Pseudorandom Strings from Pseudorandom Quantum States

Prabhanjan Ananth* — UCSB

Yao-Ting Lin† — UCSB

Henry Yuen‡ — Columbia University

Abstract

A fundamental result in classical cryptography is that pseudorandom generators are equivalent to one-way functions and in fact implied by nearly every classical cryptographic primitive requiring computational assumptions. In this work, we consider a variant of pseudorandom generators called *quantum pseudorandom generators (QPRGs)*, which are quantum algorithms that (pseudo)deterministically map short random seeds to long pseudorandom strings. We provide evidence that QPRGs can be as useful as PRGs by providing cryptographic applications of QPRGs such as commitments and encryption schemes.

Our main result is showing that QPRGs can be constructed assuming the existence of logarithmic-length quantum pseudorandom states. This raises the possibility of basing QPRGs on assumptions weaker than one-way functions. We also consider quantum pseudorandom functions (QPRFs) and show that QPRFs can be based on the existence of logarithmic-length pseudorandom function-like states.

Our primary technical contribution is a method for pseudodeterministically extracting uniformly random strings from Haar-random states.

*prabhanjan@cs.ucsb.edu

†yao-ting_lin@ucsb.edu

‡hyuen@cs.columbia.edu

Contents

1 Introduction

Deterministically generating long pseudorandom strings from a few random bits is a fundamental task in classical cryptography. Pseudorandom generators (PRGs) are a primitive that achieves this task and are ubiquitous throughout cryptography. Beyond cryptography, pseudorandom generators have found applications in complexity theory [RR94, LP20] and derandomization [NW94, IW97]. Since proving complexity separations such as $\mathsf{P} \neq \mathsf{NP}$ appear to be far beyond the reach of present-day techniques, the existence of PRGs must currently be based on some computational assumption. Given the widespread use of PRGs in cryptography and beyond, it is important to understand what are the *weakest possible* (plausible) assumptions for the existence of PRGs.

By the equivalence with one-way functions (OWFs) [HILL99] and pseudorandom functions (PRFs) [GGM86], PRGs can be used to construct digital signatures, private-key encryption, bit commitments, and many other functionalities that comprise "Minicrypt" from Impagliazzo's Five Worlds [Imp95]. This equivalence also gives a precise sense in which PRGs constitute a *minimal assumption* in classical cryptography: virtually every cryptosystem requiring computational assumptions implies the existence of PRGs [Gol90].

However, recent research has revealed the intriguing possibility of basing quantum cryptography on weaker computational assumptions than the existence of classical OWFs/PRGs. For example, a private-key encryption scheme where the honest parties can compute with and transmit quantum information no longer implies the existence of a OWF in an obvious fashion. In fact, there are primitives such as pseudorandom states (PRS) [JLS18] and EFI pairs [BCQ22] that that are apparently even weaker than PRGs and OWFs, in the sense that PRS can be constructed from PRGs/OWFs but are formally separated from them relative to an oracle [Kre21]. Furthermore, it was shown by [AQY22, MY21, AGQY22, MY22, BCQ22] that PRS, variants of PRS, and EFI can be used to construct commitment schemes, zero-knowledge protocols, pseudo one-time pads, and more. Thus it is a fascinating open question to understand what other cryptographic functionalities or primitives can be built from weaker quantum assumptions.

In this work we ask whether there is a *quantum* analogue of PRGs that (a) can be based on quantum assumptions such as PRS or EFI, and (b) are as useful as classical PRGs for cryptographic applications. Such a quantum analogue would have interesting ramifications for the foundations of cryptography as well as complexity theory.

Our Work. We give positive answers to both questions. We introduce the notion of *quantum pseudorandom generators (QPRGs)*: which are like classical PRGs in that the input is a short classical string and the output is a longer classical string that is computationally indistinguishable from uniform, but (a) generation algorithm is a quantum algorithm, and (b) the mapping from seed to output only has to be *pseudodeterministic* (i.e., for a fixed seed, the output is a fixed string with high probability). We first show that assumptions that are plausibly weaker than the existence of classical OWFs/PRGs can be used to build QPRGs: we show that QPRGs can be constructed from PRS. In other words, we can generate pseudorandom strings using pseudorandom quantum states in a (pseudo-)deterministic fashion. We then present cryptographic applications of QPRGs and highlight some implications for the structure of classical versus quantum cryptography.

The reader might wonder whether the notion of quantum generation of classical pseudorandomness is trivial. After all, since quantum computation is inherently probabilistic and can generate unlimited randomness starting from a fixed input, why would one need *pseudo*randomness? However, for cryptographic applications having a source of randomness is not enough; it is important that some random-looking string can be *deterministically generated* using a secret key.

1.1 Our Results

Quantum PRGs from PRS. Informally, a $(1-\varepsilon)$-pseudodeterministic QPRG is a quantum algorithm G where

- (*Pseudodeterminism*) For $1-\varepsilon$ fraction of seeds $k \in \{0,1\}^\lambda$ outputs a fixed string $y_k \in \{0,1\}^n$ with probability at least $1-\varepsilon$, and
- (*Pseudorandomness*) For all efficient quantum distinguishers A,

$$\left| \Pr_{k \leftarrow \{0,1\}^\lambda}[A(G(k)) = 1] - \Pr_{y \leftarrow \{0,1\}^n}[A(y) = 1] \right| \leq \mathsf{negl}(\lambda) \ .$$

  In other words, no efficient quantum adversary can distinguish between the output of the generator and a uniformly random string.

(See Section 4 for a formal definition of QPRGs). Our first result is the following:

Theorem 1.1 (Informal)**.** *Assuming the existence of logarithmic PRS, there exist* $\left(1 - \frac{1}{\mathsf{poly}(\lambda)}\right)$*-pseudodeterministic QPRGs.*

Introduced by Ji, Liu, and Song [JLS18], PRS are a quantum analogue of PRGs: a PRS generator is an efficient quantum algorithm mapping a classical key $k \in \{0,1\}^\lambda$ to a quantum state $|\psi_k\rangle$ that cannot be efficiently distinguished from states sampled from the Haar measure (even when given $\mathsf{poly}(\lambda)$ copies of $|\psi_k\rangle$). Despite being analogous to classical PRGs, PRS are very different in other respects. For example, while classical PRGs that output fewer bits than the seed length are trivial, PRS generators with $O(\log \lambda)$-qubit outputs remain nontrivial – these are the *logarithmic PRS* referred to in Theorem 1.1. Furthermore, unlike classical PRGs, it is not known how to generically shrink or stretch the output length of a PRS; this is because PRS are entangled quantum states that cannot truncated or copied.

Nonetheless we show that when the PRS is logarithmic in length we can extract pseudorandom strings. Due to the probabilistic nature of quantum mechanics we do not get the pseudorandom strings with certainty but instead with very high probability; this is the reason we get *pseudodeterministic* QPRGs. The key idea behind the QPRG construction is to pseudodeterministically extract pseudorandomness from a matrix obtained by performing tomography on copies of the logarithmic-sized pseudorandom state. We elaborate on this in Section 1.3.

One implication of our QPRG construction is that it demonstrates an "inherently quantum" way to generate classical pseudorandomness. There are plausible candidates for PRS (even the logarithmic-length ones) that don't seem to involve any classical OWFs in them at all; for example, it is conjectured that random polynomial-size quantum circuits generate pseudorandom states [AQY22].

A Win-Win Result? Regardless of whether QPRGs can ultimately be based on weaker assumptions than classical PRGs, we get rather striking consequences about the relationships between different primitives in quantum cryptography. Consider the scenario where logarithmic PRS is a truly a weaker assumption than classical OWFs. Then as discussed we get the fascinating result we can base generating pseudorandom strings on weaker assumptions than what is possible classically.

On the other hand, consider the scenario where logarithmic PRS is *not* a weaker assumption than classical OWFs – that is, they are in some sense much closer to classical OWFs/PRGs than they are to EFI pairs or large-length PRS. In this setting, then we get a separation between logarithmic PRS and superlogarithmic PRS (i.e., PRS on $\omega(\log \lambda)$ qubits)! We note that this does not violate Kretschmer's oracle separation [Kre21]: he constructed an oracle relative to which $\mathsf{BQP} = \mathsf{QMA}$ (which implies that pseudorandom generators don't exist) but pseudorandom states still exist. Inspection of his proof shows that it applies to superlogarithmic PRS, and one might ask whether this separation can be extended to all nontrivial PRS requiring computational assumptions (i.e., PRS on $c \log \lambda$ qubits with $c > 1$). However pseudodeterministic QPRGs cannot exist if $\mathsf{BQP} = \mathsf{QMA}$: this is because deciding whether a string y is in the output of the QPRG is in QMA (in fact, QCMA). Thus logarithmic PRS cannot exist also in that same relativized world. This leads to an intriguing possibility: *generating fewer pseudorandom qubits (e.g., logarithmically many in the security parameter) could potentially require stronger assumptions than generating large amounts of pseudorandom qubits (e.g., superlogarithmically many)!* If this is true then this is rather counter-intuitive since in the classical setting, generating fewer pseudorandom bits is easier than generating many pseudorandom bits.

Applications of Quantum PRGs. Next we investigate the cryptographic applications of QPRGs. We demonstrate that QPRGs can effectively replace classical pseudorandom generators in some applications; although the QPRGs are not entirely deterministic, being $(1-\frac{1}{\mathsf{poly}(\lambda)})$-pseudodeterministic is good enough.

Concretely, we explore two applications: statistically binding and computationally hiding commitments, and pseudo one-time pads. While [AGQY22] previously demonstrated that these applications can be based on logarithmic PRS, we provide alternate proofs assuming the existence of QPRGs combined with Theorem 1.1. Moreover, our constructions resemble the textbook constructions of classical commitments and pseudo one-time pads and thus, are conceptually simpler than the ones presented by [AGQY22].

A statistically binding commitment scheme is a fundamental cryptographic notion where a sender commits to a value such that it is infeasible, even if it is computationally unbounded, for them to change their commitment to a different value. Statistically binding quantum commitments have been a critical tool to achieve another fundamental notion in cryptography, namely secure computation [BCKM21, GLSV21]. We demonstrate that statistically binding and computationally hiding commitments can be constructed from QPRGs.

Theorem 1.2 (Informal)**.** *Assuming the existence of $(1-\frac{1}{\mathsf{poly}(\lambda)})$-pseudodetermininistic QPRGs, there exist statistically binding and computationally hiding quantum commitments with classical communication.*

It is worth mentioning that there is another recent work [BBSS23] that also builds quantum commitments with classical communication albeit from incomparable assumptions[1].

[1] They consider a variant of PRS referred to as PRS with proof of deletion. On one hand, they don't have restriction on the output length like we do and on the other hand, they assume that PRS satisfies the additional proof of deletion property whereas we don't.

Pseudo one-time pads are a variation of the one-time pad encryption scheme, where the encryption key is much smaller than the message length. As demonstrated by [AQY22], pseudo one-time pads are useful for constructing classical garbling schemes [AIK06] and quantum garbling schemes [BY22], which have numerous applications in cryptography. We demonstrate that pseudo one-time pads can be constructed from QPRGs.

Theorem 1.3 (Informal)**.** *Assuming the existence of* $(1 - \frac{1}{\mathsf{poly}(\lambda)})$*-pseudodetermininistic QPRGs, there exist pseudo one-time pads.*

Quantum PRFs. In addition to the above, we also explore pseudorandom functions with a quantum generation algorithm, which we call quantum pseudorandom functions (QPRFs). We show the following theorem:

Theorem 1.4 (Informal)**.** *Assuming the existence of* $(\omega(\log \lambda), O(\log \lambda))$*-PRFS, there exists a quantum pseudorandom function (QPRF) satisfying determinism with probability at least* $\left(1 - \frac{1}{\mathsf{poly}(\lambda)}\right)$.

In the above theorem, we use pseudorandom function-like states [AQY22], a quantum analog of pseudorandom functions, to accomplish this. The notion of pseudorandom function-like states says the following: t copies of states $(|\psi_{x_1}\rangle, \ldots, |\psi_{x_q}\rangle)$ are computationally indistinguishable from t copies of q Haar states, where $|\psi_{x_i}\rangle$, for every $i \in [q]$, is produced using an efficient PRFS generator that receives as input a key $k \in \{0,1\}^\lambda$, picked uniformly at random, and an input $x_i \in \{0,1\}^\lambda$. Just like in the case of QPRGs, in the above theorem, we require PRFS with logarithmic input length.

We show how to leverage QPRFs to achieve private-key encryption with QPT algorithms and classical communication, which is the first result to achieve this notion from assumptions potentially weaker than one-way functions.

1.2 Future Directions

Our research raises several important open questions that remain to be explored. Below, we highlight two particularly interesting ones.

Separating QPRGs and QPRFs from Classical Cryptography. While QPRGs and QPRFs are similar in flavor to their classical counterparts, their ability to generate quantum states suggests that they may be based on weaker assumptions than classical pseudorandom generators and functions. A key question is whether there is a fundamental separation between QPRGs and PRGs, as well as between QPRFs and PRFs. Proving that there is no separation would require a mechanism to efficiently dequantize the generation algorithm, which is a challenging task. This is especially true if the quantum generation algorithm involves running a quantum algorithm that is believed to be difficult to efficiently dequantize; for example, Shor's algorithm.

Reducing Determinism Error. One limitation of both QPRGs and QPRFs is that they suffer from inverse polynomial determinism error. It would be interesting to explore whether this error can be reduced to negligible, or whether a negative result can be proven. Understanding the fundamental limits of determinism in quantum pseudorandomness could have important implications. For instance, due to the inverse polynomial error, it is unclear how to apply the GGM transformation [GGM86] to go from quantum pseudorandom generators to quantum pseudorandom functions.

1.3 Technical Overview

We focus on the goal of building a quantum pseudorandom generator from $O(\log(\lambda))$-qubit pseudorandom quantum states. Towards this goal, we identify the important step as follows: extracting $\mathsf{poly}(d(\lambda))$-length binary strings from $\log(d(\lambda))$-qubit Haar states in such a way, the following key properties are satisfied:

- Running the extraction process on $\mathsf{poly}(d(\lambda))$-copies of $|\psi\rangle$ should give the same string y with a very high probability. Ideally, with probability at least $1 - \frac{1}{\mathsf{poly}(d(\lambda))}$,
- The extraction process should run in time $\mathsf{poly}(d(\lambda))$,
- The string y is distributed according to the uniform distribution over $\{0,1\}^{\mathsf{poly}(d(\lambda))}$ as long as $|\psi\rangle$ is sampled from the Haar distribution.

Toy Case. Towards designing an extractor satisfying the above three properties, we first consider an alternate task. Instead of $\mathsf{poly}(d(\lambda))$-copies of the $\log(d(\lambda))$-qubit state $|\psi\rangle$, we are given all the amplitudes of $|\psi\rangle$, say $(\alpha_1, \ldots, \alpha_{d(\lambda)})$, in the clear. Can we extract true randomness from this? For instance, we could extract $b_1, \ldots, b_{d(\lambda)}$, where b_i is the first bit of the real component of α_i. Firstly, it is not even clear that b_i is distributed to according to the uniform distribution over $\{0,1\}$. Moreover, all the bits $b_1, \ldots, b_{d(\lambda)}$ are not independent and in fact, are correlated with each other due to the normalization condition $\sum_i |\alpha_i|^2 = 1$.

