x \rangle|^2 \geq 1 - \epsilon$.

   Let SG'_ℓ and Mix'_ℓ be SG_ℓ and Mix_ℓ but with U_ℓ replaced by U'_ℓ. In particular,

   $$\left| \left(\frac{1}{\sqrt{2^\ell - 1}} \sum_{k \in \{0,1\}^\ell \setminus \{0\}} \langle k | \langle \phi'_k | \right) \left(\frac{1}{\sqrt{2^\ell - 1}} \sum_{k \in \{0,1\}^\ell \setminus \{0\}} | k \rangle | \phi_k \rangle \right) \right|^2 \tag{116}$$

   $$= \frac{1}{2^\ell - 1} \left| \sum_{k \in \{0,1\}^\ell \setminus \{0\}} \langle \phi'_k | \phi_k \rangle \right|^2 \tag{117}$$

   $$\geq \frac{2^\ell - 1}{2^\ell - 1} (1 - \epsilon), \tag{118}$$

   and therefore by Lemma 2.2,

   $$\left\| SG_\ell - SG'_\ell \right\|_\diamond \leq 2\sqrt{1 - \left(1 - \frac{\delta}{T(n)}\right)^2} \leq 4\sqrt{\epsilon}. \tag{119}$$

   Similarly, by Lemma 2.4 and the fact that Mix can be seen as applying the unitary version of the measurement $\{|\phi_k\rangle\langle\phi_k|, I - |\phi_k\rangle\langle\phi_k|\}$ in succession for all k, we get

   $$\left\| Mix_\ell - Mix'_\ell \right\|_\diamond \leq 2 \cdot 2^\ell \cdot \sqrt{\epsilon}. \tag{120}$$

   Thus, applying Theorem 2.1 T times and plugging in $\epsilon = \frac{\delta^2}{16 \cdot 2^{2\ell} \cdot T^2}$ proves the claim. □

4. V_4 : This is the same as V_3 except, for all $\ell \in [d+1, T(n)]$, queries to SG_ℓ, Mix_ℓ are replaced with queries to $\mathrm{Sim}_{SG,\ell}^{SG_{\mathrm{Samp},\ell}}, \mathrm{Sim}_{Mix,\ell}^{Ver_{\mathrm{Samp},\ell}}$ with error parameter $\frac{\delta^2}{T(n)}$.

Claim 8.5. *With probability* $\geq 1 - n^{-4}$ *over the choice of* SG, Mix*, for all* Ver*,* $|acc(V_4) - acc(V_3)| \leq \delta^2$*.*

Proof. This follows immediately from applying Lemma 4.1 for each $\ell \in [d+1, T(n)]$. In particular, for each ℓ, replacing SG_ℓ, Mix_ℓ incurs a loss of $\frac{\delta^2}{T(n)} + O\left(\frac{T(n)^{2.5}+q(n)}{2^{d/8}}\right)$ with probability $\geq 1 - O(q(n) \cdot 2^{-d/4})$ over $\{U_\ell\}_{\ell\in\mathbb{N}}, \{\mathsf{RO}_\ell\}_{\ell\in\mathbb{N}}$, and we repeat this T times. For sufficiently large n, c_d, $O\left(\frac{T(n)^{2.5}+q(n)}{2^{d/8}}\right) \leq \frac{\delta^2}{2}$ and $T(n) \cdot O(q(n) \cdot 2^{-d/4}) \leq n^{-4}$. Note that Lemma 4.1 relativizes, and so we can apply it even for algorithms which query $SG_{\ell'}, Mix_{\ell'}$ for $\ell' \neq \ell$. □

5. V_5 : This is the same as V_4 except for each $\ell \in [d+1, T]$, we replace the unitary sampled by $\mathrm{Sim}_{SG,\ell}^{SG_{\mathrm{Samp},\ell}}$ with the path-recording oracle PR from Lemma 8.2.

   Claim 8.6. *For all* SG, Mix, Ver*,* $|acc(V_5) - acc(V_4)| \leq \delta$*.*

   Proof. This follows directly from Lemma 8.2. In particular, each replacement incurs a loss of $O\left(\frac{T^2}{2^{d/8}}\right) \leq \frac{\delta}{T}$ for all sufficiently large n. And so, the claim follows by repeating for each $\ell \in [d+1, T]$. □

6. V_6: This is the same as V_5, except that for each $\ell \in [d+1, T]$, we replace the random oracle $\widetilde{\mathsf{RO}}_\ell$ sampled by $\mathrm{Sim}_{SG,\ell}^{SG_{\mathrm{Samp},\ell}}$ with a compressed oracle Ctso.

   Claim 8.7. *For all* SG, Mix, Ver*,* $acc(V_6) = acc(V_5)$*.*

   Proof. This follows from Lemma 2.5. □

Hence, with probability $1 - n^{-3}$ over the choice of SG, Mix we have that for all verifiers Ver, with probability $\geq 4\delta$ over the randomness of $\mathcal{B}$, for all inputs $x \in \{0,1\}^n, w \in \{0,1\}^{poly(n)}$,

$$\left|\Pr[\widetilde{\mathsf{Ver}}^{PSPACE}(x, w) \to 1] - \Pr[\mathsf{Ver}^{SG,Mix,PSPACE}(x, w) \to 1]\right| \leq 6\delta \leq \frac{1}{100}. \tag{121}$$

This means that for all $n \in \mathbb{N}$, with probability $1 - n^{-3}$ over SG, Mix, the following holds

1. For all $x \in \{0,1\}^n \cap \Pi_{yes}$, $\Pr[\mathcal{B}^{SG,Mix,PSPACE}(x) \to 1] \geq 1 - 4\delta$
2. For all $x \in \{0,1\}^n \cap \Pi_{no}$, $\Pr[\mathcal{B}^{SG,Mix,PSPACE}(x) \to 1] \leq 4\delta$

Since $\sum_{i=1}^{\infty} i \cdot i^{-3} < \infty$, by the Borel-Cantelli lemma [ÉB09, Can17], we can swap the quantifiers to say that with probability 1 over SG, Mix, for all but finitely many $n \in \mathbb{N}$,

1. For all $x \in \{0,1\}^n \cap \Pi_{yes}$, $\Pr[\mathcal{B}^{SG,Mix,PSPACE}(x) \to 1] \geq 1 - 4\delta$
2. For all $x \in \{0,1\}^n \cap \Pi_{no}$, $\Pr[\mathcal{B}^{SG,Mix,PSPACE}(x) \to 1] \leq 4\delta$

Thus, with probability 1, there exists some $\mathcal{B}'$ which correctly decides Π for all x by hard-coding the points where $\mathcal{B}$ was wrong.

Therefore, for any fixed $\Pi \in \mathbf{QCMA}^{SG,Mix,PSPACE}$, with probability 1 over SG, Mix, $\Pi \in \mathbf{BQP}^{SG,Mix,PSPACE}$.

Taking a union bound over all $\Pi \in \mathbf{QCMA}^{SG,Mix,PSPACE}$, we get that

$$\mathbf{QCMA}^{SG,Mix,PSPACE} = \mathbf{BQP}^{SG,Mix,PSPACE}. \tag{122}$$

□

9 OWPuzz Constructions

9.1 OWPuzzs From QCCC KE

Theorem 9.1. *If there exists QCCC KE, then distributional OWPuzzs exist.*

Proof. Let (A, B) be a QCCC KE protocol. We construct a distributional OWPuzz, Samp, as follows:

1. Run the KE protocol $\mathsf{A} \leftrightarrows \mathsf{B}$, producing a transcript τ as well as outputs a, b.
2. Output $s := \tau, k := a$.