Fortunately, we can rely upon a result in random matrix theory [Mec19], that states the following: suppose $(\alpha_1, \ldots, \alpha_{d(\lambda)})$ are drawn from a Haar measure in $\mathcal{S}(\mathbb{R}^d)$ then it holds that any $o(d(\lambda))$ coordinates of $(\alpha_1, \ldots, \alpha_{d(\lambda)})$ are $1/o(d(\lambda))$-close in total variation distance with $o(d(\lambda))$-dimensional vector where each component is drawn from i.i.d Gaussian $\mathcal{N}(0, \frac{1}{d})$.

We generalize this result to the case when $(\alpha_1, \ldots, \alpha_{d(\lambda)})$ are drawn from a Haar measure in $\mathcal{S}(\mathbb{C}^d)$, and not $\mathcal{S}(\mathbb{R}^d)$ (see Corollary 2.13) at the cost of reducing the standard deviation from $\frac{1}{d}$ to $\frac{1}{2d}$. We then use our observation to come up with an extractor as follows. The extractor takes as input[2] $(\alpha_1, \ldots, \alpha_{d(\lambda)})$,

- Choose the first $k = o(d(\lambda))$ entries among $(\alpha_1, \ldots, \alpha_{d(\lambda)})$.
- Rounding step: for every $i \in [k]$, if $\mathsf{Re}(\alpha_i) > 0$, then set $b_i = 0$. Otherwise, set $b_i = 1$.
- Output $b_1 \cdots b_k$.

From our observation and the symmetricity of $\mathcal{N}(0, \frac{1}{d})$, it follows that when $(\alpha_1, \ldots, \alpha_{d(\lambda)})$ is drawn from a Haar distribution on $\mathcal{S}(\mathbb{C}^d)$ then the output of the extractor is $o(\frac{1}{d(\lambda)})$-close to uniform distribution on $\{0,1\}^k$. Moreover, the above procedure is deterministic.

[2] For the current discussion, we assume that the extractor has an infinite input tape that allows for storing infinite bits of precision of the complex numbers.

Challenges. Our hope is to leverage the above ideas to design an extractor that can extract given $\mathsf{poly}(d(\lambda))$-copies of a $O(\log(d(\lambda)))$-qubit Haar state $|\psi\rangle$. We encounter a couple of challenges.

1. First challenge: We have access only to the copies of $|\psi\rangle\langle\psi|$ without the amplitudes given to us in plain text, making it infeasible to implement the previously described method. However, we can still carry out tomography and retrieve an estimated version of the matrix $|\psi\rangle\langle\psi|$. If the amplitudes of $|\psi\rangle$ are $\{\alpha_x\}_{x\in[d]}$ then the $(x,y)^{th}$ entry in the density matrix $|\psi\rangle\langle\psi|$ is $\alpha_x\alpha_y^*$. We need to analyze the distribution corresponding to $\alpha_x\alpha_y^*$ and, design an approach for obtaining a uniform distribution from it.

2. Second challenge: Tomography is inherently a probabilistic technique, and hence, each time tomography is executed on multiple copies of $|\psi\rangle$, the output obtained may differ. Additionally, the trace distance between the density matrix obtained via tomography and the original density matrix is inversely proportional to the dimension, which is polynomial in this case, and this may be significant. Both of these factors collectively affect the determinism guarantees of the extractor. In general, it is not feasible to partition $\mathcal{S}(\mathbb{C}^d)$ into regions labeled by a bitstring such that given multiple copies of a state in a region, the corresponding bitstring can be deterministically recovered.

We tackle the above challenges using the following insights.

Addressing the first challenge. We first tackle the first bullet above. Notice that the diagonal entries in the density matrix $|\psi\rangle\langle\psi|$ is $\{|\alpha_i|^2\}_{i\in[d(\lambda)]}$, where $|\psi\rangle = \sum_i \alpha_i |i\rangle$. If $\alpha_j = a_j + ib_j$ then $|\alpha_j|^2 = a_j^2 + b_j^2$. Given our earlier observation about the closeness of $o(d(\lambda))$ entries in a vector drawn from $\mathcal{S}(\mathbb{C}^d)$ with iid Gaussian, we will make the following simplifying assumption. We assume that $(\alpha_1,\ldots,\alpha_k)$, where $k = o(d)$, is sampled such that for every $i \in [k]$, a_i and b_i are distributed according to i.i.d Gaussian $\mathcal{N}(0, \frac{1}{2d})$. From this, it follows that $|\alpha_i|^2$ is distributed according to a *chi-squared* distribution with 2 degrees of freedom. Unfortunately, chi-squared distribution does not have the same nice symmetricity property as a Gaussian distribution. So we will instead extract randomness in a different way.

We divide $(|\alpha_1|^2,\ldots,|\alpha_k|^2)$ into blocks of size r and denote ℓ to be the number of blocks, where $r,\ell = o(d)$. Then, add the elements in a block. Call the resulting elements $q_1,\ldots,q_\ell$. From central limit theorems [SM62], one can show that $q_1,\ldots,q_\ell$ are $O(1/\sqrt{r})$-close to ℓ samples drawn i.i.d from $\mathcal{N}(\frac{r}{d}, \frac{r}{d^2})$. Thus, using central limit theorem, we are back to the normal distribution, except that the mean is shifted to $\frac{r}{d}$ rather than 0. This gives rise to a natural rounding mechanism.

We will check if $q_i > \frac{r}{d}$ and if so, we set a bit $b_i = 0$ and if not, we set it to be 0. By carefully choosing the parameters k and ℓ and combining the above observations, we can argue that $b_1,\ldots,b_k$ is $O(d^{-1/6})$-close to the uniform distribution on $\{0,1\}^\ell$.

To summarise, the informal description of the extractor is as follows: given $\mathsf{poly}(d)$ copies of a d-dimensional state $|\psi\rangle$,

- First perform tomography to recover a matrix $M \in \mathbb{C}^d \times \mathbb{C}^d$ that is an approximation of $|\psi\rangle\langle\psi|$
- Then, pick $o(d)$ diagonal entries in M and break this into ℓ blocks of size r.
- Sum up all the entries in each block to get ℓ values $q_1,\ldots,q_\ell$. Round every q_i to get b_i.
- Output $b_1,\ldots,b_\ell$.

As remarked earlier, the output distribution of the above extractor is only $O(d^{-1/6})$-statistically close to the uniform distribution on $\{0,1\}^{\ell}$. To reduce the statistical distance to negligible in d, we will use a standard XOR-based security amplification mechanism [DIJK09, MT09, MT10]: we take multiple outputs of the extractor on i.i.d Haar states and then XOR all the outputs. While the analysis in the amplification theorems [DIJK09, MT09, MT10] was initially tailored to classical functions, we show that the analysis extends to the quantum setting as well.

Addressing the second challenge. While the above construction seems promising, we still have not addressed the second challenge pertaining to the determinism property. It could be the case that all the q_is are very close to the mean and due to the tomography error, every time we try to extract we set $b_i = 0$ sometimes and $b_i = 1$ the rest of the time. This should not be surprising as we said earlier, that it should not be possible to partition $\mathcal{S}(\mathbb{C}^d)$ such that for every $|\psi\rangle$, there is a bitstring b_ψ such that given many copies of $|\psi\rangle$, the extractor always outputs the same bitstring b_ψ.

In fact, we can identify a *forbidden region* in $\mathcal{N}(\frac{r}{d}, \frac{r}{d^2})$ (see Figure 1 below) such that if q_i falls into the forbidden region then there is a significant chance that q_i will be classified as either 0 or 1. The forbidden region has width $\frac{1}{d}$ on either side of the mean. Given this, we give up all hope of achieving perfect determinism and instead shoot for determinism with $o(1/d)$ error.

We identify a set of $d(\lambda)$-dimensional states $\mathcal{G}_\Delta$, where $\Delta = \frac{1}{d}$, such that if a state $|\psi\rangle$ is in $\mathcal{G}_\Delta$ then it holds that none of $q_1, \ldots, q_\ell$, generated from $|\psi\rangle$, lies in the forbidden region. The setting of Δ is carefully chosen to accommodate for the error in tomography.

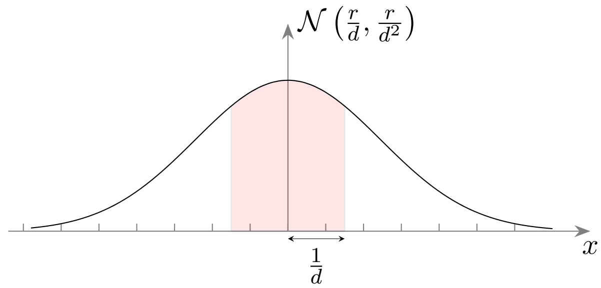

Figure 1: The red region denotes the *forbidden region*.

Once $\mathcal{G}_\Delta$ is identified, we prove two things:

- Firstly, if a state is sampled from the Haar distribution on $\mathcal{S}(\mathbb{C}^{d(\lambda)})$ then with at least $1 - o\left(\frac{1}{d}\right)$ probability, $|\psi\rangle \in \mathcal{G}_\Delta$.

- Secondly, for every $|\psi\rangle \in \mathcal{G}_\Delta$, the probability that the extractor, given $\mathsf{poly}(d(\lambda))$-copies of $|\psi\rangle$, outputs the same string twice is at least $1 - o\left(\frac{1}{d}\right)$. Roughly, this follows from the fact that $(q_1, \ldots, q_\ell)$, generated from $|\psi\rangle$, gets misclassified with very small probability.

We can leverage the above two observations to show that our extractor satisfies determinism with probability at least $1 - o\left(\frac{1}{d}\right)$.

Other Results. We can extend the above technique to get a quantum pseudorandom function (QPRF) starting from a PRFS with $\log(d)$-output length. Similar to the QPRG case, the resulting QPRF has $\frac{1}{\mathsf{poly}}$ determinism error.

We show that QPRGs imply both statistically binding commitments and pseudo one-time pads. The constructions are similar to the existing constructions from (classical) pseudorandom generators. For instance, the construction of statistically binding commitments from QPRGs is inspired by Naor commitments [Nao91]. However, due to the fact that there is an inverse polynomial determinism, the classical constructions cannot be immediately adopted and we come up with new techniques to work around the determinism issue. Similarly, we also demonstrate that QPRFs imply private-key encryption.

2 Preliminaries

We refer the reader to [NC10] for a comprehensive reference on the basics of quantum information and quantum computation. We use I to denote the identity operator. We use $\mathcal{S}(\mathcal{H})$ to denote the set of unit vectors in the Hilbert space $\mathcal{H}$. We use $\mathcal{D}(\mathcal{H})$ to denote the set of density matrices in the Hilbert space $\mathcal{H}$. Let P, Q be distributions. We use $d_{TV}(P, Q)$ to denote the total variation distance between them. Let $\rho, \sigma \in \mathcal{D}(\mathcal{H})$ be density matrices. We write $\mathrm{TD}(\rho, \sigma)$ to denote the trace distance between them, i.e.,

$$\mathrm{TD}(\rho, \sigma) = \frac{1}{2}\|\rho - \sigma\|_1$$

where $\|X\|_1 = \mathrm{Tr}(\sqrt{X^\dagger X})$ denotes the trace norm. We denote $\|X\| := \sup_{|\psi\rangle}\{\langle\psi|X|\psi\rangle\}$ to be the operator norm where the supremum is taken over all unit vectors. For a vector $|x\rangle$, we denote its Euclidean norm to be $\||x\rangle\|_2$. We use the notation $M \geq 0$ to denote the fact that M is positive semi-definite.

Haar Measure. The Haar measure over $\mathbb{C}^d$, denoted by $\mathscr{H}(\mathbb{C}^d)$ is the uniform measure over all d-dimensional unit vectors. One useful property of the Haar measure is that for all d-dimensional unitary matrices U, if a random vector $|\psi\rangle$ is distributed according to the Haar measure $\mathscr{H}(\mathbb{C}^d)$, then the state $U|\psi\rangle$ is also distributed according to the Haar measure. For notational convenience we write $\mathscr{H}_m$ to denote the Haar measure over m-qubit space, or $\mathscr{H}((\mathbb{C}^2)^{\otimes m})$.

2.1 Quantum Algorithms

A quantum algorithm A is a family of generalized quantum circuits $\{A_\lambda\}_{\lambda\in\mathbb{N}}$ over a discrete universal gate set (such as $\{CNOT, H, T\}$). By generalized, we mean that such circuits can have a subset of input qubits that are designated to be initialized in the zero state, and a subset of output qubits that are designated to be traced out at the end of the computation. Thus a generalized quantum circuit A_λ corresponds to a *quantum channel*, which is a is a completely positive trace-preserving (CPTP) map. When we write $A_\lambda(\rho)$ for some density matrix ρ, we mean the output of the generalized circuit A_λ on input ρ. If we only take the quantum gates of A_λ and ignore the subset of input/output qubits that are initialized to zeroes/traced out, then we get the *unitary part* of A_λ, which corresponds to a unitary operator which we denote by $\hat{A}_\lambda$. The *size* of a generalized quantum circuit is the number of gates in it, plus the number of input and output qubits.

We say that $A = \{A_\lambda\}_\lambda$ is a *quantum polynomial-time (QPT) algorithm* if there exists a polynomial p such that the size of each circuit A_λ is at most $p(\lambda)$. We furthermore say that A is *uniform* if there exists a deterministic polynomial-time Turing machine M that on input 1^λ outputs the description of A_λ.

We also define the notion of a *non-uniform* QPT algorithm A that consists of a family $\{(A_\lambda, \rho_\lambda)\}_\lambda$ where $\{A_\lambda\}_\lambda$ is a polynomial-size family of circuits (not necessarily uniformly generated), and for each λ there is additionally a subset of input qubits of A_λ that are designated to be initialized with the density matrix ρ_λ of polynomial length. This is intended to model nonuniform quantum adversaries who may receive quantum states as advice. Nevertheless, the reductions we show in this work are all uniform.

The notation we use to describe the inputs/outputs of quantum algorithms will largely mimick what is used in the classical cryptography literature. For example, for a state generator algorithm G, we write $G_\lambda(k)$ to denote running the generalized quantum circuit G_λ on input $|k\rangle\langle k|$, which outputs a state ρ_k.

Ultimately, all inputs to a quantum circuit are density matrices. However, we mix-and-match between classical, pure state, and density matrix notation; for example, we may write $A_\lambda(k, |\theta\rangle, \rho)$ to denote running the circuit A_λ on input $|k\rangle\langle k| \otimes |\theta\rangle\langle\theta| \otimes \rho$. In general, we will not explain all the input and output sizes of every quantum circuit in excruciating detail; we will implicitly assume that a quantum circuit in question has the appropriate number of input and output qubits as required by context.

2.2 Pseudorandomness Notions

The notion of pseudorandom quantum states was first introduced by Ji, Liu, and Song in [JLS18]. We present the following relaxed definition of pseudorandom state (PRS) generators.[3] We note that the relaxation is due to [AQY22].

Definition 2.1 (Pseudorandom State (PRS) Generator)**.** *We say that a QPT algorithm G is a* pseudorandom state (PRS) generator *if the following holds.*

1. **State Generation.** *For all $\lambda \in \mathbb{N}$ and all $k \in \{0,1\}^\lambda$, the algorithm G behaves as $G_\lambda(k) = \rho_k$ for some $n(\lambda)$-qubit (possibly mixed) quantum state ρ_k.*

2. **Pseudorandomness.** *For all polynomials $t(\cdot)$ and any (non-uniform) QPT distinguisher A, there exists a negligible function $\varepsilon(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have*

$$\left| \Pr_{k \leftarrow \{0,1\}^\lambda}[A_\lambda(G_\lambda(k)^{\otimes t(\lambda)}) = 1] - \Pr_{|\vartheta\rangle \leftarrow \mathscr{H}_{n(\lambda)}}[A_\lambda(|\vartheta\rangle^{\otimes t(\lambda)}) = 1] \right| \leq \varepsilon(\lambda).$$

We also say that G is an $n(\lambda)$-PRS generator *to succinctly indicate that the output length of G is $n(\lambda)$.*

Definition 2.2 (Selectively Secure Pseudorandom Function-Like State (PRFS) Generators)**.** *We say that a QPT algorithm G is a* selectively secure pseudorandom function-like state (PRFS) generater *if for all polynomials $q(\cdot), t(\cdot)$, any (non-uniform) QPT distinguisher A, and any family of pairwise distinct indices $\left(\{x_1, \ldots, x_{q(\lambda)}\} \subseteq \{0,1\}^{m(\lambda)}\}\right)_\lambda$, there exists a negligible function $\varepsilon(\cdot)$ such that for all $\lambda \in \mathbb{N}$,*

$$\Big| \Pr_{k \leftarrow \{0,1\}^\lambda}\left[A_\lambda(x_1, \ldots, x_{q(\lambda)}, G_\lambda(k, x_1)^{\otimes t(\lambda)}, \ldots, G_\lambda(k, x_{q(\lambda)})^{\otimes t(\lambda)}) = 1\right]$$

[3] In [JLS18], the output of the generator needs to be pure; while we allow it to be mixed.

$$- \Pr_{|\vartheta_1\rangle,\ldots,|\vartheta_{q(\lambda)}\rangle \leftarrow \mathscr{H}_{n(\lambda)}} \left[A_\lambda(x_1,\ldots,x_{q(\lambda)}, |\vartheta_1\rangle^{\otimes t(\lambda)}, \ldots, |\vartheta_{q(\lambda)}\rangle^{\otimes t(\lambda)}) = 1 \right] \Bigg| \leq \varepsilon(\lambda).$$

We also say that G is an $(m(\lambda), n(\lambda))$-PRFS generator *to succinctly indicate that its input length is $m(\lambda)$ and its output length is $n(\lambda)$.*

2.3 Basics of Statistics and Haar Measure

A simple yet useful observation is that for any two density matrices, the difference between any of their diagonal entries is bounded above by their trace distance.

Fact 2.3. *For any density matrices $\rho, \sigma \in \mathcal{D}(\mathbb{C}^d)$, it holds that $\max_{i\in[d]} |\rho_{ii} - \sigma_{ii}| \leq \mathrm{TD}(\rho, \sigma)$, where ρ_{ii}, σ_{ii} denote the i-th diagonal entry of ρ, σ respectively, i.e., $\rho_{ii} = \langle i|\rho|i\rangle$ and $\sigma_{ii} = \langle i|\sigma|i\rangle$.*

Proof. Note that the trace distance has the following variational form:

$$\mathrm{TD}(\rho, \sigma) = \max_{0\leq M\leq I} \mathrm{Tr}(M(\rho - \sigma)).$$

Furthermore, trace distance is symmetric. Therefore, taking $M := |i\rangle\langle i|$ for $i \in [d]$, we have $\mathrm{TD}(\rho, \sigma) \geq \max(\rho_{ii} - \sigma_{ii}, \sigma_{ii} - \rho_{ii}) = |\rho_{ii} - \sigma_{ii}|$ as desired. □

Fact 2.4. *Let X, Y be random variables and f be a function. Then $d_{TV}(f(X), f(Y)) \leq d_{TV}(X, Y)$.*

Lemma 2.5 (Chernoff-Hoeffding Inequality)**.** *Let $X_1, X_2, \ldots, X_n$ be independent random variables, such that $0 \leq X_i \leq 1$ for all $i \in [n]$. Let $X = \sum_{i=1}^n X_i$ and $\mu = \mathbb{E}[X]$. Then for any $\varepsilon > 0$,*

$$\Pr[|X - \mu| > \varepsilon)] \leq 2e^{-\frac{2\varepsilon^2}{n}}.$$

2.3.1 Chi-Squared Distributions

We present the definition and properties of the chi-squared distribution in the following.

Definition 2.6 (Chi-Squared Distribution)**.** *Let $Z_1, \ldots, Z_k$ be i.i.d. Gaussian random variables $\mathcal{N}(0, 1)$. The random variable*

$$Q := \sum_{i\in[k]} Z_i^2.$$

is distributed according to the chi-squared distribution with k degrees of freedom*, denoted by $Q \sim \chi_k^2$.*

Fact 2.7. *Let $Z \sim \mathcal{N}(0, 1)$. Z^2 has a finite third moment.*

Fact 2.8. *For all $k \in \mathbb{N}$, the following holds. Let $Q \sim \chi_k^2$. The mean of Q is k and the variance of Q is $2k$. Moreover, suppose $Q_1 \sim \chi_{k_1}^2$ and $Q_2 \sim \chi_{k_2}^2$, then $Q_1 + Q_2 \sim \chi_{k_1+k_2}^2$. When $k = 1$, we often omit the subscript and denote it by χ^2.*

We introduce a strong version of the central limit theorem that characterizes the *total variation distance* between the sum of i.i.d. *absolutely continuous*[4] random variables and Gaussian random variables. Note that most versions of central limit theorems state only the convergence in *cumulative density function*, which is not sufficient for our purpose.

[4] A random variable X is absolutely continuous if there exists a (probability density) function $f : \mathbb{R} \to [0, 1]$ such that $\Pr[X \leq x] = \int_{-\infty}^{x} f(t)\, dt$ for all $x \in \mathbb{R}$ and $\int_{-\infty}^{\infty} f(t)\, dt = 1$.

Lemma 2.9 ([SM62, Theorem 1], restated)**.** *Let $X_1, \dots, X_k$ be i.i.d. random variables. If X_1 is absolutely continuous and has a finite third moment, then*

$$d_{TV}\left(\frac{\sum_{i\in[k]}(X_i-\mu)}{\sqrt{k}\sigma}, Z\right) = O\left(\frac{1}{\sqrt{k}}\right),$$

where μ is the mean of X_1, σ is the standard deviation of X_1 and $Z \sim \mathcal{N}(0,1)$. Equivalently, $d_{TV}(\sum_{i\in[k]} X_i, Z') = O(1/\sqrt{k})$, where $Z' \sim \mathcal{N}(k\mu, k\sigma^2)$.

Since a random variable with a chi-squared distribution is the sum of squared i.i.d. Gaussian random variables, we have the following immediate corollary.

Corollary 2.10. *Let Q be a random variable with a distribution χ_k^2. Then $d_{TV}(Q, Z) = O(1/\sqrt{k})$, where $Z \sim \mathcal{N}(k, 2k)$.*

Proof. By definition, $Q = \sum_{i\in[k]} Z_i^2$ where $Z_i \sim \mathcal{N}(0,1)$. It immediately follows from the facts that Z_i^2 is absolutely continuous, $\mathbb{E}\left[Z_i^2\right] = 1$, $\mathsf{Var}\left(Z_i^2\right) = 2$, the third moment of Z_i^2 is finite and Lemma 2.9. □

2.3.2 Haar Measure

Given a d-dimensional Haar state, all coordinates of the state are correlated due to the unit-norm condition. The following theorem states that the *joint distribution* of $k = o(d)$ fraction of the coordinates in $\mathcal{S}(R^d)$ is statistically close to a random vector with i.i.d. Gaussian entries. The theorem was first proven in [DF87]. We will use the version stated in [Mec19].

Theorem 2.11 ([Mec19, Theorem 2.8])**.** *For every integer $d \geq 5$ and every $k \in \mathbb{N}$ that satisfies $1 \leq k \leq d-4$, let $X = (X_1, \dots, X_d)$ be a uniform point on $\mathcal{S}(\mathbb{R}^d)$. Let Z be a random vector in $\mathbb{R}^k$ with i.i.d. Gaussian entries $\mathcal{N}(0, 1/d)$. Then*

$$d_{TV}\left((X_1, \dots, X_k), Z\right) \leq \frac{2(k+2)}{d-k-3}.$$

The above lemma can be extended to uniformly random vectors on $\mathcal{S}(\mathbb{C}^d)$. For a complex number α, we denote by $\mathsf{Re}(\alpha)$ and $\mathsf{Im}(\alpha)$, in order, the real part and imaginary part of α.

Lemma 2.12. *Let $|\psi\rangle = \sum_{i\in[d]} \alpha_i |i\rangle$ be a uniform point on $\mathcal{S}(\mathbb{C}^d)$. Then $(\mathsf{Re}(\alpha_1), \dots, \mathsf{Re}(\alpha_d), \mathsf{Im}(\alpha_1), \dots, \mathsf{Im}(\alpha_d))$ is a uniform point on $\mathcal{S}(\mathbb{R}^{2d})$.*

Proof. First proposed by Muller [Mul59], a uniform point on $\mathcal{S}(\mathbb{R}^{2d})$ can be sampled via the following procedures:

1. For $i \in [2d]$, sample $a_i \leftarrow \mathcal{N}(0, \sigma^2)$.
2. Output $\sum_{i\in[2d]} \frac{a_i}{\sqrt{\sum_{j\in[2d]} a_j^2}} |i\rangle$.

Where σ^2 in step 1 could be an arbitrary positive number.
On the other hand, a uniform point on $\mathcal{S}(\mathbb{C}^d)$ can be sampled as follows:

1. For $i \in [d]$, sample $\alpha_i \sim \mathbb{C}\mathcal{N}(0,1)$.

2. Output $\sum_{i\in[d]} \frac{\alpha_i}{\sqrt{\sum_{j\in[d]} |\alpha_j|^2}} |i\rangle$.

Particularly, in step 1, sampling $\alpha \sim \mathbb{C}\mathcal{N}(0,1)$ is equivalent to sampling $\alpha = a + ib$ according to $a \sim \mathcal{N}(0, 1/2)$, $b \sim \mathcal{N}(0, 1/2)$ by the definition of the complex normal distribution. Hence, picking $\sigma^2 = 1/2$ completes the proof. ☐

Corollary 2.13. *For every integer $d \geq 3$ and every $k \in \mathbb{N}$ that satisfies $1 \leq 2k \leq 2d - 4$, let $|\psi\rangle = \sum_{i\in[d]} \alpha_i |i\rangle$ be a random point on $\mathcal{S}(\mathbb{C}^d)$. Let Z be a random vector in $\mathbb{R}^{2k}$ with i.i.d. Gaussian entries $\mathcal{N}(0, 1/(2d))$. Then*

$$d_{TV}\left((\mathsf{Re}(\alpha_1), \ldots, \mathsf{Re}(\alpha_k), \mathsf{Im}(\alpha_1), \ldots, \mathsf{Im}(\alpha_k)), Z\right) \leq \frac{2(2k+2)}{2d - 2k - 3}.$$

Proof. It immediately follows from Theorem 2.11 and Lemma 2.12. ☐

Next, the following simple fact gives an upper bound of the probability that a Gaussian random variable takes values near its mean.

Fact 2.14. *Let $Z \sim \mathcal{N}(\mu, \sigma^2)$. For any $\Delta > 0$,*

$$\Pr[|Z - \mu| \leq \Delta] \leq \sqrt{\frac{2}{\pi}} \frac{\Delta}{\sigma}.$$

Proof. Let $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$ be the probability density function of $\mathcal{N}(\mu, \sigma^2)$. The probability $\int_{\mu-\Delta}^{\mu+\Delta} f(x)\, dx$ can be upper-bounded by $f(\mu) \cdot 2\Delta = \sqrt{\frac{2}{\pi}} \frac{\Delta}{\sigma}$. ☐

2.4 Quantum State Tomography

Lemma 2.15 ([AGQY22, Corollary 7.6])**.** *There exists a tomography procedure* $\mathsf{Tomography}$ *that satisfies the following. For any error tolerance $\delta = \delta(\lambda) \in (0, 1]$ and any dimension $d = d(\lambda) \in \mathbb{N}$, given at least $t = t(\lambda) := 36\lambda d^3/\delta$ copies of a d-dimensional density matrix ρ,* $\mathsf{Tomography}(\rho^{\otimes t})$ *outputs a matrix $M \in \mathbb{C}^{d\times d}$ such that the following holds:*

$$\Pr\left[\|M - \rho\|_F^2 \leq \delta : M \leftarrow \mathsf{Tomography}(\rho^{\otimes t})\right] \geq 1 - \mathsf{negl}(\lambda).$$

Moreover, the running time of $\mathsf{Tomography}$ *is polynomial in $1/\delta, d$ and λ.*

By using the fact that $\|A\|_1 \leq \sqrt{d}\|A\|_F$, we have the following immediate corollary.

Corollary 2.16. *There exists a tomography procedure* $\mathsf{Tomography}$ *that satisfies the following. For any error tolerance $\delta = \delta(\lambda) \in (0, 1]$ and any dimension $d = d(\lambda) \in \mathbb{N}$, given at least $t = t(\lambda) := 144\lambda d^4/\delta^2$ copies of a d-dimensional density matrix ρ,* $\mathsf{Tomography}(\rho^{\otimes t})$ *outputs a matrix $M \in \mathbb{C}^{d\times d}$ such that the following holds:*

$$\Pr\left[\mathrm{TD}(M, \rho) \leq \delta : M \leftarrow \mathsf{Tomography}(\rho^{\otimes t})\right] \geq 1 - \mathsf{negl}(\lambda).$$

Moreover, the running time of $\mathsf{Tomography}$ *is polynomial in $1/\delta, d$ and λ.*

3 Deterministically Extracting Classical Strings from Quantum States

In this section, we show how to pseudodeterministically extract classical strings from $O(\log(\lambda))$-qubit quantum states in *polynomial* time. We first present the outline of our construction.

1. Take as input $t(\lambda)$ copies of a $d(\lambda)$-dimensional (possibly mixed) quantum state ρ. Note that for our applications, we require $d(\lambda) = \mathsf{poly}(\lambda)$ and $t(\lambda) = \mathsf{poly}(\lambda)$.
2. Perform $\mathsf{Tomography}$ on the input $\rho^{\otimes t(\lambda)}$ to get an approximation $M \in \mathbb{C}^{d\times d}$ of its classical description.
3. Pick the first $k = o(d)$ diagonal entries of M, denoted by $p_1, \ldots, p_k$. Divide them into ℓ groups where each of them is of size r (namely, $k = \ell \cdot r$).
4. In each group, consider the sum of all the elements. By q_i we denote the sum of the i-th group.
5. For each q_i, we round it to a bit, called b_i, according to which side it deviates from r/d.
6. Output the concatenation of every bit $b_1 || \ldots || b_\ell$.