Define A_τ to be the distribution of A's output a conditioned on the transcript being τ. Observe that A_τ is exactly the distribution over k conditioned on $s = \tau$. Let $a_\tau = \mathsf{argmax}_a \Pr[\mathsf{A}_\tau \to a]$. We can see that with probability $\geq \frac{99}{100}$ over τ, $\Pr[\mathsf{A}_\tau \to a_\tau] \geq \frac{99}{100}$. Otherwise, A and B would disagree on the key with probability $\geq \frac{1}{10000}$, which violates correctness. In particular, this means that

$$\Pr\left[\ a = a_\tau \ \ :\ \ \mathsf{A} \leftrightarrows \mathsf{B} \to (\tau, a, b)\ \right] \geq \frac{98}{100}. \tag{123}$$

Let us assume towards contradiction that Samp is not a distributional OWPuzz. Then there exists a QPT adversary $\mathcal{A}$ such that

$$\Delta((\tau, \mathcal{E}(\tau)), (\tau, a)) \leq \frac{1}{100}. \tag{124}$$

Equations (123) and (124) give us

$$\Pr[a_\tau \leftarrow \mathcal{E}(\tau)] \geq \frac{97}{100} \tag{125}$$

since otherwise we could inefficiently distinguish $(\tau, \mathcal{E}(\tau))$ from (τ, a). Applying the union bound to Equations (124) and (125) then gives us

$$\Pr[\mathcal{E}(\tau) = a : \mathsf{A} \leftrightarrows \mathsf{B} \to (\tau, a, b)] \geq \frac{95}{100} \tag{126}$$

and therefore $\mathcal{E}$ is also an attacker for the KE protocol. Since the KE protocol (A, B) is secure, so is the distributional OWPuzz Samp. □

9.2 OWPuzzs From QCCC Commitments

Theorem 9.2. *If QCCC commitment schemes exist, then distributional OWPuzzs exist.*

Proof. Let $(\mathsf{Com}, \mathsf{Rec})$ be any QCCC commitment scheme with an opening stage taking t rounds. We will assume that in each round, the receiver sends one message x_i and receives a response d_i.

We will construct a distributional OWPuzz, Samp, as follows:

1. Sample $i \leftarrow [t]$.
2. Sample $m \leftarrow \{0, 1\}$.
3. Run the committing stage on m, producing a transcript z.
4. Run the first i rounds of the opening stage, producing a transcript $x_1, d_1, \ldots, x_i, d_i$.

5. Output $s := (i, z, x_1, d_1, \ldots, x_i)$ and $k := d_i$.

Assume that Samp is not a distributional OWPuzzs. Then, for all constants $c \geq 1$, there exists an adversary $\mathcal{A}$ such that for infinitely many $n \in \mathbb{N}$,

$$\Delta((i, z, x_1, d_1, \ldots, x_i, d_i), (i, z, x_1, d_1, \ldots, x_i, \mathcal{A}(i, z, x_1, d_1, \ldots, x_i))) \leq n^{-c}. \tag{127}$$

However,

$$\begin{aligned}
&\Delta((i, z, x_1, d_1, \ldots, x_i, d_i), (i, z, x_1, d_1, \ldots, x_i, \mathcal{A}(i, z, x_1, d_1, \ldots, x_i))) && (128)\\
&= \frac{1}{t} \sum_{i \in [t]} \Delta((z, x_1, d_1, \ldots, x_i, d_i), (z, x_1, d_1, \ldots, x_i, \mathcal{A}(i, z, x_1, d_1, \ldots, x_i))) && (129)\\
&\geq \frac{1}{t} \Delta((z, x_1, d_1, \ldots, x_i, d_i), (z, x_1, d_1, \ldots, x_i, \mathcal{A}(i, z, x_1, d_1, \ldots, x_i))) && (130)
\end{aligned}$$

for each $i \in [t]$. Therefore, for each $i \in [t]$,

$$\Delta((z, x_1, d_1, \ldots, x_i, d_i), (z, x_1, d_1, \ldots, x_i, \mathcal{A}(i, z, x_1, d_1, \ldots, x_i))) \leq t \cdot n^{-c}. \tag{131}$$

We can use $\mathcal{A}$ to build an adversary $\mathcal{B}$ that breaks hiding as follows.

1. Run the committing stage, producing transcript z and receiver state $\widetilde{\rho}^0_{\mathsf{Rec}}$.
2. For each $i \in [t]$
   (a) Run $\widetilde{x}_i \leftarrow \mathsf{Rec}(i, z, \widetilde{x}_1, \widetilde{d}_1, \ldots, \widetilde{x}_{i-1}, \widetilde{d}_{i-1}, \widetilde{\rho}^{i-1}_{\mathsf{Rec}})$ to produce the receiver's message in the ith round. Note that this will update $\widetilde{\rho}^{i-1}_{\mathsf{Rec}}$ to a new state, which we will denote $\widetilde{\rho}^{i}_{\mathsf{Rec}}$.
   (b) Run $\widetilde{d}_i \leftarrow \mathcal{A}(i, z, \widetilde{x}_1, \widetilde{d}_1, \ldots, \widetilde{x}_i)$ to simulate the committer's message in the ith round.
3. Produce the receiver's output $m' \leftarrow \mathsf{Rec}(\widetilde{\rho}^t_{\mathsf{Rec}})$, and output m'.

Now we show that such m' is equal to the originally committed message m with high probability, and therefore $\mathcal{B}$ breaks hiding. Intuitively, if m' is another message than m, a malicious committer that runs $\mathcal{A}$ can open $m' \neq m$, which breaks the binding. Moreover, we can also show that the probability that $\mathcal{B}$ outputs $\perp$ is bounded.

More precisely, the argument is the following. Note that an adversarial committer could also simulate its messages in this way. That is, define Com' as follows

1. Let z be the transcript of the committing stage, and let $x_1, d_1, \ldots, x_i$ be the current transcript of the opening stage.
2. Run $d_i \leftarrow \mathcal{A}(i, z, x_1, d_1, \ldots, x_i)$.
3. Output d_i.

Observe that the resulting distribution on the output of Rec when interacting with Com' on a message m is identical to the output of our attacker when run on a transcript z generated from $\mathsf{Com}(m) \rightleftarrows \mathsf{Rec}$. Thus, by binding we have that

$$\Pr[\mathcal{B}(z, \rho_{\mathsf{Rec}}) \to m' \notin \{m, \perp\} : m \leftarrow \{0,1\}, \mathsf{Com}(m) \rightleftarrows \mathsf{Rec} \to (z, \rho_{\mathsf{Rec}})] \leq \mathsf{negl}(n). \tag{132}$$

Consider the mixed state $\mathcal{I}_i = (z, \widetilde{x}_1, \widetilde{d}_1, \ldots, \widetilde{x}_i, \widetilde{d}_i, \widetilde{\rho}^i_{\mathsf{Rec}})$ produced by $\mathcal{B}$ after round i. We will also define the mixed state $\mathcal{J}_i = (z, x_1, d_1, \ldots, x_i, d_i, \rho^i_{\mathsf{Rec}})$ produced by an honest interaction between Com and Rec after i rounds of the opening stage.