In particular, we are interested in the case where the input is (polynomially many copies of) a *Haar state*. Informally, a Haar state can be thought of as a uniformly random point on a high-dimensional sphere. We can partition the sphere into many regions and assign each region a unique bitstring. Given the input quantum state, the goal of the extractor is to find the corresponding bitstring. Hence, Haar states can be viewed as a natural source of randomness. Below, we present our main theorem.

Theorem 3.1. *There exists a quantum algorithm* Ext *such that for all* $d(\cdot)$*, there exists a (deterministic) function* $f : \mathcal{D}(\mathbb{C}^{d(\lambda)}) \to \{0,1\}^{\ell(\lambda)}$ *associated with* Ext*, where* $\ell(\lambda) = \lfloor d(\lambda)^{1/6} \rfloor$*. On input* $t(\lambda) = \mathsf{poly}(d(\lambda), \lambda)$ *copies of a* $d(\lambda)$*-dimensional density matrix* ρ*, the algorithm* Ext *outputs an* $\ell(\lambda)$*-bit string* y *and satisfies the following conditions.*

- **Efficiency:** *The running time is polynomial in* d *and* λ*.*
- **Correctness:** *For all* $\lambda \in \mathbb{N}$*, there exists a set* $\mathcal{G}_\Delta \subseteq \mathcal{S}(\mathbb{C}^{d(\lambda)})$ *such that*
  1. $\Pr\left[|\psi\rangle \in \mathcal{G}_\Delta : |\psi\rangle \leftarrow \mathscr{H}(\mathbb{C}^{d(\lambda)})\right] \geq 1 - O(d(\lambda)^{-1/6})$.
  2. *For all* $|\psi\rangle \in \mathcal{G}_\Delta$*,*
  $$\Pr\left[y = f(|\psi\rangle\langle\psi|) : y \leftarrow \mathsf{Ext}\left(|\psi\rangle\langle\psi|^{\otimes t(\lambda)}\right)\right] \geq 1 - \mathsf{negl}(\lambda),$$
  where the probability is over the randomness of the extractor Ext*.*
- **Statistical Closeness to Uniformity:** *For sufficiently large* $\lambda \in \mathbb{N}$*,*
  $$d_{TV}(Y_\lambda, U_{\ell(\lambda)}) \leq O(d(\lambda)^{-1/6}),$$
  where $U_{\ell(\lambda)}$ *is the uniform distribution over all* $\ell(\lambda)$*-bit strings and the random variable* Y_λ *is defined by the following process:*
  $$|\psi\rangle \leftarrow \mathscr{H}(\mathbb{C}^{d(\lambda)}), Y_\lambda \leftarrow \mathsf{Ext}\left(|\psi\rangle\langle\psi|^{\otimes t(\lambda)}\right).$$

Proof. Here we present our construction of the extractor Ext.

Construction 3.2 (The Extractor Ext)**.**

- *Input:* $t(\lambda) := \lceil 144\lambda d(\lambda)^8 \rceil$ *copies of a* $d(\lambda)$*-dimensional quantum state* $\rho \in \mathcal{D}(\mathbb{C}^{d(\lambda)})$.
- *Perform* $\mathsf{Tomography}(\rho^{\otimes t(\lambda)})$ *with error tolerance* $\delta(\lambda) := d(\lambda)^{-5/3}$ *to get the classical description* $M \in \mathbb{C}^{d(\lambda)\times d(\lambda)}$ *that approximates* ρ.
- *Run* $\mathsf{Round}(M)$ *to get* $y \in \{0,1\}^{\ell(\lambda)}$.
- *Output* y.

The classical post-processing procedure $\mathsf{Round}(M)$ is defined as follows:

$\mathsf{Round}(M)$:

- Input: a matrix $M \in \mathbb{C}^{d(\lambda)\times d(\lambda)}$.
- Set parameters $k(\lambda) := d(\lambda)^{5/6}$, $r(\lambda) := d(\lambda)^{2/3}$ and $\ell(\lambda) := d(\lambda)^{1/6}$.
- Let $p_1, \ldots, p_{d(\lambda)}$ be the diagnal entries of M. For $i \in \{1, \ldots, \ell(\lambda)\}$, let

$$q_i := \sum_{j=1}^{r} p_{(i-1)r+j}.$$

- For $i \in \{1, \ldots, \ell(\lambda)\}$, define

$$b_i = \begin{cases} 0, & \text{if } q_i < r/d \\ 1, & \text{if } q_i > r/d. \end{cases}$$

- Output $b_1 || \ldots || b_{\ell(\lambda)}$.

By Corollary 2.16 and the fact that $t(\lambda) = \mathsf{poly}(d, \lambda)$ and $\delta(\lambda) = 1/\mathsf{poly}(d)$, it is easy to see that the running time of the extractor Ext is polynomial in d and λ. Before proving the correctness and statistical closeness to uniformity, we present several statistical properties.

First, the distribution of the real and imaginary parts of any $k = o(d)$ coordinates of a Haar state $|\psi\rangle \sim \mathscr{H}(\mathbb{C}^d)$ is statistically close to a random vector with i.i.d. Gaussian entries.

Claim 3.3. *Let* $|\psi\rangle = \sum_{i=1}^{d} \alpha_i |i\rangle$ *be a uniformly random point on* $\mathcal{S}(\mathbb{C}^d)$*. Then*

$$d_{TV}\left((\mathsf{Re}(\alpha_1), \mathsf{Im}(\alpha_1), \ldots, \mathsf{Re}(\alpha_k), \mathsf{Im}(\alpha_k)), Z\right) = O\left(k/d\right),$$

where Z *is a random variable in* $\mathbb{R}^{2k}$ *with i.i.d. Gaussian entries* $\mathcal{N}(0, 1/(2d))$.

Proof. It immediately follows from Corollary 2.13. □

Next, since the i-th diagonal entry p_i of $|\psi\rangle\langle\psi|$ is the squared absolute value of the i-th coordinate α_i of $|\psi\rangle$, the joint distribution of $(p_1, \ldots, p_k)$ is statistically close to a random vector in $\mathbb{R}^k$ with i.i.d. χ^2_2 entries.

Claim 3.4. *$d_{TV}((p_1,\ldots,p_k), Q/(2d)) = O(k/d)$ where Q is a random variable in $\mathbb{R}^k$ with i.i.d. χ_2^2 entries.*

Proof. For $i \in [k]$, each diagonal entry $p_i = |\alpha_i|^2 = \mathsf{Re}(\alpha_i)^2 + \mathsf{Im}(\alpha_i)^2$. By Claim 3.3, the total variation distance induced by replacing the real and imaginary parts of the amplitudes with i.i.d. Gaussians $\mathcal{N}(0, 1/(2d))$ is $O(k/d)$. Then by setting $f(x_1,\ldots,x_{2k}) := \left(x_1^2 + x_2^2, \ldots, x_{2k-1}^2 + x_{2k}^2\right)$ in Fact 2.4 and the definition of χ_2^2, we complete the proof. $\square$

Now, we consider the distribution of q_i's. Note that the sum of r i.i.d. χ_2^2 random variables is identically distributed to χ_{2r}^2 by the property of the χ^2-distribution in Fact 2.8. Namely, the joint distribution of $(q_1,\ldots,q_\ell)$ is statistically close to a random vector in $\mathbb{R}^\ell$ with i.i.d. χ_{2r}^2 entries.

Claim 3.5. *$d_{TV}((q_1,\ldots,q_\ell), R/(2d)) = O(k/d)$ where R is a random variable in $\mathbb{R}^\ell$ with i.i.d. χ_{2r}^2 entries.*

Proof. Recall that $q_i := \sum_{j=1}^{r} p_{(i-1)r+j}$. From Claim 3.4, we have $d_{TV}((p_1,\ldots,p_k), Q/(2d)) = O(k/d)$, where $Q = (Q_1,\ldots,Q_k)$ is a random variable in $\mathbb{R}^k$ with i.i.d. χ_2^2 entries. Hence by Fact 2.4 and setting

$$f(x_1,\ldots,x_k) := \left(\sum_{j=1}^{r} x_j, \sum_{j=1}^{r} x_{r+j}, \ldots, \sum_{j=1}^{r} x_{(\ell-1)r+j}\right),$$

we have $d_{TV}((q_1,\ldots,q_\ell), R/(2d)) = O(k/d)$, where we use the fact that the sum of r i.i.d. χ_2^2 random variables is identically distributed to χ_{2r}^2. $\square$

Moreover, a χ_{2r}^2 random variable is the sum of r i.i.d. absolutely continuous random variables. Hence, relying on the aforementioned central limit theorem, it is statistically close to a Gaussian distribution.

Lemma 3.6. *$d_{TV}((q_1,\ldots,q_\ell), Z/(2d)) = O(k/d) + O(\ell/\sqrt{r})$ where Z is a random variable in $\mathbb{R}^\ell$ with i.i.d. $\mathcal{N}(2r, 4r)$ entries, i.e., $Z/(2d)$ has i.i.d. $\mathcal{N}(r/d, r/d^2)$ entries.*

Proof. By Corollary 2.10 and hybrids over every coordinate for $i \in [\ell]$, we have $d_{TV}(R/(2d), Z/(2d)) = O(\ell/\sqrt{r})$, where R is defined in Claim 3.5. Together with Claim 3.5 finishes the proof. $\square$

Now, we are ready to prove the correctness and the statistical closeness to uniform properties.

Correctness. First, define the function $f : \mathcal{D}(\mathbb{C}^{d(\lambda)}) \to \{0,1\}^{\ell(\lambda)}$ associated with the extractor as

$$f(\sigma) := \mathsf{Round}(\sigma).$$

Due to the continuous nature of quantum states, it is impossible to discretize them perfectly. For any $\sigma \in \mathcal{D}(\mathbb{C}^d)$, consider the corresponding $q_1,\ldots,q_\ell$ defined in Construction 3.2. If *all* $q_1,\ldots,q_\ell$ are sufficiently away from r/d, then the extractor is able to output the correct string with high probability by the correctness of $\mathsf{Tomography}$. Here, we define the set $\mathcal{G}_\Delta$ of "good states" whose $q_1,\ldots,q_\ell$ are *all* Δ-away from r/d (the parameter $\Delta(\lambda)$ will be chosen later). The following claim characterizes the probability of a Haar random state being in $\mathcal{G}_\Delta$.

Claim 3.7. *Let the set* $\mathcal{G}_\Delta \subseteq \mathcal{S}(\mathbb{C}^{d(\lambda)})$ *be*

$$\mathcal{G}_\Delta := \left\{ |\psi\rangle \in \mathcal{S}(\mathbb{C}^{d(\lambda)}) : \forall i \in [\ell],\ \left| q_i - \frac{r}{d} \right| > \Delta \right\},$$

where each q_i *is defined on the matrix* $|\psi\rangle\langle\psi|$. *It holds that*

$$\Pr\left[|\psi\rangle \in \mathcal{G}_\Delta : |\psi\rangle \leftarrow \mathscr{H}(\mathbb{C}^{d(\lambda)}) \right] \geq 1 - O\left(\frac{k}{d}\right) - O\left(\frac{\ell}{\sqrt{r}}\right) - O\left(\frac{\Delta \ell d}{\sqrt{r}}\right).$$

Proof. By Lemma 3.6, the total variation distance between $(q_1, \ldots, q_\ell)$ and the random variable $Z = (Z_1, \ldots, Z_\ell)$ with i.i.d. Gaussian entries $Z_i \sim \mathcal{N}(r/d, r/d^2)$ is $O(k/d) + O(\ell/\sqrt{r})$. Hence,

$$\begin{aligned} \Pr\left[|\psi\rangle \in \mathcal{G}_\Delta : |\psi\rangle \leftarrow \mathscr{H}(\mathbb{C}^{d(\lambda)}) \right] &= \Pr\left[\forall i \in [\ell],\ \left| q_i - \frac{r}{d} \right| > \Delta \right] \\ &\geq \Pr\left[\forall i \in [\ell],\ \left| Z_i - \frac{r}{d} \right| > \Delta \right] - O\left(\frac{k}{d}\right) - O\left(\frac{\ell}{\sqrt{r}}\right). \end{aligned}$$

Moreover, by Fact 2.14, for every coordinate $i \in [\ell]$, it holds that

$$\Pr\left[\left| Z_i - \frac{r}{d} \right| \leq \Delta \right] \leq O\left(\frac{\Delta}{\sqrt{r}/d}\right).$$

By a union bound over $i \in [\ell]$, with all but $O\left(\frac{\Delta \ell d}{\sqrt{r}}\right)$ probability every Z_i is Δ-away from r/d. Collecting the probabilities completes the proof. □

Hence, by setting $\Delta(\lambda) = 1/d(\lambda)$, the choice of parameters $r(\lambda) = d(\lambda)^{2/3}$, $\ell(\lambda) = d(\lambda)^{1/6}$, $k(\lambda) = d(\lambda)^{5/6}$ and Claim 3.7, we have

$$\Pr\left[|\psi\rangle \in \mathcal{G}_\Delta : |\psi\rangle \leftarrow \mathscr{H}(\mathbb{C}^{d(\lambda)}) \right] \geq 1 - O(d(\lambda)^{-1/6}).$$

Next, given a state which is in $\mathcal{G}_\Delta$, the output bitstring extracted from it will be $f(|\psi\rangle\langle\psi|)$ with overwhelming probability by the correctness of $\mathsf{Tomography}$ in Corollary 2.16.

Claim 3.8. *If* $|\psi\rangle \in \mathcal{G}_\Delta$, *then running* Ext *in Construction 3.2 with error tolerance* $\delta(\lambda) = d^{-5/3} = \Delta(\lambda)/r(\lambda)$ *for* $\mathsf{Tomography}$ *satisfies*

$$\Pr\left[y = f(|\psi\rangle\langle\psi|) : y \leftarrow \mathsf{Ext}\left(|\psi\rangle\langle\psi|^{\otimes t(\lambda)} \right) \right] \geq 1 - \mathsf{negl}(\lambda),$$

where the probability is over the randomness of the extractor Ext.