We will show by induction that $\|\mathcal{I}_i - \mathcal{J}_i\|_1 \leq itn^{-c}$. The base case is trivial, since $\mathcal{I}_0 = \mathcal{J}_0$. For the inductive step, assume that $\|\mathcal{I}_i - \mathcal{J}_i\|_1 \leq itn^{-c}$. Define a new distribution $\mathcal{I}'_i$ obtained by applying the ith iteration of $\mathcal{B}$ to a sample from $\mathcal{J}_{i-1}$. In particular, $\mathcal{I}'_i = (z, x_1, d_1, \ldots, d_{i-1}, x'_i, d'_i, \rho^i_{\mathsf{Rec}})$ where $(z, x_1, d_1, \ldots, d_{i-1}) \leftarrow \mathcal{J}_{i-1}$, $x'_i = \mathsf{Rec}(\rho^{i-1}_{\mathsf{Rec}})$ (producing ρ^i_{Rec}), and $d'_i = \mathcal{A}(i, z, x_1, \ldots, d_{i-1}, x'_i)$. Since trace distance is contractive under CPTP maps, we have

$$\left\|\mathcal{I}_i - \mathcal{I}'_i\right\|_1 \leq \|\mathcal{I}_{i-1} - \mathcal{J}_{i-1}\|_1 \leq (i-1)tn^{-c}. \tag{133}$$

But note that the internal state of the receiver at any point in time is a deterministic function of the transcript. In particular, ρ^i_{Rec} is the state achieved by running the protocol in superposition and then postselecting on the transcript being $(z, x_1, d_1, \ldots, x_i, d_i)$. Furthermore, x'_i and x_i are sampled identically. Thus,

$$\begin{aligned}
&\left\|\mathcal{I}'_i - \mathcal{J}_i\right\|_1 && (134)\\
&= \left\|(z, x_1, d_1, \ldots, x_i, \mathcal{A}(i, z, x_1, d_1, \ldots, x_i), \rho^i_{\mathsf{Rec}}) - (z, x_1, d_1, \ldots, x_i, d_i, \rho^i_{\mathsf{Rec}})\right\|_1 && (135)\\
&= \|(z, x_1, d_1, \ldots, x_i, \mathcal{A}(i, z, x_1, d_1, \ldots, x_i)) - (z, x_1, d_1, \ldots, x_i, d_i)\|_1 && (136)\\
&= \Delta((z, x_1, d_1, \ldots, x_i, \mathcal{A}(i, z, x_1, d_1, \ldots, x_i)), (z, x_1, d_1, \ldots, x_i, d_i)) \leq tn^{-c}, && (137)
\end{aligned}$$

where the last inequality follows from Equation (131). Combining Equations (133) and (134) gives us

$$\|\mathcal{I}_i - \mathcal{J}_i\|_1 \leq \left\|\mathcal{I}_i - \mathcal{I}'_i\right\|_1 + \left\|\mathcal{I}'_i - \mathcal{J}_i\right\|_1 \leq (i-1)tn^{-c} + tn^{-c} = itn^{-c}. \tag{138}$$

In particular $\|\mathcal{I}_t - \mathcal{J}_t\|_1 \leq t^2 n^{-c}$. But we know that running Rec on the internal state from $\mathcal{J}_t$ will output something which is not $\perp$ with all but negligible probability by correctness. Thus, as long as $n^{-c} \leq \frac{1}{4t^2}$,

$$\Pr[\perp \leftarrow \mathcal{B}] \leq \frac{1}{4} + \mathsf{negl}(n). \tag{139}$$

Putting together Equations (132) and (139), we get

$$\begin{aligned}
\Pr[m \leftarrow \mathcal{B}] &= 1 - \Pr[m \not\leftarrow \mathcal{B}] && (140)\\
&\geq 1 - \Pr[\mathcal{B} \to m' \notin \{m, \perp\}] - \Pr[\perp \leftarrow \mathcal{B}] && (141)\\
&\geq \frac{3}{4} - \mathsf{negl}(n), && (142)
\end{aligned}$$

which gives a contradiction. Thus, Samp is a distributional OWPuzz. □

9.3 2 Round QKD Implies State Puzzles

Definition 9.3 (State Puzzles [KT25])**.** *A state puzzle is defined by a uniform QPT algorithm* $\mathsf{Samp}(1^n)$ *which outputs a pair* $(s, |\psi_s\rangle)$ *of a bit string* s *and a quantum state* $|\psi_s\rangle$ *such that given* s*, it is quantum computationally infeasible to output* ρ *which overlaps notably with* $|\psi_s\rangle$*. Formally, we define* σ_n *to be the mixed state corresponding to the output of* $\mathsf{Samp}(1^n)$*. For a non-uniform QPT* $\mathcal{A}$*, we define* $\sigma_n^{\mathcal{A}}$ *to be the mixed state corresponding to the following process*

$$\begin{aligned}
(s, |\psi_s\rangle) &\leftarrow \mathsf{Samp}(1^n) && (143)\\
\rho &\leftarrow \mathcal{A}(1^n, s) && (144)\\
&\textit{Output } (s, \rho). && (145)
\end{aligned}$$

We say that Samp *is a state puzzle if for all QPT* $\mathcal{A}$*,*

$$\mathsf{TD}(\sigma_n, \sigma_n^{\mathcal{A}}) \geq 1 - \mathsf{negl}(n). \tag{146}$$

Remark 9.4. *Note that this definition at first seems different than the one in [KT25], which says that the expected overlap between* ρ *and* $|\psi_s\rangle$ *is negligible. Formally,*

$$\mathbb{E}\left[\mathrm{Tr}(|\psi_s\rangle\langle\psi_s| \, \rho) \ : \ \begin{matrix}(s, |\psi_s\rangle) \leftarrow \mathsf{Samp}(1^n) \\ \rho \leftarrow \mathcal{A}(1^n, s)\end{matrix} \right] \leq \mathsf{negl}(n). \tag{147}$$

However, these definitions are equivalent. In particular, the best distinguisher between σ *and* $\sigma^{\mathcal{A}}$ *is the map which on input* $(s, |\psi_s\rangle)$ *applies the measurement* $\{|\psi_s\rangle\langle\psi_s|, I - |\psi_s\rangle\langle\psi_s|\}$*. But the advantage of this distinguisher is exactly*

$$1 - \mathbb{E}\left[\mathrm{Tr}(|\psi_s\rangle\langle\psi_s| \, \rho) \ : \ \begin{matrix}(s, |\psi_s\rangle) \leftarrow \mathsf{Samp}(1^n) \\ \rho \leftarrow \mathcal{A}(1^n, s)\end{matrix} \right]. \tag{148}$$

For simplicity of presentation, we will not consider quantum advice. Observing the proof, our result also holds for quantum advice.

Informally, QKD describes a two party protocol where two QPT parties, Alice and Bob, send each other authenticated classical messages and unauthenticated quantum messages that satisfies the following.

1. (Correctness): Either Alice and Bob output the same key, or one of the parties outputs $\bot$.

2. (Security): If either Alice or Bob does not output $\bot$, then Eve cannot learn their final key.

We consider a weaker notion of security, which we call one-sided computational security. In particular, we will only require that if Bob does not output $\bot$, then Eve cannot predict Bob's key. This definition immediately follows from the more standard stronger definition. Since we build OWPuzzs from the weaker definition, it is also possible to build OWPuzzs from the stronger definition.

Definition 9.5 (Modified from [MW24])**.** *A two-round quantum key distribution (2QKD) scheme is defined as a tuple of algorithms* $(\mathsf{QKDFirst}, \mathsf{QKDSecond}, \mathsf{QKDDecode})$ *with the following syntax.*

1. $\mathsf{QKDFirst}(1^n) \to (msg, \mu, st)$*: A QPT algorithm which, on input the security parameter* 1^n*, outputs a message, consisting of a classical component* msg *and a quantum state* μ*, as well as an internal quantum state* st*. We allow* μ *to be a mixed state, and in particular* μ *and* st *may be entangled.*

2. $\mathsf{QKDSecond}(msg, \mu) \to \{(resp, \eta, k), \bot\}$*: A QPT algorithm which, on input the first message* (msg, μ)*, outputs a response consisting of a classical component* $resp$ *and a quantum mixed state* η*, as well as a classical key* $k \in \{0,1\}^n$*. We also allow* $\mathsf{QKDSecond}$ *to output a special symbol* $\bot$*, denoting rejection.*

3. $\mathsf{QKDDecode}(st, resp, \eta) \to \{k, \bot\}$*: A QPT algorithm which, on input the internal state* st *and the response* $(resp, \eta)$*, outputs a key* $k \in \{0,1\}^n$*. We also allow* $\mathsf{QKDDecode}$ *to output a special symbol* $\bot$*, denoting rejection.*

Definition 9.6 (2QKD Correctness)**.** *We say that a 2QKD protocol* $(\mathsf{QKDFirst}, \mathsf{QKDSecond}, \mathsf{QKDDecode})$ *satisfies correctness if there exists a negligible function* negl *such that for all* $n \in \mathbb{N}$*,*

$$\Pr[\bot \leftarrow \mathsf{QKDSecond}(msg, \mu) : (msg, \mu, st) \leftarrow \mathsf{QKDFirst}(1^n)] \leq \mathsf{negl}(n) \tag{149}$$

and

$$\Pr\left[\mathsf{QKDDecode}(st, resp, \eta) = k \ : \ \begin{array}{c} (msg, \mu, st) \leftarrow \mathsf{QKDFirst}(1^n) \\ (resp, \eta, k) \leftarrow \mathsf{QKDSecond}(msg, \mu) \end{array} \right] \geq 1 - \mathsf{negl}(n). \tag{150}$$

Definition 9.7 (2QKD One-sided Computational Security)**.** *For any non-uniform QPT algorithm* $\mathcal{A}$*, we define the following security game* $\mathbf{QKDSec}(\mathcal{A}, 1^n)$ *as follows.*

1. *Run* $(msg, \mu, st) \leftarrow \mathsf{QKDFirst}(1^n)$ *and send* (msg, μ) *to* $\mathcal{A}$*.*
2. $\mathcal{A}$ *returns a quantum state* ρ*, which may be entangled with* $\mathcal{A}$*'s internal state.*
3. *Run* $\{(resp, \eta, k), \bot\} \leftarrow \mathsf{QKDSecond}(msg, \rho)$*.*
4. *Forward the response to* $\mathcal{A}$*.*
5. $\mathcal{A}$ *outputs* $k' \in \{0,1\}^n$*.*
6. *The game outputs* 1 *if* $k' = k \neq \bot$*.*

We say a 2QKD protocol $(\mathsf{QKDFirst}, \mathsf{QKDSecond}, \mathsf{QKDDecode})$ *satisfies one-sided computational security if for any non-uniform QPT algorithm* $\mathcal{A}$*, there exists a negligible function* negl *such that*

$$\Pr[1 \leftarrow \mathbf{QKDSec}(\mathcal{A}, 1^n)] \leq \mathsf{negl}(n). \tag{151}$$

Theorem 9.8. *If there exists a 2QKD protocol satisfying correctness and one-sided computational security, then OWPuzzs exist.*

Proof. We will assume without loss of generality that the combined quantum state μ, st is a single pure state $|\phi\rangle_{\mathbf{BC}}$ over two registers. We can do this by running $\mathsf{QKDFirst}$, deferring all measurements to the end. This will end up with a state over four registers $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ where $\mathbf{A}$ is the output register for msg, $\mathbf{B}$ is the output register for μ, $\mathbf{C}$ is the output register for st, and $\mathbf{D}$ is some ancilla register. We will then set st to be the state over the registers $\mathbf{C}$ and $\mathbf{D}$ combined. $\mathsf{QKDDecode}$ will simply ignore the $\mathbf{D}$ register. This leads to a protocol with identical behavior, but where the state μ, st is pure.

In particular, we will build a state puzzle. Samp will be defined as follows.

1. Run $(msg, |\phi\rangle_{\mathbf{BC}}) \leftarrow \mathsf{QKDFirst}(1^n)$.
2. Output $s = msg$, $|\psi_s\rangle = |\phi\rangle_{\mathbf{BC}}$.

We then claim that Samp is a state puzzle. In particular, let p be a polynomial and let $\mathcal{A}$ be a non-uniform QPT adversary breaking state puzzle security of Samp with advantage $\epsilon(n)$. Formally, for $\sigma_n, \sigma_n^{\mathcal{A}}$ from Definition 9.3

$$\mathsf{TD}(\sigma_n, \sigma_n^{\mathcal{A}}) \leq 1 - \epsilon(n) \tag{152}$$

for infinitely many n.

We will construct an attacker $\mathcal{A}'$ breaking QKD one-sided computational security with advantage dependent on ϵ.

1. On receiving (msg, μ), run $|\phi\rangle_{\mathbf{BC}} \leftarrow \mathcal{A}(1^n, msg)$.
2. Output $|\phi\rangle_{\mathbf{B}}$, keeping internal state $|\phi\rangle_{\mathbf{C}}$.
3. On receiving $(resp, \eta)$, run $k' \leftarrow \mathsf{QKDDecode}(|\phi\rangle_{\mathbf{B}}, resp, \eta)$.
4. Output k'.

We can construct an inefficient adversary $\mathcal{A}''$ which acts the same as $\mathcal{A}'$, but always perfectly inverts the puzzle. That is, $\mathcal{A}''$ is defined as follows.

1. On receiving (msg, μ), output $|\phi_{msg}\rangle$, the residual state of $\mathsf{QKDFirst}$'s quantum output conditioned on the classical message being msg.
2. Output $|\phi_{msg}\rangle_{\mathbf{B}}$, keeping internal state $|\phi_{msg}\rangle_{\mathbf{C}}$.
3. On receiving $(resp, \eta)$, run $k' \leftarrow \mathsf{QKDDecode}(|\phi_{msg}\rangle_{\mathbf{B}}, resp, \eta)$.
4. Output k'.

By the definition of $\mathcal{A}$,

$$\mathsf{TD}((msg, \mathcal{A}(1^n, msg)), (msg, |\phi_{msg}\rangle)) \leq 1 - \epsilon(n) \tag{153}$$

and so since the only difference between $\mathbf{QKDSec}(\mathcal{A}')$ and $\mathbf{QKDSec}(\mathcal{A}'')$ is that $\mathcal{A}(msg) = |\phi\rangle$ is replaced by $|\phi_{msg}\rangle$, we get

$$\left|\Pr[1 \leftarrow \mathbf{QKDSec}(\mathcal{A}'', 1^n)] - \Pr[1 \leftarrow \mathbf{QKDSec}(\mathcal{A}', 1^n)]\right| \leq 1 - \epsilon(n). \tag{154}$$

But we can also observe that $\mathbf{QKDSec}(\mathcal{A}'', 1^n)$ simply runs the honest QKD protocol, with $\mathcal{A}''$ running $\mathsf{QKDDecode}$ on the correct state. Thus, by correctness, we have that in $\mathbf{QKDSec}(\mathcal{A}'', 1^n)$, the following hold

$$\Pr[k = \bot] \leq \mathsf{negl}(n) \tag{155}$$
$$\Pr[k' = k] \geq 1 - \mathsf{negl}(n). \tag{156}$$

Thus,

$$\Pr[1 \leftarrow \mathbf{QKDSec}(\mathcal{A}'', 1^n)] \geq 1 - \mathsf{negl}(n). \tag{157}$$

Equations (154) and (157) together give us

$$\Pr[1 \leftarrow \mathbf{QKDSec}(\mathcal{A}', 1^n)] \geq \epsilon(n) - \mathsf{negl}(n) \tag{158}$$

and therefore by 2QKD security, $\epsilon(n) \leq \mathsf{negl}(n)$. □

Theorem 9.9 (Corollary of Theorem 6.1)**.** *There exists an oracle relative to which 2QKD exists, but* $\mathbf{BQP} = \mathbf{QCMA}$.