Proof. Let M be the classical description obtained by running $\mathsf{Tomography}(|\psi\rangle\langle\psi|^{\otimes t(\lambda)})$ with error tolerance δ and $t \geq 144\lambda d^8 \geq 144\lambda d^4/\delta^2$. Let $\hat{p}_i$'s and $\hat{q}_j$'s be the corresponding diagonal entries and sums of M. By Corollary 2.16 , $\mathrm{TD}(|\psi\rangle\langle\psi|, M) \leq \delta$ holds with overwhelming probability. For the rest of the proof, we assume that this event holds. Then by Fact 2.3, we have $|p_i - \hat{p}_i| \leq \delta$ for every $i \in [k]$. Since $|\psi\rangle \in \mathcal{G}_\Delta$, we have $|q_i - r/d| > \Delta$ for every $i \in [\ell]$. We now show that if $q_i > r/d + \Delta$, then $\hat{q}_i > r/d$. For every $i \in [\ell]$, by the triangle inequality and the fact that $\delta = \Delta/r$, we have

$$\hat{q}_i = q_i - (q_i - \hat{q}_i) > \left(\frac{r}{d} + \Delta\right) - \sum_{j=1}^{r} \left| p_{(i-1)r+j} - \hat{p}_{(i-1)r+j} \right| \geq \frac{r}{d} + \Delta - r \cdot \delta = \frac{r}{d}.$$

Similarly, we have $q_i < r/d - \Delta$ implies that $\hat{q}_i < r/d$. Hence, this ensures the consistency between $\mathsf{Round}(M)$ and $\mathsf{Round}(|\psi\rangle\langle\psi|)$ and completes the proof. □

Statistical Closeness to Uniformity. We finish the proof with a hybrid argument:

- H_1 : In the first hybrid, the output is generated according to Construction 3.2.
  1. Sample $|\psi\rangle \leftarrow \mathscr{H}(\mathbb{C}^{d(\lambda)})$.
  2. Perform $\mathsf{Tomography}(\rho^{\otimes t(\lambda)})$ with $t(\lambda) := \lceil 144\lambda d(\lambda)^8 \rceil$ and error tolerance $\delta(\lambda) := d(\lambda)^{-5/3}$ to get the classical description $M \in \mathbb{C}^{d(\lambda)\times d(\lambda)}$ that approximates ρ.
  3. Output $y = \mathsf{Round}(M)$.
- H_2 : In the second hybrid, the input of Round is changed to the exact description of the quantum state.
  1. Sample $|\psi\rangle \leftarrow \mathscr{H}(\mathbb{C}^{d(\lambda)})$.
  2. Output $y = \mathsf{Round}(|\psi\rangle\langle\psi|)$.
- H_3 : In the third hybrid, the output is generated by rounding i.i.d. Gaussians.
  1. Sample $z_1, \ldots, z_\ell \leftarrow \mathcal{N}(r/d, r/d^2)$.
  2. For $i \in [\ell]$,
  $$b_i = \begin{cases} 0, & \text{if } z_i < r/d \\ 1, & \text{if } z_i > r/d. \end{cases}$$
  3. Output $b_1 || \ldots || b_{\ell(\lambda)}$.

We can bound the total variation distance between H_1 and H_2 by $O(\delta) = O(d^{-5/3})$ using Corollary 2.16 and our chosen error tolerance. Additionally, the total variation distance between H_2 and H_3 is at most $O(d^{-1/6})$ from Lemma 3.6. Finally, since Gaussians are symmetric about the mean, the output string in H_3 is uniformly and randomly distributed. This finishes the proof. □

4 Quantum PRGs and PRFs

In this section, we present our main application of the extractor in Section 3. We introduce the notion of *pseudodeterministic quantum pseudorandom generators* (QPRGs). As the name suggests, QPRGs is a pseudorandom generator with quantum generation satisfying only *pseudodeterminism* property. To be more precise, by pseudodeterminism we mean that there exist some constant $c > 0$ and at least $1 - O(\lambda^{-c})$ fraction of "good seeds" for which the output is almost certain. That is, for each good seed, the probability (over the randomness of the QPRG) of the most likely output is at least $1 - O(\lambda^{-c})$.

4.1 Construction of QPRGs

Definition 4.1 (Weak/Strong Pseudodeterministic Quantum Pseudorandom Generator)**.** *A weak pseudodeterministic quantum pseudorandom generator G_λ, abbreviated as wQPRG, is a uniform QPT algorithm that on input a seed $k \in \{0,1\}^\lambda$, outputs a bitstring of length $\ell(\lambda)$ with the following guarantees:*

- **Pseudodeterminism:** *There exists a constant $c > 0$ and a function $\mu(\lambda) = O(\lambda^{-c})$ such that for every $\lambda \in \mathbb{N}$, there exists a set of "good seeds" $\mathcal{K}_\lambda \subseteq \{0,1\}^\lambda$ satisfying the following:*

1. $\Pr[k \in \mathcal{K}_\lambda : k \leftarrow \{0,1\}^\lambda] \geq 1 - \mu(\lambda)$.
2. *For any $k \in \mathcal{K}_\lambda$, it holds that*

$$\max_{y \in \{0,1\}^{\ell(\lambda)}} \Pr[y = G_\lambda(k)] \geq 1 - \mu(\lambda),$$

where the probability is over the randomness of G_λ.

- **Stretch:** *The output length of G_λ, namely $\ell(\lambda)$, is strictly greater than λ.*

- **Weak Security:** *For every (non-uniform) QPT distinguisher A, there exists a polynomial $\nu(\cdot)$ such that the following holds for sufficiently large $\lambda \in \mathbb{N}$,*

$$\left|\Pr\left[A_\lambda(y) = 1 : {}^{k \leftarrow \{0,1\}^\lambda,}_{y \leftarrow G_\lambda(k)}\right] - \Pr\left[A_\lambda(y) = 1 : y \leftarrow \{0,1\}^{\ell(\lambda)}\right]\right| \leq 1 - \frac{1}{\nu(\lambda)}, \tag{1}$$

where the probability of the first experiment is over the choice of k and the randomness of G_λ and A_λ.

If G further satisfies the strong security property defined below, we call G a strong pseudodeterministic quantum pseudorandom generator*, abbreviated as sQPRG.*

- **Strong Security:** *For every (non-uniform) QPT distinguisher A, there exists a negligible function $\varepsilon(\cdot)$ such that the following holds for sufficiently large $\lambda \in \mathbb{N}$,*

$$\left|\Pr\left[A_\lambda(y) = 1 : {}^{k \leftarrow \{0,1\}^\lambda,}_{y \leftarrow G_\lambda(k)}\right] - \Pr\left[A_\lambda(y) = 1 : y \leftarrow \{0,1\}^{\ell(\lambda)}\right]\right| \leq \varepsilon(\lambda),$$

where the probability of the first experiment is over the choice of k and the randomness of G_λ and A_λ.

We call the left-hand side of Equation (1) the distinguishing advantage *of A. We say that G is* $(1-\delta(\lambda))$-pseudorandom *or* has pseudorandomness $1-\delta(\lambda)$ *if the maximum distinguishing advantage over all non-uniform QPT adversaries is at most $\delta(\lambda)$. We say that G has pseudodeterminism $1-\mu(\lambda)$ if it satisfies the pseudodeterminism property for the function $\mu(\cdot)$.*

We begin with an $n(\lambda)$-PRS (recall that $n(\lambda)$ is its output length), where $n(\lambda) = O(\log \lambda)$ and the dimension of its output is $d(\lambda) = 2^{n(\lambda)} = \mathsf{poly}(\lambda)$.

Theorem 4.2 ($O(\log \lambda)$-PRS implies wQPRG)**.** *Assuming the existence of $(c \log \lambda)$-PRS for some constant $c > 6$, then there exists a $\left(1 - O(\lambda^{-c/6})\right)$-pseudorandom wQPRG with pseudodeterminism $1 - O(\lambda^{-c/12})$ and output length $\ell(\lambda) = \lambda^{c/6} > \lambda$.*

Proof. Consider the following construction.

Construction 4.3 (Weak Quantum Pseudorandom Generators)**.**

1. *Input: a security parameter 1^λ and a seed $k \in \{0,1\}^\lambda$.*

2. *Run $(c \log \lambda)$-$\mathsf{PRS}(k)$ t times to get $\rho_k^{\otimes t(\lambda)}$, where $t(\lambda) = \lceil 144 \lambda d(\lambda)^8 \rceil = O(\lambda^{8c+1})$ as defined in Construction 3.2.*

3. *Run $\mathsf{Ext}(\rho_k^{\otimes t(\lambda)})$ defined in Construction 3.2 to get $y \in \{0,1\}^{\ell(\lambda)}$.*

4. *Output y.*

Efficiency. Since $t(\lambda) = \mathsf{poly}(\lambda)$ and $d(\lambda) = O(\lambda^c)$, the running time is polynomial in λ from Theorem 3.1.

Pseudodeterminism. We complete the proof by a hybrid argument. Consider the following hybrids.

- H_1 : In the first hybrid, y is generated according to Construction 4.3.
  1. Sample $k \leftarrow \{0,1\}^\lambda$.
  2. Run $\mathsf{PRS}(k)$ t times to get $\rho_k^{\otimes t(\lambda)}$.
  3. Run $y \leftarrow \mathsf{Ext}(\rho_k^{\otimes t(\lambda)})$.
  4. Output y.
- H_2 : In the second hybrid, the input is changed to a Haar state.
  1. Sample $|\psi\rangle \leftarrow \mathscr{H}(\mathbb{C}^{d(\lambda)})$.
  2. Run $y \leftarrow \mathsf{Ext}(|\psi\rangle\langle\psi|^{\otimes t(\lambda)})$.
  3. Output y.

For the sake of contradiction, suppose there exists at least $\mu(\lambda) \neq O(\lambda^{-c/12})$ fraction of "bad seeds" (the complement of the set $\mathcal{K}_\lambda$ of good seeds) for which the probability of the most likely output is at most $1 - \mu(\lambda)$. Then we construct an efficient distinguisher for PRS as follows:

1. Take as input $2t(\lambda) = \mathsf{poly}(\lambda)$ copies of ρ which is either sampled from PRS with a random key or $\mathscr{H}(\mathbb{C}^{d(\lambda)})$.
2. Run $\mathsf{Ext}(\rho^{\otimes t(\lambda)})$ twice independently and get the output y_1, y_2 respectively.

First, if ρ is sampled from $\mathscr{H}(\mathbb{C}^{d(\lambda)})$, then by the correctness of Ext in Theorem 3.1, we have

$$p_1 := \Pr\left[y_1 = y_2 : \rho \leftarrow \mathscr{H}(\mathbb{C}^{d(\lambda)})\right] \geq (1 - O(d(\lambda)^{1/6})) \cdot (1 - \mathsf{negl}(\lambda))^2 \geq 1 - h(\lambda),$$

where $h(\lambda) = O(\lambda^{-c/6})$.
On the other hand, consider the case in which ρ is sampled from PRS. Without loss of generality, we can assume that $\mu(\lambda) < 1/2$ for sufficiently large λ. Otherwise, the distinguishing advantage would already be non-negligible. Then,

$$\begin{aligned} p_2 := \Pr[y_1 = y_2 : \rho \leftarrow \mathsf{PRS}(k)] &= \Pr[k \in \mathcal{K}_\lambda] \Pr[y_1 = y_2 \mid k \in \mathcal{K}_\lambda] + \Pr[k \notin \mathcal{K}_\lambda] \Pr[y_1 = y_2 \mid k \notin \mathcal{K}_\lambda] \\ &\leq (1 - \mu(\lambda)) \cdot 1 + \mu(\lambda) \cdot (1 - \mu(\lambda)) = 1 - \mu(\lambda)^2. \end{aligned}$$

Now, for any sufficiently large $\lambda \in \mathbb{N}$ such that $1/2 > \mu(\lambda)$, we do a case analysis. Suppose $h(\lambda) \geq \mu(\lambda)^2$, then we have $\sqrt{h(\lambda)} > \mu(\lambda)$. Otherwise, if $h(\lambda) < \mu(\lambda)^2$, then the distinguishing advantage $|p_1 - p_2|$ satisfies

$$\mathsf{negl}(\lambda) = |p_1 - p_2| = p_1 - p_2 \geq \mu(\lambda)^2 - h(\lambda)$$

due to the security of PRS. Hence, it holds that $\sqrt{h(\lambda) + \mathsf{negl}(\lambda)} > \mu(\lambda)$. However, combining two cases would imply that $\mu(\lambda) = O(\lambda^{-c/12})$ and lead to a contradiction.

Stretch. From Theorem 3.1, the output length of Construction 4.3 is given by $\ell(\lambda) = d(\lambda)^{1/6} = \lambda^{c/6} > \lambda$ since $c > 6$.

Weak Security. We complete the proof by a hybrid argument. Consider the following hybrids:

- H_1 : In the first hybrid, the adversary receives a string y which is generated according to Construction 4.3.
  1. Sample $k \leftarrow \{0,1\}^\lambda$.
  2. Run $\mathsf{PRS}(k)$ t times to get $\rho_k^{\otimes t(\lambda)}$.
  3. Run $y \leftarrow \mathsf{Ext}(\rho_k^{\otimes t(\lambda)})$.
  4. Output $y \in \{0,1\}^{\ell(\lambda)}$.
- H_2 : In the second hybrid, the input is changed to a Haar state.
  1. Sample $|\psi\rangle \leftarrow \mathscr{H}(\mathbb{C}^{d(\lambda)})$.
  2. Run $y \leftarrow \mathsf{Ext}(|\psi\rangle\langle\psi|^{\otimes t(\lambda)})$.
  3. Output $y \in \{0,1\}^{\ell(\lambda)}$.
- H_3 : Sample $y \leftarrow \{0,1\}^{\ell(\lambda)}$. Output $y \in \{0,1\}^{\ell(\lambda)}$. In the third hybrid, the adversary receives a string sampled from the uniform distribution.

The computational indistinguishability of hybrids H_1 and H_2 follows from the security of PRS. Otherwise, running Ext on the samples once would be an efficient distinguisher in the PRS security experiment. The statistical indistinguishability of hybrids H_2 and H_3 follows from the statistical closeness to uniform property of Theorem 3.1. In particular, the statistical distance is $O(d(\lambda)^{-1/6}) = O(\lambda^{-c/6})$. □

While we do not have a non-trivial way to amplify the pseudodeterminism property, the security amplification can be achieved by techniques in [CHS05, DIJK09, MT09, MT10]. In particular, we will use the security amplification for (classical) weak PRGs in [DIJK09]. The construction is to run the weak PRG G on input $s(\lambda) = \omega(\log \lambda)$ independently and randomly chosen seeds $k_1, \ldots, k_s$ and then output the bit-wise XOR of the s strings $G(k_1), \ldots, G(k_s)$.

Theorem 4.4 ([DIJK09, Theorem 6])**.** *Let $s(\lambda) = \omega(\log \lambda)$. Let $G : \{0,1\}^\lambda \to \{0,1\}^{\ell(\lambda)}$ be a weak PRG with $(1-\delta)$-pseudorandomness such that $\delta < 1/2$ and $\ell(\lambda) > s(\lambda) \cdot \lambda$. Define the function $G^{\oplus s} : \{0,1\}^{s(\lambda)\lambda} \to \{0,1\}^{\ell(\lambda)}$ as $G^{\oplus s}(k_1, \ldots, k_s) := \bigoplus_{i=1}^s G(k_i)$. Then $G^{\oplus s}$ is a strong PRG.*

We observed that Theorem 4.4 could be extended to QPRGs.

Theorem 4.5 (Security amplification for QPRGs)**.** *Let $G : \{0,1\}^\lambda \to \{0,1\}^{\ell(\lambda)}$ be a wQPRG that has pseudodeterminism $1 - O(\lambda^{-c})$ and pseudorandomness $1-\delta$ such that $c > 1$, $\delta(\lambda) \le 0.49 + o(1)$ and $\ell(\lambda) > s(\lambda) \cdot \lambda$, where $s(\lambda) = \Theta(\lambda)$. Define the QPT algorithm $G^{\oplus s} : \{0,1\}^{s(\lambda)\lambda} \to \{0,1\}^{\ell(\lambda)}$ as $G^{\oplus s}(k_1, \ldots, k_s) := \bigoplus_{i=1}^s G(k_i)$. Then $G^{\oplus s}$ is a sQPRG with pseudodeterminism $1 - O(\lambda^{-(c-1)})$ and output length $\ell(\lambda)$.*

Proof sketch. We first prove the pseudodeterminism property. Fix the security parameter λ. The set of good seeds of $G^{\oplus s}$ is defined to be the $s(\lambda)$-fold Cartesian product $\mathcal{K}_\lambda \times \cdots \times \mathcal{K}_\lambda \subseteq \{0,1\}^{s(\lambda)\lambda}$, where $\mathcal{K}_\lambda$ is the set of good seeds of G. By the pseudodeterminism of G and a union bound over $i \in [s]$, we have

$$\Pr\left[\forall i \in [s],\ k_i \in \mathcal{K}_\lambda : k = k_\lambda || \ldots || k_\lambda \leftarrow \{0,1\}^{s(\lambda)\lambda}\right] \geq 1 - O(s/\lambda^c) = 1 - O(\lambda^{-(c-1)}).$$

Next, for every $(k_1, \ldots, k_s) \in \mathcal{K}_\lambda \times \cdots \times \mathcal{K}_\lambda$, there exists some $y_i \in \{0,1\}^{\ell(\lambda)}$ such that $\Pr[y_i = G(k_i)] \geq 1 - 1/\lambda^c$ for every $i \in [s]$. Hence, by a union bound over $i \in [s]$, it holds that

$$\Pr\left[\bigoplus_{i=1}^{s} y_i = G^{\oplus s}(k_1, \ldots, k_s)\right] \geq \Pr\left[\bigwedge_{i=1}^{s} y_i = G(k_i)\right] \geq 1 - O(s/\lambda^c) = 1 - O(\lambda^{-(c-1)}).$$

That is, $G^{\oplus s}$ has pseudodeterminism $1 - O(\lambda^{-(c-1)})$. The full proof of (strong) security is deferred to Appendix A. □

From Theorem 4.2, Theorem 4.5 and picking $s(\lambda) = \lambda$, we have the following corollary.

Corollary 4.6. *Assuming the existence of* $(c \log \lambda)$*-PRS for some constant* $c > 12$*, then there exists a sQPRG* $G^{\oplus \lambda} : \{0,1\}^{\lambda^2} \to \{0,1\}^{\ell(\lambda)}$ *with pseudodeterminism* $1 - O(\lambda^{-(c/12-1)})$ *and output length* $\ell(\lambda) = \lambda^{c/6} > \lambda^2$.

4.2 Construction of Selectively Secure QPRFs

In the same spirit, it is natural to consider the concept of pseudodeterministic quantum pseudorandom functions (QPRFs). However, when the pseudodeterminism is only $1 - O(\lambda^{-c})$, there is a caveat. An attacker that can make *adaptive* queries can easily distinguish a QPRF with this level of pseudodeterminism from a random function as follows: the distinguisher simply queries the oracle on *the same point* polynomially many times and checks if the answers are all the same. A random function will always produce identical outputs, while a QPRF with pseudodeterminism $1 - O(\lambda^{-c})$ will generate different outputs with constant probability. Intuitively, non-determinism allows the QPRF output to appear more random, thus it should strengthen its security.

Below, we show that we can use a selectively secure $(m(\lambda), n(\lambda))$-PRFS, where $m(\lambda) = \omega(\log \lambda)$ and $n(\lambda) = O(\log \lambda)$, to construct a selectively secure QPRF with input length $m(\lambda)$ and output length $\mathsf{poly}(\lambda)$.

Definition 4.7 (Selectively Secure Quantum Pseudorandom Functions)**.** *A selectively secure quantum pseudorandom function* $F : \{0,1\}^\lambda \times \{0,1\}^{m(\lambda)} \to \{0,1\}^{\ell(\lambda)}$ *is a QPT algorithm with the following guarantees:*

- **Pseudodeterminism:** *There exists a constant* $c > 0$ *and a function* $\mu(\lambda) = O(\lambda^{-c})$ *such that for every* $\lambda \in \mathbb{N}$ *and every* $x \in \{0,1\}^{m(\lambda)}$*, there exists a set of "good keys"* $\mathcal{K}_{\lambda,x} \subseteq \{0,1\}^\lambda$ *satisfying the following:*

  1. $\Pr[k \in \mathcal{K}_{\lambda,x} : k \leftarrow \{0,1\}^\lambda] \geq 1 - \mu(\lambda)$.

2. *For any $k \in \mathcal{K}_{\lambda,x}$, it holds that*

$$\max_{y \in \{0,1\}^{\ell(\lambda)}} \Pr[y = F(k,x)] \geq 1 - \mu(\lambda),$$

where the probability is over the randomness of F.

- **Selective Security:** *For any polynomial $q(\cdot)$, any (non-uniform) QPT distinguisher A and any family of pairwise distinct indices $\left(\left\{x_1, \ldots, x_{q(\lambda)}\right\} \subseteq \{0,1\}^{m(\lambda)}\}\right)_\lambda$, there exists a negligible function $\varepsilon(\cdot)$ such that for all $\lambda \in \mathbb{N}$,*

$$\Bigg| \Pr\left[A_\lambda(x_1, \ldots, x_{q(\lambda)}, y_1, \ldots, y_{q(\lambda)}) = 1 : {}^{k \leftarrow \{0,1\}^\lambda,}_{y_1 \leftarrow F(k,x_1), \ldots, y_{q(\lambda)} \leftarrow F(k, x_{q(\lambda)})}\right]$$
$$- \Pr\left[A_\lambda(x_1, \ldots, x_{q(\lambda)}, y_1, \ldots, y_{q(\lambda)}) = 1 : y_1, \ldots, y_{q(\lambda)} \leftarrow \{0,1\}^{\ell(\lambda)}\right] \Bigg| \leq \varepsilon(\lambda).$$

We will construct a selectively secure quantum pseudorandom function based on a selectively secure $(m(\lambda), n(\lambda))$-PRFS, where $m(\lambda) = \omega(\log \lambda)$ and $n(\lambda) = O(\log \lambda)$.

Theorem 4.8 ($(\omega(\log \lambda), O(\log \lambda))$-PRFS implies selectively secure QPRF)**.** *Assuming the existence of selectively secure $(m(\lambda), c \log \lambda)$-PRFS for some constant $c > 12$ and $m(\lambda) = \omega(\log \lambda)$, then there exists a selectively secure QPRF $F : \{0,1\}^{\lambda^2} \times \{0,1\}^{m(\lambda)} \to \{0,1\}^{\ell(\lambda)}$ with pseudodeterminism $1 - O(\lambda^{-(c/12-1)})$, input length $m(\lambda)$ and output length $\ell(\lambda) = \lambda^{c/6}$.*

Proof. Consider the following construction:

Construction 4.9 (Selectively Secure Quantum Pseudorandom Functions)**.**

1. *Input: a key $k \in \{0,1\}^{\lambda^2}$ and input $x \in \{0,1\}^{m(\lambda)}$.*
2. *Parse k as $k_1 || \ldots || k_\lambda$ such that $k_i \in \{0,1\}^\lambda$ for every $i \in [\lambda]$.*
3. *For $i \in [\lambda]$, run $\mathsf{PRFS}(k_i, x)$ to get $\rho_{k_i,x}^{\otimes t(\lambda)}$, where $t(\lambda) = \lceil 144 \lambda d(\lambda)^8 \rceil$.*
4. *For $i \in [\lambda]$, run $\mathsf{Ext}(\rho_{k_i,x}^{\otimes t(\lambda)})$ to get $y_i \in \{0,1\}^{\ell(\lambda)}$.*
5. *Let $y = \bigoplus_{i=1}^{\lambda} y_i$, output $y \in \{0,1\}^{\ell(\lambda)}$.*

Pseudodeterminism. We complete the proof by a hybrid argument. For any fixed $x \in \{0,1\}^{m(\lambda)}$, consider the following hybrids.

- H_1 : In the first hybrid, y is generated according to Construction 4.9.
  1. Sample $k \leftarrow \{0,1\}^{\lambda^2}$.
  2. Parse k as $k_1 || \ldots || k_\lambda$ such that $k_i \in \{0,1\}^\lambda$ for every $i \in [\lambda]$.
  3. For $i \in [\lambda]$, run $\mathsf{PRFS}(k_i, x)$ t times to get $\rho_{k_i,x}^{\otimes t(\lambda)}$.

4. For $i \in [\lambda]$, run $\mathsf{Ext}(\rho_{k_i,x}^{\otimes t(\lambda)})$ to get $y_i \in \{0,1\}^{\ell(\lambda)}$.
5. Let $y = \bigoplus_{i=1}^{\lambda} y_i$, output $y \in \{0,1\}^{\ell(\lambda)}$.

- H_2 : In the second hybrid, the input of Ext is changed to Haar states.
    1. For $i \in [\lambda]$, sample $|\psi_i\rangle \leftarrow \mathscr{H}(\mathbb{C}^{d(\lambda)})$.
    2. For $i \in [\lambda]$, run $\mathsf{Ext}(|\psi_i\rangle\langle\psi_i|^{\otimes t(\lambda)})$ to get $y_i \in \{0,1\}^{\ell(\lambda)}$.
    3. Let $y = \bigoplus_{i=1}^{\lambda} y_i$, output $y \in \{0,1\}^{\ell(\lambda)}$.

Similar to proving pseudodeterminism in Theorem 4.2, there exists at least $1 - O(\lambda^{-c/12})$ fraction of good keys $\mathcal{K}'_{\lambda,x} \subseteq \{0,1\}^{\lambda}$ such that for any $k \in \mathcal{K}'_{\lambda,x}$,

$$\max_{y \in \{0,1\}^{\ell(\lambda)}} \Pr[y = \mathsf{Ext}(\mathsf{PRFS}(k,x)^{\otimes t})] \geq 1 - O(\lambda^{-c/12}),$$

where the probability is over the randomness of Ext. Otherwise, running Ext independently twice on input $\mathsf{PRFS}(k,x)^{\otimes t}$ and comparing the output would be an efficient distinguisher that contradicts the security of PRFS. The set of good keys for F is defined to be the λ-fold Cartesian product $\mathcal{K}'_{\lambda,x} \times \cdots \times \mathcal{K}'_{\lambda,x} \subseteq \{0,1\}^{\lambda^2}$. Then following the same lines for proving pseudodeterminism in Theorem 4.5, we can conclude that Construction 4.9 has pseudodeterminism $1 - O(\lambda^{-(c/12-1)})$.

Selective Security. Before we prove the security, we introduce a simple lemma regarding the indistinguishability of polynomially many samples of Haar states and the output of a PRS generator with i.i.d. uniform seeds.

Lemma 4.10. *Let PRS be an $n(\cdot)$-PRS. Then for any polynomials $t(\cdot), p(\cdot)$ and any QPT distinguisher A, there exists a negligible function $\varepsilon(\cdot)$ such that*

$$\Pr\left[A\left(\rho_1^{\otimes t(\lambda)}, \ldots, \rho_{q(\lambda)}^{\otimes t(\lambda)}\right) = 1 : \substack{k_1,\ldots,k_{q(\lambda)} \leftarrow \{0,1\}^{\lambda}, \\ \rho_1 \leftarrow \mathsf{PRS}(k_1),\ldots,\rho_{q(\lambda)} \leftarrow \mathsf{PRS}(k_{q(\lambda)})}\right] \\ - \Pr\left[A\left(|\vartheta_1\rangle^{\otimes t(\lambda)}, \ldots, |\vartheta_{q(\lambda)}\rangle^{\otimes t(\lambda)}\right) = 1 : |\vartheta_1\rangle, \ldots, |\vartheta_{q(\lambda)}\rangle \leftarrow \mathscr{H}_{n(\lambda)}\right] \leq \varepsilon(\lambda).$$

Proof. Consider the following hybrids H_i for $i \in \{0, 1, \ldots, q\}$:

1. For $1 \leq j \leq i$, sample $k_j \leftarrow \{0,1\}^{\lambda}$ and run $\mathsf{PRS}(k_j)$ t times to get $\rho_j^{\otimes t}$.
2. For $i+1 \leq j \leq q$, sample $|\vartheta_j\rangle \leftarrow \mathscr{H}_{n(\lambda)}$.
3. Output $\left(\rho_1^{\otimes t(\lambda)}, \ldots, \rho_i^{\otimes t(\lambda)}, |\vartheta_{i+1}\rangle^{\otimes t(\lambda)}, \ldots, |\vartheta_{q(\lambda)}\rangle^{\otimes t(\lambda)}\right)$.

It is sufficient to prove the computational indistinguishability between H_i and H_{i+1}. Note that the only difference is the $(i+1)$-th coordinate of the sample. Suppose there exist polynomials $t(\cdot), q(\cdot)$ and a QPT adversary A that has a non-negligible advantage for distinguishing H_i from H_{i+1}. Based on A, we will construct a reduction R to break the security of PRS. The reduction R is defined as follows:

1. Input: $\sigma^{\otimes t(\lambda)}$ where σ is sampled from either $\mathsf{PRS}(k)$ with a random k or $\mathscr{H}_{n(\lambda)}$.

2. For $1 \leq j \leq i$, sample $k_j \leftarrow \{0,1\}^\lambda$ and run $\mathsf{PRS}(k_j)$ t times to get $\rho_j^{\otimes t(\lambda)}$.
3. For $i+2 \leq j \leq q$, sample a $t(\lambda)$-state design γ_j.[5]
4. Run $A\left(\rho_1^{\otimes t(\lambda)}, \dots, \rho_i^{\otimes t(\lambda)}, \sigma^{\otimes t(\lambda)}, \gamma_{i+2}, \dots, \gamma_q\right)$ and output whatever A outputs.