Proof. This oracle is exactly the same as the QCCC KE oracle. In particular, we observe that our QCCC KE protocol consists of exactly two rounds. But any 2 round QCCC KE protocol is trivially also a 2QKD protocol. Therefore the theorem follows. □

Acknowledgement

Eli Goldin is supported by a NSF Graduate Student Research Fellowship. Tomoyuki Morimae is supported by JST CREST JPMJCR23I3, JST Moonshot JPMJMS20615-1-1, JST FOREST, MEXT QLEAP, the Grant-in Aid for Transformative Research Areas (A) 21H05183, and the Grant-in-Aid for Scientific Research (A) No.22H00522.

References

[AC12] Scott Aaronson and Paul Christiano. Quantum money from hidden subspaces. In Howard J. Karloff and Toniann Pitassi, editors, *44th Annual ACM Symposium on Theory of Computing*, pages 41–60, New York, NY, USA, May 19–22, 2012. ACM Press.

[ACC+22] Per Austrin, Hao Chung, Kai-Min Chung, Shiuan Fu, Yao-Ting Lin, and Mohammad Mahmoody. On the impossibility of key agreements from quantum random oracles. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology – CRYPTO 2022, Part II*, volume 13508 of *Lecture Notes in Computer Science*, pages 165–194, Santa Barbara, CA, USA, August 15–18, 2022. Springer, Cham, Switzerland.

[AG25] Prabhanjan Ananth and Eli Goldin. Less is more: On copy complexity in quantum cryptography. To appear, 2025.

[AGKL24] Prabhanjan Ananth, Aditya Gulati, Fatih Kaleoglu, and Yao-Ting Lin. Pseudorandom isometries. In Marc Joye and Gregor Leander, editors, *Advances in Cryptology – EUROCRYPT 2024, Part IV*, volume 14654 of *Lecture Notes in Computer Science*, pages 226–254, Zurich, Switzerland, May 26–30, 2024. Springer, Cham, Switzerland.

[AGL24] Prabhanjan Ananth, Aditya Gulati, and Yao-Ting Lin. Cryptography in the common Haar state model: Feasibility results and separations. Cryptology ePrint Archive, Report 2024/1043, 2024.

[AHU18] Andris Ambainis, Mike Hamburg, and Dominique Unruh. Quantum security proofs using semi-classical oracles. Cryptology ePrint Archive, Paper 2018/904, 2018.

[AQY22] Prabhanjan Ananth, Luowen Qian, and Henry Yuen. Cryptography from pseudorandom quantum states. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology – CRYPTO 2022, Part I*, volume 13507 of *Lecture Notes in Computer Science*, pages 208–236, Santa Barbara, CA, USA, August 15–18, 2022. Springer, Cham, Switzerland.

[BB14] Charles H. Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. *Theoretical Computer Science*, 560:7–11, dec 2014.

[BCG90] Mihir Bellare, Lenore Cowen, and Shafi Goldwasser. On the structure of secret key exchange protocols (rump session). In Gilles Brassard, editor, *Advances in Cryptology – CRYPTO'89*, volume 435 of *Lecture Notes in Computer Science*, pages 604–605, Santa Barbara, CA, USA, August 20–24, 1990. Springer, New York, USA.

[BCN24] John Bostanci, Boyang Chen, and Barak Nehoran. Oracle separation between quantum commitments and quantum one-wayness. arXiv, 2024, 2410.03358.

[BCQ23] Zvika Brakerski, Ran Canetti, and Luowen Qian. On the computational hardness needed for quantum cryptography. In Yael Tauman Kalai, editor, *ITCS 2023: 14th Innovations in Theoretical Computer Science Conference*, volume 251, pages 24:1–24:21, Cambridge, MA, USA, January 10–13, 2023. LIPIcs.

[BMM+24] Amit Behera, Giulio Malavolta, Tomoyuki Morimae, Tamer Mour, and Takashi Yamakawa. A new world in the depths of microcrypt: Separating OWSGs and quantum money from QEFID. Cryptology ePrint Archive, Paper 2024/1567, 2024.

[BS19] Zvika Brakerski and Omri Shmueli. (Pseudo) random quantum states with binary phase. In Dennis Hofheinz and Alon Rosen, editors, *TCC 2019: 17th Theory of Cryptography Conference, Part I*, volume 11891 of *Lecture Notes in Computer Science*, pages 229–250, Nuremberg, Germany, December 1–5, 2019. Springer, Cham, Switzerland.

[Can17] Francesco Cantelli. Sulla probabilista come limita della frequencza. *Rend. Accad. Lincei*, 26:39, 1917.

[CCS24] Boyang Chen, Andrea Coladangelo, and Or Sattath. The power of a single haar random state: constructing and separating quantum pseudorandomness. arXiv, 2024, 2404.03295.

[CDGS18] Sandro Coretti, Yevgeniy Dodis, Siyao Guo, and John P. Steinberger. Random oracles and non-uniformity. In Jesper Buus Nielsen and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2018, Part I*, volume 10820 of *Lecture Notes in Computer Science*, pages 227–258, Tel Aviv, Israel, April 29 – May 3, 2018. Springer, Cham, Switzerland.

[CGG+23] Bruno Cavalar, Eli Goldin, Matthew Gray, Peter Hall, Yanyi Liu, and Angelos Pelecanos. On the computational hardness of quantum one-wayness. arXiv, 2023, 2312.08363.

[CGG24] Kai-Min Chung, Eli Goldin, and Matthew Gray. On central primitives for quantum cryptography with classical communication. In Leonid Reyzin and Douglas Stebila, editors, *Advances in Cryptology – CRYPTO 2024, Part VII*, volume 14926 of *Lecture Notes in Computer Science*, pages 215–248, Santa Barbara, CA, USA, August 18–22, 2024. Springer, Cham, Switzerland.

[ÉB09] M. Émile Borel. Les probabilités dénombrables et leurs applications arithmétiques. *Rendiconti del Circolo Matematico di Palermo (1884-1940)*, 27(1):247–271, Dec 1909.

[GKKT20] M Guţă, J Kahn, R Kueng, and J A Tropp. Fast state tomography with optimal error bounds. *Journal of Physics A: Mathematical and Theoretical*, 53(20):204001, apr 2020.

[GLLZ21] Siyao Guo, Qian Li, Qipeng Liu, and Jiapeng Zhang. Unifying presampling via concentration bounds. In Kobbi Nissim and Brent Waters, editors, *TCC 2021: 19th Theory of Cryptography Conference, Part I*, volume 13042 of *Lecture Notes in Computer Science*, pages 177–208, Raleigh, NC, USA, November 8–11, 2021. Springer, Cham, Switzerland.

[GYZ17] Sumegha Garg, Henry Yuen, and Mark Zhandry. New security notions and feasibility results for authentication of quantum data. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology – CRYPTO 2017, Part II*, volume 10402 of *Lecture Notes in Computer Science*, pages 342–371, Santa Barbara, CA, USA, August 20–24, 2017. Springer, Cham, Switzerland.

[GZ25] Eli Goldin and Mark Zhandry. Translating between the common haar random state model and the unitary model. In Yael Tauman Kalai and Seny F. Kamara, editors, *Advances in Cryptology – CRYPTO 2025*, pages 269–300, Cham, 2025. Springer Nature Switzerland.