First, the running time R is polynomial in λ. Moreover, R perfectly simulates the view of A and thus has the same distinguishing advantage as that of A. However, this contradicts the security of PRS. $\square$

We complete the proof of selective security by hybrid arguments. Here, we outline the structure of the hybrids: H_1 is Construction 4.9. In $\mathsf{H}_{1.i}$ for $i \in \{0,1,\dots,\lambda\}$, we replace the output of $\mathsf{PRFS}(k_i,\cdot)$ with independent Haar states. The computational indistinguishability between $\mathsf{H}_{1.i}$ and $\mathsf{H}_{1.i+1}$ follows from the selective security of PRFS. Finally, in H_2, all the input quantum states of the extractor Ext are now independent Haar states. It remains to show that the resulting output strings are computationally indistinguishable from independent, uniform strings. Fortunately, we observe that we can recycle the proof of strong security of QPRGs in Theorem 4.5 as follows. In H_3, all the independent Haar states are replaced with the output of PRS with i.i.d. uniform seeds. The computational indistinguishability between H_2 and H_3 follows from Lemma 4.10. However, the description of H_3 is exactly the same as running the strong QPRG $G^{\oplus s}$ defined in Theorem 4.5 on i.i.d. uniform seeds. Hence, the output strings are computationally indistinguishable from independent, uniform strings due to the strong security of $G^{\oplus s}$. Formally, consider the following hybrids:

- H_1 : In the first hybrid, the adversary receives input-output pairs according to the selective security experiment and Construction 4.9.
  1. Receive $x_1, \dots, x_{q(\lambda)} \in \{0,1\}^{m(\lambda)}$ from the adversary.
  2. For $j \in [\lambda]$, sample $k_j \leftarrow \{0,1\}^\lambda$.
  3. For $i \in [q]$, do the following,
     (a) For $j \in [\lambda]$, run $\mathsf{PRFS}(k_j, x_i)$ to get $\rho_{k_j,x_i}^{\otimes t(\lambda)}$, and run $\mathsf{Ext}(\rho_{k_j,x_i}^{\otimes t(\lambda)})$ to get $y_{i,j} \in \{0,1\}^{\ell(\lambda)}$.
     (b) Let $y_i = \bigoplus_{j=1}^{\lambda} y_{i,j} \in \{0,1\}^{\ell(\lambda)}$.
  4. Output $(x_1,y_1),\dots,(x_q,y_q)$.
- $\mathsf{H}_{1.a}$ for $a \in \{0,1,\dots,\lambda\}$:
  1. Receive $x_1, \dots, x_{q(\lambda)} \in \{0,1\}^{m(\lambda)}$ from the adversary.
  2. For every $j \in \{a+1,\dots,\lambda\}$, sample $k_j \leftarrow \{0,1\}^\lambda$.
  3. For $i \in [q]$, do the following,
     (a) For $j \in \{1,\dots,a\}$, sample $|\psi_{i,j}\rangle \leftarrow \mathscr{H}(\mathbb{C}^{d(\lambda)})$, and run $\mathsf{Ext}(|\psi_{i,j}\rangle\langle\psi_{i,j}|^{\otimes t(\lambda)})$ to get $y_{i,j} \in \{0,1\}^{\ell(\lambda)}$.

[5] Note that is not efficient for the security reduction to sample Haar random states in each hybrid. Instead of sampling Haar random states, the security reduction uses $t(\lambda)$-state designs. It is known that $t(\lambda)$-state designs can be efficiently generated (in time polynomial in $t(\lambda)$) [AE07, DCEL09].

(b) For $j \in \{a+1, \ldots, \lambda\}$, run $\mathsf{PRFS}(k_j, x_i)$ to get $\rho_{k_j,x_i}^{\otimes t(\lambda)}$, and run $\mathsf{Ext}(\rho_{k_j,x_i}^{\otimes t(\lambda)})$ to get $y_{i,j} \in \{0,1\}^{\ell(\lambda)}$.

(c) Let $y_i = \bigoplus_{j=1}^{\lambda} y_{i,j} \in \{0,1\}^{\ell(\lambda)}$.

4. Output $(x_1, y_1), \ldots, (x_q, y_q)$.

- H_2 : In the second hybrid, all the input of Ext is changed to Haar random states.
  1. Receive $x_1, \ldots, x_{q(\lambda)} \in \{0,1\}^{m(\lambda)}$ from the adversary.
  2. For $i \in [q]$, do the following,
     (a) For $j \in [\lambda]$, sample $|\psi_{i,j}\rangle \leftarrow \mathscr{H}(\mathbb{C}^{d(\lambda)})$ and run $\mathsf{Ext}(|\psi_{i,j}\rangle\langle\psi_{i,j}|^{\otimes t(\lambda)})$ to get $y_{i,j} \in \{0,1\}^{\ell(\lambda)}$.
     (b) Let $y_i = \bigoplus_{j=1}^{\lambda} y_{i,j} \in \{0,1\}^{\ell(\lambda)}$.
  3. Output $(x_1, y_1), \ldots, (x_q, y_q)$.

- H_3 : In the third hybrid, all the input of Ext is changed to the output of an $n(\lambda)$-PRS.
  1. Receive $x_1, \ldots, x_{q(\lambda)} \in \{0,1\}^{m(\lambda)}$ from the adversary.
  2. For $i \in [q]$, $j \in [\lambda]$, sample $k_{i,j} \leftarrow \{0,1\}^{\lambda}$.
  3. For $i \in [q]$, do the following,
     (a) For $j \in [\lambda]$, run $\mathsf{PRS}(k_{i,j})$ t times to get $\rho_{k_{i,j}}^{\otimes t(\lambda)}$, and run $\mathsf{Ext}(\rho_{k_{i,j}}^{\otimes t(\lambda)})$ to get $y_{i,j} \in \{0,1\}^{\ell(\lambda)}$.
     (b) Let $y_i = \bigoplus_{j=1}^{\lambda} y_{i,j} \in \{0,1\}^{\ell(\lambda)}$.
  4. Output $(x_1, y_1), \ldots, (x_q, y_q)$.

- H_4 : In the fourth hybrid, each y_i is the output of the sQPRG defined in Theorem 4.5 with $s(\lambda)$ set to be λ, where the underlying wQPRG is defined to be the one in Construction 4.3.
  1. Receive $x_1, \ldots, x_{q(\lambda)} \in \{0,1\}^{m(\lambda)}$ from the adversary.
  2. For $i \in [q]$, sample $k_i \leftarrow \{0,1\}^{\lambda^2}$.
  3. For $i \in [q]$, run $y_i \leftarrow \mathsf{sQPRG}(k_i)$.
  4. Output $(x_1, y_1), \ldots, (x_q, y_q)$.

- H_5 : In the last hybrid, the adversary receives independently and uniformly sampled query-answer pairs.
  1. Receive $x_1, \ldots, x_{q(\lambda)} \in \{0,1\}^{m(\lambda)}$ from the adversary.
  2. Sample $y_1, \ldots, y_{q(\lambda)} \leftarrow \{0,1\}^{\ell(\lambda)}$.
  3. Output $(x_1, y_1), \ldots, (x_q, y_q)$.

Hybrids H_1 and $\mathsf{H}_{1,0}$ are identically distributed. For $a \in [\lambda]$, hybrids $\mathsf{H}_{1,a-1}$ and $\mathsf{H}_{1,a}$ are computationally indistinguishable due to the selective security of PRFS. Formally, suppose there exists some $a \in [\lambda]$ and a QPT adversary A that can distinguish $\mathsf{H}_{1,a-1}$ from $\mathsf{H}_{1,a}$ with non-negligible advantage. We construct a reduction R that breaks the selective security of PRFS as follows:[6]

[6] Recall that in the definition of selective security, the indices are required to be pairwise distinct.

1. Receive $x_1, \ldots, x_{q(\lambda)} \in \{0,1\}^{m(\lambda)}$ from the adversary A.

2. For every $j \in \{a+1, \ldots, \lambda\}$, sample $k_j \leftarrow \{0,1\}^\lambda$.

3. Send $x_1, \ldots, x_q$ to the challenger and receive $\sigma_1^{\otimes t(\lambda)}, \ldots, \sigma_q^{\otimes t(\lambda)}$, where each σ_i is sampled either from $\mathsf{PRFS}(k, x_i)$ with the same uniformly random key or $\mathscr{H}(\mathbb{C}^d(\lambda))$.

4. For $i \in [q]$, do the following,

   (a) For $j \in \{1, \ldots, a-1\}$, sample a $t(\lambda)$-state design $\gamma_{i,j}$ and run $\mathsf{Ext}(\gamma_{i,j})$ to get $y_{i,j}$.

   (b) Run $\mathsf{Ext}(\sigma_i^{\otimes t(\lambda)})$ to get $y_{i,a}$.

   (c) For $j \in \{a+1, \ldots, \lambda\}$, run $\mathsf{PRFS}(k_j, x_i)$ t times to get $\rho_{k_j,x_i}^{\otimes t(\lambda)}$, and run $\mathsf{Ext}(\rho_{k_j,x_i}^{\otimes t(\lambda)})$ to get $y_{i,j}$.

   (d) Let $y_i = \bigoplus_{j=1}^{\lambda} y_{i,j}$.

5. Run $A((x_1, y_1), \ldots, (x_q, y_q))$ and output whatever A outputs.

The reduction R perfectly simulates A's view. Hence, the distinguishing advantage of R is non-negligible, which leads to a contradiction. Hybrids $\mathsf{H}_{1,\lambda}$ and H_2 are identically distributed. The computational indistinguishability of hybrids H_2 and H_3 follows from the security of PRS. In particular, polynomially many samples from PRS with independent, uniform keys are computationally indistinguishable from i.i.d. Haar states as shown in Lemma 4.10. Suppose there exists a QPT adversary A that has a non-negligible advantage for distinguishing H_2 from H_3. We construct a reduction R that contradicts Lemma 4.10 as follows:

1. Input: $\left\{\sigma_{i,j}^{\otimes t(\lambda)}\right\}_{i\in[q], j\in[\lambda]}$ where all the samples are sampled either from $\mathsf{PRS}(k_{i,j})$ with i.i.d. uniform keys or $\mathscr{H}_{n(\lambda)}$. Note that the number of samples is $q(\lambda) \cdot \lambda = \mathsf{poly}(\lambda)$.

2. Receive $x_1, \ldots, x_{q(\lambda)} \in \{0,1\}^{m(\lambda)}$ from the adversary A.

3. For $i \in [q]$, do the following,

   (a) For $j \in [\lambda]$, run $\mathsf{Ext}(\sigma_{i,j}^{\otimes t(\lambda)})$ to get $y_{i,j}$.

   (b) Let $y_i = \bigoplus_{j=1}^{\lambda} y_{i,j}$.

4. Run $A((x_1, y_1), \ldots, (x_q, y_q))$ and output whatever A outputs.

Since R runs in polynomial time and has the same distinguishing advantage as that of A, this contradicts Lemma 4.10. Hybrids H_3 and H_4 are syntactically identical. Finally, the computationally indistinguishability between H_4 and H_5 follows from the strong security of sQPRG. To be more precise, similar to classical secure PRGs, polynomially many samples of either the output of a sQPRG with i.i.d. uniform seeds or i.i.d. uniform bitstrings are computationally indistinguishable. The proof is similar to that of Lemma 4.10. $\square$

5 Applications

In this section, we present applications based on sQPRGs and selectively secure QPRFs introduced in Section 4. One key advantage of using sQPRGs or selectively secure QPRFs as the starting point is that we can build higher-level primitives simply by following the classical construction with a slight modification, and then security will follow from the same reasoning. However, we must address the issue of correctness since sQPRGs and selectively secure QPRFs have only $1 - O(\lambda^{-c})$ pseudodeterminism. To resolve the issue, we apply a simple parallel repetition followed by a majority vote to boost correctness at the expense of increased communication complexity and key length.

5.1 Pseudorandom One-Time Pad (POTP)

We construct a pseudorandom one-time pad (POTP) scheme with classical communication from sQPRGs. A POTP is an encryption scheme in which the message length is strictly greater than the key length.

Definition 5.1. *A* pseudorandom one-time pad (POTP) *for messages of length $\ell(\lambda)$ is a triple of QPT algorithms* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *such that the following holds:*

- **Correctness:** *There exists a negligible function $\varepsilon(\cdot)$ such that for every $\lambda \in \mathbb{N}$ and every message $m \in \{0,1\}^{\ell(\lambda)}$,*

$$\Pr\left[m = m' : \substack{k \leftarrow \mathsf{Gen}(1^\lambda), \\ c \leftarrow \mathsf{Enc}(1^\lambda, k, m), \\ m' \leftarrow \mathsf{Dec}(1^\lambda, k, c)}\right] \geq 1 - \varepsilon(\lambda).$$

- **Stretch:** *For every $\lambda \in \mathbb{N}$, $\ell(\lambda) > k(\lambda)$, where $k(\lambda)$ is the output length of $\mathsf{Gen}(1^\lambda)$, i.e., the key length.*

- **Security:** *For any (non-uniform) QPT adversary A, there exists a negligible function $\varepsilon(\cdot)$ such that for every $m_0, m_1 \in \{0,1\}^{\ell(\lambda)}$ and $\lambda \in \mathbb{N}$,*

$$\left|\Pr\left[A_\lambda(c) = 1 : \substack{k \leftarrow \mathsf{Gen}(1^\lambda), \\ c \leftarrow \mathsf{Enc}(1^\lambda, k, m_0)}\right] - \Pr\left[A_\lambda(c) = 1 : \substack{k \leftarrow \mathsf{Gen}(1^\lambda), \\ c \leftarrow \mathsf{Enc}(1^\lambda, k, m_1)}\right]\right| \leq \varepsilon(\lambda).$$

Suppose $G_\lambda : \{0,1\}^\lambda \to \{0,1\}^{\ell(\lambda)}$ is a sQPRG with output length $\ell(\lambda) > \lambda^2$ and pseudodeterminism $1 - O(\lambda^{-c})$ for arbitrary $c > 0$. Consider the following construction:

Construction 5.2 (Pseudorandom One-Time Pad (POTP))**.**

- $\mathsf{Gen}(1^\lambda)$: *on input 1^λ, outputs a key $k \leftarrow \{0,1\}^{\lambda^2}$.*

- $\mathsf{Enc}(1^\lambda, k, m)$: *on input 1^λ, a key $k \in \{0,1\}^{\lambda^2}$ and a message $m \in \{0,1\}^{\ell(\lambda)}$,*

  1. *Parse k as $k_1 || \ldots || k_\lambda$ such that $k_i \in \{0,1\}^\lambda$ for every $i \in [\lambda]$.*
  2. *For $i \in [\lambda]$, compute $c_i := m \oplus G_\lambda(k_i) \in \{0,1\}^{\ell(\lambda)}$.*
  3. *Output $c := c_1 || \ldots || c_\lambda \in \{0,1\}^{\lambda\ell(\lambda)}$.*

- $\mathsf{Dec}(1^\lambda, k, c)$: *on input 1^λ, a key $k \in \{0,1\}^{\lambda^2}$ and a ciphertext $c \in \{0,1\}^{\lambda\ell(\lambda)}$,*

1. *Parse k as $k_1||\dots||k_\lambda$ and c as $c_1||\dots||c_\lambda$ such that $k_i \in \{0,1\}^\lambda$ and $c_i \in \{0,1\}^{\ell(\lambda)}$ for all $i \in [\lambda]$.*
2. *For $i \in [\lambda]$, compute $m_i := c_i \oplus G_\lambda(k_i) \in \{0,1\}^{\ell(\lambda)}$.*
3. *Output $m := \mathsf{MAJ}(m_1,\dots,m_\lambda)$, where MAJ denotes the majority function.*

For the stretch property, the message length of Construction 5.2 satisfies $\ell(\lambda) > k(\lambda) = \lambda^2$.