[HKOT23] Jeongwan Haah, Robin Kothari, Ryan O'Donnell, and Ewin Tang. Query-optimal estimation of unitary channels in diamond distance. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 363–390, 2023.

[HR07] Iftach Haitner and Omer Reingold. Statistically-hiding commitment from any one-way function. In David S. Johnson and Uriel Feige, editors, *39th Annual ACM Symposium on Theory of Computing*, pages 1–10, San Diego, CA, USA, June 11–13, 2007. ACM Press.

[IL89] Russell Impagliazzo and Michael Luby. One-way functions are essential for complexity based cryptography (extended abstract). In *30th Annual Symposium on Foundations of Computer Science*, pages 230–235, Research Triangle Park, NC, USA, October 30 – November 1, 1989. IEEE Computer Society Press.

[INN+22] Sandy Irani, Anand Natarajan, Chinmay Nirkhe, Sujit Rao, and Henry Yuen. Quantum Search-To-Decision Reductions and the State Synthesis Problem. In Shachar Lovett, editor, *37th Computational Complexity Conference (CCC 2022)*, volume 234 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 5:1–5:19, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[IR89] Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *21st Annual ACM Symposium on Theory of Computing*, pages 44–61, Seattle, WA, USA, May 15–17, 1989. ACM Press.

[JLS18] Zhengfeng Ji, Yi-Kai Liu, and Fang Song. Pseudorandom quantum states. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology – CRYPTO 2018, Part III*, volume 10993 of *Lecture Notes in Computer Science*, pages 126–152, Santa Barbara, CA, USA, August 19–23, 2018. Springer, Cham, Switzerland.

[KMNY24] Fuyuki Kitagawa, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa. Quantum public-key encryption with tamper-resilient public keys from one-way functions. In Leonid Reyzin and Douglas Stebila, editors, *Advances in Cryptology – CRYPTO 2024, Part VII*, volume 14926 of *Lecture Notes in Computer Science*, pages 93–125, Santa Barbara, CA, USA, August 18–22, 2024. Springer, Cham, Switzerland.

[KNY23] Fuyuki Kitagawa, Ryo Nishimaki, and Takashi Yamakawa. Publicly verifiable deletion from minimal assumptions. In Guy N. Rothblum and Hoeteck Wee, editors, *TCC 2023: 21st Theory of Cryptography Conference, Part IV*, volume 14372 of *Lecture Notes in Computer Science*, pages 228–245, Taipei, Taiwan, November 29 – December 2, 2023. Springer, Cham, Switzerland.

[KQST23] William Kretschmer, Luowen Qian, Makrand Sinha, and Avishay Tal. Quantum cryptography in algorithmica. In Barna Saha and Rocco A. Servedio,

editors, *55th Annual ACM Symposium on Theory of Computing*, pages 1589–1602, Orlando, FL, USA, June 20–23, 2023. ACM Press.

[KQT25] William Kretschmer, Luowen Qian, and Avishay Tal. Quantum-computable one-way functions without one-way functions. In *57th Annual ACM Symposium on Theory of Computing*, pages 189–200. ACM Press, June 2025.

[Kre21] William Kretschmer. Quantum Pseudorandomness and Classical Complexity. In Min-Hsiu Hsieh, editor, *16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021)*, volume 197 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 2:1–2:20, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[KT24] Dakshita Khurana and Kabir Tomer. Commitments from quantum one-wayness. In Bojan Mohar, Igor Shinkar, and Ryan O'Donnell, editors, *56th Annual ACM Symposium on Theory of Computing*, pages 968–978, Vancouver, BC, Canada, June 24–28, 2024. ACM Press.

[KT25] Dakshita Khurana and Kabir Tomer. Founding quantum cryptography on quantum advantage, or, towards cryptography from #p hardness. In *57th Annual ACM Symposium on Theory of Computing*, pages 178–188. ACM Press, June 2025.

[Kuk18] Ryszard Kukulski. Distinguishability of quantum measurements. Presented at Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, 2018.

[LMW24] Alex Lombardi, Fermi Ma, and John Wright. A one-query lower bound for unitary synthesis and breaking quantum cryptography. In Bojan Mohar, Igor Shinkar, and Ryan O'Donnell, editors, *56th Annual ACM Symposium on Theory of Computing*, pages 979–990, Vancouver, BC, Canada, June 24–28, 2024. ACM Press.

[Mec19] Elizabeth S. Meckes. *The Random Matrix Theory of the Classical Compact Groups*. Cambridge Tracts in Mathematics. Cambridge University Press, 2019.

[MH25] Fermi Ma and Hsin-Yuan Huang. How to construct random unitaries. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, STOC '25, page 806–809, New York, NY, USA, 2025. Association for Computing Machinery.

[MPY23] Tomoyuki Morimae, Alexander Poremba, and Takashi Yamakawa. Revocable quantum digital signatures. Cryptology ePrint Archive, Report 2023/1937, 2023.

[MW24] Giulio Malavolta and Michael Walter. Robust quantum public-key encryption with applications to quantum key distribution. In Leonid Reyzin and Douglas Stebila, editors, *Advances in Cryptology – CRYPTO 2024, Part VII*, volume 14926 of *Lecture Notes in Computer Science*, pages 126–151, Santa Barbara, CA, USA, August 18–22, 2024. Springer, Cham, Switzerland.

[MY22] Tomoyuki Morimae and Takashi Yamakawa. Quantum commitments and signatures without one-way functions. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology – CRYPTO 2022, Part I*, volume 13507

of *Lecture Notes in Computer Science*, pages 269–295, Santa Barbara, CA, USA, August 15–18, 2022. Springer, Cham, Switzerland.

[MY24] Tomoyuki Morimae and Takashi Yamakawa. One-wayness in quantum cryptography. In Frédéric Magniez and Alex Bredariol Grilo, editors, *19th Conference on the Theory of Quantum Computation, Communication and Cryptography, TQC 2024, September 9-13, 2024, Okinawa, Japan*, volume 310 of *LIPIcs*, pages 4:1–4:21. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024.

[NC11] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, USA, 10th edition, 2011.

[QRZ25] Luowen Qian, Justin Raizes, and Mark Zhandry. Hard quantum extrapolations in quantum cryptography. In *Advances in Cryptology – EUROCRYPT 2025: 44th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Madrid, Spain, May 4–8, 2025, Proceedings, Part VII*, page 53–81, Berlin, Heidelberg, 2025. Springer-Verlag.

[Unr07] Dominique Unruh. Random oracles and auxiliary input. In Alfred Menezes, editor, *Advances in Cryptology – CRYPTO 2007*, volume 4622 of *Lecture Notes in Computer Science*, pages 205–223, Santa Barbara, CA, USA, August 19–23, 2007. Springer, Berlin, Heidelberg, Germany.

[Unr16] Dominique Unruh. Computationally binding quantum commitments. In Marc Fischlin and Jean-Sébastien Coron, editors, *Advances in Cryptology – EUROCRYPT 2016, Part II*, volume 9666 of *Lecture Notes in Computer Science*, pages 497–527, Vienna, Austria, May 8–12, 2016. Springer, Berlin, Heidelberg, Germany.

[Vaz98] Umesh Vazirani. On the power of quantum computation. 1998.

[Wie83] Stephen Wiesner. Conjugate coding. *SIGACT News*, 15(1):78–88, January 1983.

[Yan22] Jun Yan. General properties of quantum bit commitments (extended abstract). In Shweta Agrawal and Dongdai Lin, editors, *Advances in Cryptology – ASIACRYPT 2022, Part IV*, volume 13794 of *Lecture Notes in Computer Science*, pages 628–657, Taipei, Taiwan, December 5–9, 2022. Springer, Cham, Switzerland.

[Zha19] Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology – CRYPTO 2019, Part II*, volume 11693 of *Lecture Notes in Computer Science*, pages 239–268, Santa Barbara, CA, USA, August 18–22, 2019. Springer, Cham, Switzerland.

[Zha21] Mark Zhandry. Quantum lightning never strikes the same state twice. or: Quantum money from cryptographic assumptions. *Journal of Cryptology*, 34(1):6, January 2021.

[Zha24] Mark Zhandry. The space-time cost of purifying quantum computations. In Venkatesan Guruswami, editor, *ITCS 2024: 15th Innovations in Theoretical*

Computer Science Conference, volume 287, pages 102:1–102:22, Berkeley, CA, USA, January 30 – February 2, 2024. LIPIcs.

.1 Proof of Lemma 4.3

Here we give a proof of Lemma 4.3, which is used to show Lemma 4.1. Lemma 4.3 is essentially a generalization of Lemma 4.4 from [GZ25]. To our knowledge, the idea that superpositions of states with a random phase can be simulated using copies of the states first appeared in [Zha24] (see Lemma 5.5). A generalization of this technique also appears in a concurrent (but not independent) work [AG25]. For completeness, we include a full proof along with an explicit description of the simulator.

Lemma .10 (Lemma 4.3 restated)**.** *There exists an efficient oracle algorithm* $\mathrm{Sim}^{(\cdot)}(1^t)$ *such that the following is true: Let* n *be any natural number. Let* $\{|\phi_k\rangle\}_{k\in\{0,1\}^n}$ *be any collection of states. For a function* $f : \{0,1\}^n \to \{0,1\}^n = [2^n]$*, define the state*

$$|\phi^f\rangle := \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2^n-1}}\sum_{k\in\{0,1\}^n\setminus\{0\}} \omega_{2^n}^{f(k)}|k\rangle|\phi_k\rangle\,. \tag{159}$$

Let $Gen^{\{|\phi_k\rangle\}_k}$ *be the CPTP oracle which acts as follows*

1. *Sample* $k \leftarrow \{0,1\}^n \setminus \{0\}$.
2. *Output* $|k\rangle|\phi_k\rangle$.

Then for all t*, we have*

$$\mathop{\mathbb{E}}_f\left[|\phi^f\rangle\langle\phi^f|^{\otimes t}\right] = \mathrm{Sim}^{Gen^{\{|\phi_k\rangle\}_k}}(1^t). \tag{160}$$

Proof. For ease of notation, we will write $|\phi_0\rangle = |0\rangle$ and $f(0) = 0$.

$\mathrm{Sim}(1^t)$ will be defined as follows

1. Sample $\eta \leftarrow Bin(t, 1/2)$ where $Bin(t, 1/2)$ is the binomial distribution. That is,
$$\Pr[Bin(t,1/2) = \eta] = \frac{\binom{t}{\eta}}{2^t}. \tag{161}$$
2. Query Gen a total of η times, receiving output states $|k_1\rangle|\phi_{k_1}\rangle\ldots|k_\eta\rangle|\phi_{k_\eta}\rangle$.
3. For $i > \eta$, set $k_i = 0$. For ease of notation, we will define $|\phi_0\rangle = |0\rangle$.
4. Generate the state
$$\left(\sum_{\pi\in Sym(t)}\bigotimes_{i=1}^{t}|k_{\pi(i)}\rangle_{\mathbf{A_i}}|0\rangle_{\mathbf{B_i}}\right)\otimes\left(\bigotimes_{j=1}^{t}|k_j\rangle_{\mathbf{C_j}}|\phi_{k_j}\rangle_{\mathbf{D_j}}\right). \tag{162}$$
5. For all i, j, apply $SWAP_{\mathbf{B_i},\mathbf{D_j}}$ controlled on registers $\mathbf{A_i}, \mathbf{C_j}$ containing the same value in the standard basis. This results in state
$$\left(\sum_{\pi\in Sym(t)}\bigotimes_{i=1}^{t}|k_{\pi(i)}\rangle_{\mathbf{A_i}}|\phi_{k_{\pi(i)}}\rangle_{\mathbf{B_i}}\right)\otimes\left(\bigotimes_{j=1}^{t}|k_j\rangle_{\mathbf{C_j}}|0\rangle_{\mathbf{D_j}}\right). \tag{163}$$

6. Finally, output registers $\mathbf{A_1B_1}\dots\mathbf{A_tB_t}$. This will contain the state

$$\sum_{\pi\in Sym(t)}\bigotimes_{i=1}^{t}|k_{\pi(i)}\rangle_{\mathbf{A_i}}|\phi_{k_{\pi(i)}}\rangle_{\mathbf{B_i}}. \tag{164}$$

For a vector $\vec{k}\in(\{0,1\}^n)^t$, define $type(\vec{k})\in[2^n]^{2^n}$ to be the vector defined by

$$type(\vec{k})_i \coloneqq \#\text{ times } i \text{ appears in } \vec{k}. \tag{165}$$

We will denote $|T|$ by the number of $\vec{k}$ such that $type(\vec{k})=T$. and

$$types \coloneqq \left\{T\in[2^n]^{2^n} : \sum_i T_i = t\right\}. \tag{166}$$

We will then define

$$|\psi_T\rangle \coloneqq \frac{1}{\sqrt{|T|}}\sum_{\vec{k}\in T}\bigotimes_{j=1}^{t}|k_j\rangle|\phi_{k_j}\rangle. \tag{167}$$

Note that after sampling $\vec{k}$, $\mathrm{Sim}^{Gen}(1^t)$ outputs $|\psi_{type(\vec{k})}\rangle$.

Note that for any given T, the probability that $c=T_0$ is chosen is $\frac{\binom{t}{t-T_0}}{2^t}$. After this, the probability that $k_1,\dots,k_c$ are chosen such that $(k_1,\dots,k_c,0,\dots,0)\in T$ is

$$\frac{\frac{|T|}{\binom{t}{t-T_0}}}{(2^n-1)^{t-T_0}}. \tag{168}$$

This is because $T/\binom{t}{t-T_0}$ is the number of vectors in T which begin with $t-T_0$ non-zeros.

In particular, consider the distribution over types (i.e. elements of $[2^n]^{2^n}$ with total weight t) defined by

$$\Pr[\mathcal{D}\to T] = \frac{\binom{t}{t-T_0}}{2^t}\cdot\frac{\frac{|T|}{\binom{t}{t-T_0}}}{(2^n-1)^{t-T_0}} = \frac{|T|}{2^t\cdot(2^n-1)^{t-T_0}}. \tag{169}$$

$\mathrm{Sim}^{Gen}(1^t)$ outputs the state

$$\mathop{\mathbb{E}}_{T\leftarrow\mathcal{D}}[|\phi_T\rangle\langle\phi_T|]. \tag{170}$$

We will now take a closer look at the state $|\phi^f\rangle^{\otimes t}$.

$$\begin{aligned}|\phi^f\rangle^{\otimes t} = &\left(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2^n-1}}\sum_{k\in\{0,1\}^n\setminus\{0\}}\omega_{2^n}^{f(x)}|k\rangle|\phi_k\rangle\right)^{\otimes t}\\ &\sum_{\vec{k}\in\{0,1\}^n}\left(\frac{1}{\sqrt{2}}\right)^t\left(\frac{-1}{\sqrt{2^n-1}}\right)^{t-type(\vec{k})_0}\bigotimes_{j=1}^{t}\omega_{2^n}^{f(k_j)}|k_j\rangle|\phi_{k_j}\rangle\\ &\sum_{\vec{k}\in\{0,1\}^n}\left(\frac{1}{\sqrt{2}}\right)^t\left(\frac{-1}{\sqrt{2^n-1}}\right)^{t-type(\vec{k})_0}\omega_{2^n}^{\sum_{j=1}^t f(k_j)}\bigotimes_{j=1}^{t}|k_j\rangle|\phi_{k_j}\rangle.\end{aligned} \tag{171}$$

But note that $\sum_{j=1}^{t} f(k_j) = f \cdot type(k_j)$ where f is viewed as a vector in $[2^n]^{2^n}$.

And so

$$|\phi^f\rangle^{\otimes t} = \sum_{T \in types} \omega_{2^n}^{f \cdot T} \left(\frac{1}{\sqrt{2}}\right)^t \left(\frac{-1}{\sqrt{2^n-1}}\right)^{t-T_0} \sum_{\vec{k} \in T} \bigotimes_{j=1}^{t} |k_j\rangle |\phi_{k_j}\rangle \tag{172}$$

$$= \sum_{T \in types} \omega_{2^n}^{f \cdot T} (-1)^{t-T_0} \left(\sqrt{\frac{|T|}{2^t \cdot (2^n-1)^{t-T_0}}}\right) |\psi_T\rangle . \tag{173}$$

We can then explicitly compute

$$\mathop{\mathbb{E}}_{f} \left[|\psi^f\rangle\langle\psi^f|^{\otimes t}\right] \tag{174}$$

$$= \frac{1}{(2^n)^{2^n}} \sum_{f,T,T' \in types} \omega_{2^n}^{f \cdot T - f \cdot T'} (-1)^{T_0+T_0'} \sqrt{\frac{|T|}{2^t \cdot (2^n-1)^{t-T_0}}} \sqrt{\frac{|T'|}{2^t \cdot (2^n-1)^{t-T_0'}}} |\psi_T\rangle \langle\psi_{T'}| \tag{175}$$

$$= \frac{1}{(2^n)^{2^n}} \sum_{f,T \in types} \left(\sum_f \omega_{2^n}^{f \cdot 0}\right) \frac{|T|}{2^t \cdot (2^n-1)^{t-T_0}} |\psi_T\rangle\langle\psi_T| \tag{176}$$

$$+ \frac{1}{(2^n)^{2^n}} \sum_{T \neq T' \in types} \left(\sum_f \omega_{2^n}^{f \cdot (T-T')}\right) (-1)^{T_0+T_0'} \sqrt{\frac{|T| \cdot |T'|}{2^{2t} \cdot (2^n-1)^{2t-T_0-T_0'}}} |\phi_T\rangle \langle\psi_{T'}| . \tag{177}$$

But when $T - T' \neq 0$,

$$\sum_f \omega_{2^n}^{f \cdot (T-T')} = 0. \tag{178}$$

and so we finally can compute

$$\mathop{\mathbb{E}}_{f} \left[\left|\phi^f\right\rangle\left\langle\phi^f\right|^{\otimes t}\right] = \sum_{T \in types} \frac{|T|}{2^t \cdot (2^n-1)^{t-T_0}} |\psi_T\rangle\langle\psi_T| \tag{179}$$

$$= \mathop{\mathbb{E}}_{T \leftarrow \mathcal{D}} [|\psi_T\rangle\langle\psi_T|] \tag{180}$$

$$= \mathrm{Sim}^{Gen}(1^t). \tag{181}$$

□

A Proof of Lemma 2.24

Lemma A.1 (Lemma 2.24, restated)**.** *Let $N, N > 0$ be integers with $N \leq M$. Let ρ be the mixed state of dimension NM defined by sampling N states $|\phi_k\rangle \leftarrow \mu_M^s$ for $k \in [N]$. That is,*

$$\rho = \mathop{\mathbb{E}}_{|\phi_k\rangle \leftarrow \mu_M^s, k \in [N]} \left[\bigotimes_{k=1}^{N} |\phi_k\rangle\langle\phi_k|\right]. \tag{182}$$

Let ρ' be the same state but where we require that each $|\phi_k\rangle, |\phi_{k'}\rangle$ are orthogonal. Formally, ρ' is the mixed state of dimension NM defined as

$$\rho' = \mathop{\mathbb{E}}_{U \leftarrow \mu_M} \left[\bigotimes_{k=1}^{N} U|k\rangle\langle k|U^\dagger \right]. \tag{183}$$

Then $\mathsf{TD}(\rho, \rho') \leq \frac{N(N+1)}{2M}$.

Proof. Define ρ_i to be the state defined by sampling the first k states to all be orthogonal and the last $N - i$ states uniformly. Formally,

$$\rho_i = \mathop{\mathbb{E}}_{U \leftarrow \mu_M, |\phi_k\rangle \leftarrow \mu_M^s} \left(\bigotimes_{k=1}^{i} U|k\rangle\langle k|U^\dagger \right) \otimes \left(\bigotimes_{k=i+1}^{N} |\phi_k\rangle\langle\phi_k| \right). \tag{184}$$

Here $\rho_0 = \rho'$ and $\rho_n = \rho$. We will show that for all i, $\mathsf{TD}(\rho_i, \rho_{i+1}) \leq \frac{i}{M}$. In particular, define

$$S_i = \left\{ \bigotimes_{k=1}^{N} |\psi_k\rangle : \text{for all } j, j' \leq i, \langle\psi_j|\psi_{j'}\rangle = 0 \right\}. \tag{185}$$

Define Π_i to be the projector onto the subspace spanned by S_i. We observe that $\Pi_{i+1}\rho_i$ is proportional to ρ_{i+1}. That is, $I - \Pi_{i+1}$ is the measurement which optimally distinguishes ρ_i and ρ_{i+1}. In particular,

$$\mathsf{TD}(\rho_i, \rho_{i+1}) \leq \mathrm{Tr}((I - \Pi_{i+1})(\rho_i - \rho_{i+1})) = 1 - \mathrm{Tr}(\Pi_{i+1}\rho_i). \tag{186}$$

But $\mathrm{Tr}(\Pi_{i+1}\rho_i)$ is the probability that applying $\{\sum_{k \leq i} |\psi_k\rangle\langle\psi_k|, I - \sum_{k<i} |\psi_k\rangle\langle\psi_k|\}$ to ρ_i results in the first measurement outcome (here $|\psi_k\rangle = U|k\rangle$). So

$$\begin{aligned}
\mathrm{Tr}(\Pi_{i+1}\rho_i) &= \mathbb{E}\left[\sum_{k \leq i} |\langle\psi_{i+1}| U |k\rangle|^2 \ : \ \begin{matrix} U \leftarrow \mu_m \\ |\psi_{i+1}\rangle \leftarrow \mu_M^s \end{matrix} \right] && (187)\\
&\leq \sum_{k \leq i} \mathbb{E}\left[|\langle\psi_{i+1}| U |k\rangle|^2 \ : \ \begin{matrix} U \leftarrow \mu_m \\ |\psi_{i+1}\rangle \leftarrow \mu_M^s \end{matrix} \right] && (188)\\
&\leq i \cdot \mathbb{E}\left[|\langle\psi|\phi\rangle|^2 : |\psi\rangle, |\phi\rangle \leftarrow \mu_M^s \right] && (189)\\
&= \frac{i}{M}. && (190)
\end{aligned}$$

And so, we get that

$$\begin{aligned}
\mathsf{TD}(\rho, \rho') &\leq \sum_{i=1}^{N} \frac{i}{M} && (191)\\
&= \frac{N(N+1)}{2M}. && (192)
\end{aligned}$$

□