Lemma 5.3. *Construction 5.2 satisfies the correctness property.*

Proof. On input a random key $k = k_1||\dots||k_\lambda \in \{0,1\}^{\lambda^2}$, for every $i \in [\lambda]$ we have $\Pr[k_i \in \mathcal{K}_\lambda] \geq 1 - O(\lambda^{-c}) > 0.9$ for sufficiently large λ by the pseudodeterminism of G, where $\mathcal{K}_\lambda$ is the set of good seeds of G. We denote by Good the event that at least 0.8λ of the k_i's belong in $\mathcal{K}_\lambda$. For each k_i, let $G_E(k_i)$ and $G_D(k_i)$ denote the output of G evaluated by Enc and Dec, respectively. If $k_i \in \mathcal{K}_\lambda$, then the probability that $G_E(k_i) = G_D(k_i)$ is at least $(1 - O(\lambda^{-c}))^2$. Hence, the success probability of the majority vote is at least

$$\begin{aligned}&\Pr_{k \leftarrow \{0,1\}^{\lambda^2}}\left[|\{i \in [\lambda] : G_E(k_i) = G_D(k_i)\}| > \lambda/2\right]\\ \geq\ &\Pr[\mathsf{Good}] \cdot \Pr_{k \leftarrow \{0,1\}^{\lambda^2}}\left[|\{i \in [\lambda] : G_E(k_i) = G_D(k_i)\}| > \lambda/2 \mid \mathsf{Good}\right],\end{aligned}$$

where the probability is over k, G_E and G_D.

First, $\Pr[\mathsf{Good}] = 1 - 2^{-\Omega(\lambda)}$ by Lemma 2.5. Moreover, conditioned on the event Good happening, the expected number of i's such that $G_E(k_i) = G_D(k_i)$ is at least $0.8\lambda \cdot (1 - O(\lambda^{-c}))^2 > 0.7\lambda$ for sufficiently large λ. Using Lemma 2.5 again, the result of the majority vote is correct with probability at least $1 - \mathsf{negl}(\lambda)$. $\square$

Lemma 5.4. *Construction 5.2 satisfies the security property.*

Proof. The security follows from the security of G and a hybrid argument. In particular, consider the following hybrids. Fix m_0, m_1.

- H_0 : Run $k \leftarrow \mathsf{Gen}(1^\lambda)$, Run $c \leftarrow \mathsf{Enc}(1^\lambda, k, m_0)$. Output c.
- $\mathsf{H}_{0.a}$ for $a \in \{0,1,\dots,\lambda\}$:
  1. For $i \in \{1,\dots,a\}$, sample $k_i \leftarrow \{0,1\}^{\ell(\lambda)}$.
  2. For $i \in \{a+1,\dots,\lambda\}$, sample $k_i \leftarrow \{0,1\}^\lambda$.
  3. Output $c = m_0 \oplus k_1||\dots||m_0 \oplus k_a||m_0 \oplus G_\lambda(k_{a+1})||\dots||m_0 \oplus G_\lambda(k_\lambda)$.
- H_1 : For $i \in [\lambda]$, sample $c_i \leftarrow \{0,1\}^{\ell(\lambda)}$. Output $c = c_1||\dots||c_\lambda$.
- $\mathsf{H}_{1.b}$ for $b \in \{0,1,\dots,\lambda\}$:
  1. For $i \in \{1,\dots,b\}$, sample $k_i \leftarrow \{0,1\}^\lambda$.
  2. For $i \in \{b+1,\dots,\lambda\}$, sample $k_i \leftarrow \{0,1\}^{\ell(\lambda)}$.
  3. Output $c = m_1 \oplus G_\lambda(k_1)||\dots||m_1 \oplus G_\lambda(k_b)||m_1 \oplus k_{b+1}||\dots||m_1 \oplus k_\lambda$.
- H_2 : Run $k \leftarrow \mathsf{Gen}(1^\lambda)$. Run $c \leftarrow \mathsf{Enc}(1^\lambda, k, m_1)$. Output c.

First, hybrids H_0 and $\mathsf{H}_{0.0}$ are identically distributed. For $a \in [\lambda]$, hybrids $\mathsf{H}_{0.a-1}$ and $\mathsf{H}_{0.a}$ are computational indistinguishabile due to the security of G. Specifically, suppose there exist some $a \in [\lambda]$ and a QPT adversary A that has a non-negligible advantage for distinguishing $\mathsf{H}_{0.a-1}$ from $\mathsf{H}_{0.a}$. Then consider the following reduction R that breaks the strong security of G:

1. Input: $y \in \{0,1\}^{\ell(\lambda)}$ that is either sampled from $G_\lambda(k)$ with a uniform seed k or a unifrom $\ell(\lambda)$-bit string.
2. For $i \in \{1,\ldots,a-1\}$, sample $k_i \leftarrow \{0,1\}^{\ell(\lambda)}$.
3. For $i \in \{a+1,\ldots,\lambda\}$, sample $k_i \leftarrow \{0,1\}^{\lambda}$.
4. Output $c = m_0 \oplus k_1 || \ldots || m_0 \oplus k_{a-1} || m_0 \oplus y || m_0 \oplus G_\lambda(k_{a+1}) || \ldots || m_0 \oplus G_\lambda(k_\lambda)$.

As R runs in polynomial time and has the same distinguishing advantage as that of A, it contradicts the strong security of G. Hybrids $\mathsf{H}_{0.\lambda}$ and H_1 are identically distributed. Hybrids H_1 and $\mathsf{H}_{1.0}$ are identically distributed. Similarly, for $b \in [\lambda]$, hybrids $\mathsf{H}_{1.b-1}$ and $\mathsf{H}_{1.b}$ are computational indistinguishabile due to the security of G. Finally, hybrids $\mathsf{H}_{1.\lambda}$ and H_2 are identically distributed. ☐

5.2 Quantum Commitment with Classical Communication

Next, we construct a (bit) commitment scheme with classical communication from sQPRGs. We follow the definition in [AQY22, AGQY22] closely.

Definition 5.5. *A bit commitment scheme is given by a pair of (uniform) QPT algorithms (C, R), where $C = \{C_\lambda\}_{\lambda\in\mathbb{N}}$ is called the* committer *and $R = \{R_\lambda\}_{\lambda\in\mathbb{N}}$ is called the* receiver*. There are two phases in a commitment scheme: a commit phase and a reveal phase.*

- **Commit phase:** *In the (possibly interactive) commitment phase between C_λ and R_λ, the committer C_λ commits to a bit b. The communication between C_λ and R_λ is classical.*[7] *We denote the execution of the commit phase to be $\sigma_{CR} \leftarrow \mathsf{Commit}\langle C_\lambda(b), R_\lambda\rangle$, where σ_{CR} is the tensor product of C_λ's state and R_λ's state after the commit phase.*
- **Reveal phase:** *In the reveal phase C_λ interacts with R_λ and the output is a trit $\mu \in \{0,1,\perp\}$ indicating the receiver's output bit or a rejection flag. We denote an execution of the reveal phase where the committer and receiver start with the joint state σ_{CR} by $\mu \leftarrow \mathsf{Reveal}\langle C_\lambda(b), R_\lambda, \sigma_{CR}\rangle$.*

We anticipate the commitment scheme to satisfy the following properties:

- **Correctness:** *We say that a commitment scheme (C, R) satisfies correctness if*

$$\Pr\left[b' = b : \substack{\sigma_{CR} \leftarrow \mathsf{Commit}\langle C_\lambda(b), R_\lambda\rangle, \\ b' \leftarrow \mathsf{Reveal}\langle C_\lambda(b), R_\lambda, \sigma_{CR}\rangle}\right] \geq 1 - \varepsilon(\lambda),$$

  where $\varepsilon(\cdot)$ is a negligible function.

[7]Alternately, both the committer and the receiver measure every message they receive in the computational basis.

- **Computational Hiding:** *We say that a commitment scheme (C, R) satisfies computational hiding if for any malicious QPT receiver $\{R^*_\lambda\}_{\lambda\in\mathbb{N}}$, for any QPT distinguisher $\{D_\lambda\}_{\lambda\in\mathbb{N}}$, the following holds:*

$$\left| \Pr_{(\tau,\sigma_{CR^*})\leftarrow \mathsf{Commit}\langle C_\lambda(0),R^*_\lambda\rangle}[D_\lambda(\sigma_{R^*})=1] - \Pr_{(\tau,\sigma_{CR^*})\leftarrow \mathsf{Commit}\langle C_\lambda(1),R^*_\lambda\rangle}[D_\lambda(\sigma_{R^*})=1] \right| \leq \varepsilon(\lambda),$$

where $\varepsilon(\cdot)$ is a negligible function and τ is the transcript in the commitment phase.

- **Statistical Binding:** *We say that a commitment scheme (C, R) satisfies statistical binding if for any malicious computational unbounded committer $\{C^*_\lambda\}_{\lambda\in\mathbb{N}}$, the following holds:*

$$\Pr\left[\mathsf{Reveal}\langle C^*, R_\lambda, \sigma_{C^*R}\rangle = 0 \wedge \mathsf{Reveal}\langle C^*, R_\lambda, \sigma_{C^*R}\rangle = 1 : (\tau,\sigma_{C^*R})\leftarrow \mathsf{Commit}\langle C^*,R_\lambda\rangle\right] \leq \varepsilon(\lambda).$$

where $\varepsilon(\cdot)$ is a negligible function and τ is the transcript in the commitment phase.

Suppose $G_\lambda : \{0,1\}^\lambda \to \{0,1\}^{\ell(\lambda)}$ is a sQPRG with output length $\ell(\lambda) = 3\lambda$ and pseudodeterminism $1 - O(\lambda^{-c})$ for arbitrary $c > 0$. Consider the following construction, which is adapted from Naor's commitment scheme [Nao89]:

Construction 5.6 (Quantum Bit Commitment with Classical Communication)**.**

- **Commit phase:**
  1. *The receiver R samples $r \leftarrow \{0,1\}^{3\lambda}$ and sends it to the committer C.*
  2. *For $i \in [\lambda]$, the committer C samples $k_i \leftarrow \{0,1\}^\lambda$.*
  3. *The committer C on input $b \in \{0,1\}$, outputs*

$$\mathsf{Com} = \begin{cases} G(k_1)||\ldots||G(k_\lambda) & \text{if } b=0 \\ G(k_1)\oplus r||\ldots||G(k_\lambda)\oplus r & \text{if } b=1. \end{cases}$$

- **Reveal phase:**
  1. *The committer C sends the decommitment message $(b, k_1, \ldots, k_\lambda)$ to the receiver R.*
  2. *The receiver R parses Com as $y_1||\ldots||y_\lambda$ where $y_i \in \{0,1\}^{3\lambda}$ for all $i \in [\lambda]$.*
  3. *For $i \in [\lambda]$, the receiver R checks whether $y_i = G(k_i)$ if $b = 0$; checks whether $y_i = G(k_i) \oplus r$ if $b = 1$. Let $N \in \{0,1,\ldots,\lambda\}$ be the number of occurrences where the equality holds*
  4. *If $N \geq 2\lambda/3$, the receiver R outputs b; otherwise outputs $\perp$.*

Lemma 5.7. *Construction 5.6 satisfies the correctness property.*

Proof. The proof is similar to that of Lemma 5.3. In particular, the correctness follows from the pseudodeterminism of G and Lemma 2.5. □

Lemma 5.8. *Construction 5.6 satisfies the computational hiding property.*

Proof. The proof is similar to the proof of Lemma 5.4. Consider the following hybrids for any fixed $r \in \{0,1\}^{3\lambda}$.

- H_0 : For $i \in [\lambda]$, sample $k_i \leftarrow \{0,1\}^\lambda$. Output $\mathsf{Com} = G(k_1)||\ldots||G(k_\lambda)$;
- $\mathsf{H}_{0.a}$ for $a \in \{0,1,\ldots,\lambda\}$:
  1. For $i \in \{1,\ldots,a\}$, sample $k_i \leftarrow \{0,1\}^{3\lambda}$.
  2. For $i \in \{a+1,\ldots,\lambda\}$, sample $k_i \leftarrow \{0,1\}^\lambda$.
  3. Output $\mathsf{Com} = k_1||\ldots||k_a||G(k_{a+1})||\ldots||G(k_\lambda)$.
- H_1 : For $i \in [\lambda]$, sample $k_i \leftarrow \{0,1\}^{3\lambda}$. Output $\mathsf{Com} = k_1||\ldots||k_\lambda$.
- H_2 : For $i \in [\lambda]$, sample $k_i \leftarrow \{0,1\}^{3\lambda}$. Output $\mathsf{Com} = k_1 \oplus r||\ldots||k_\lambda \oplus r$.
- $\mathsf{H}_{2.b}$ for $b \in \{0,1,\ldots,\lambda\}$:
  1. For $i \in \{1,\ldots,b\}$, sample $k_i \leftarrow \{0,1\}^\lambda$.
  2. For $i \in \{b+1,\ldots,\lambda\}$, sample $k_i \leftarrow \{0,1\}^{3\lambda}$.
  3. Output $c = G(k_1) \oplus r||\ldots||G(k_b) \oplus r||k_{b+1} \oplus r||\ldots||k_\lambda \oplus r$.
- H_3 : For $i \in [\lambda]$, sample $k_i \leftarrow \{0,1\}^\lambda$. Output $G(k_1) \oplus r||\ldots||G(k_\lambda) \oplus r$;

Hybrids H_0 and $\mathsf{H}_{0.0}$ are identically distributed. For $a \in [\lambda]$, following the same lines in Lemma 5.4, hybrids $\mathsf{H}_{0.a-1}$ and $\mathsf{H}_{0.a}$ are computational indistinguishabile due to the strong security of G. Hybrids $\mathsf{H}_{0.\lambda}$ and H_1 are identically distributed. Hybrids H_1 and H_2 are identically distributed. Hybrids H_2 and $\mathsf{H}_{2.0}$ are identically distributed. Similarly, for $b \in [\lambda]$, hybrids $\mathsf{H}_{2.b-1}$ and $\mathsf{H}_{2.b}$ are computational indistinguishabile due to the strong security of G. Finally, hybrids $\mathsf{H}_{2.\lambda}$ and H_3 identically distributed. □

Lemma 5.9. *Construction 5.6 satisfies the statistical binding property.*

Proof. To prove the statistical binding property, we introduce the following definition. For every $k \in \{0,1\}^\lambda$, define $F(k) := \mathrm{argmax}_{y\in\{0,1\}^{3\lambda}} \Pr[G(k) = y]$ (if it is not unique, then we pick the lexicographically first one). Let the set of "bad randomness" $\mathsf{Bad} \subseteq \{0,1\}^{3\lambda}$ be

$$\mathsf{Bad} := \left\{ r \in \{0,1\}^{3\lambda} \mid \exists k, k' \in \{0,1\}^\lambda \text{ s.t. } F(k) \oplus F(k') = r \right\}.$$

Then it is easy to see that $\Pr[r \in \mathsf{Bad} : r \leftarrow \{0,1\}^{3\lambda}] \leq 2^\lambda \cdot 2^\lambda / 2^{3\lambda} = 2^{-\lambda}$. Now, the analysis starts to deviate from the proof of the classical case. Classically, if $r \notin \mathsf{Bad}$, then it is impossible for the malicious committer to succeed. However, since now G is *pseudodeterministic*, there is still a chance that $G(k) \oplus G(k') = r$ for some k, k' even if $r \notin \mathsf{Bad}$. Fortunately, according to the definition of the set Bad, the XOR of the *most likely* output of $G(k)$ and $G(k')$ for any k, k' must not equal r. Below, we will show that the probability that $G(k) \oplus G(k') = r$ conditioned on $r \notin \mathsf{Bad}$ is at most $1/2$ for any k, k'.

First, we state a basic fact regarding the inner product of two probability vectors that have distinct mostly likely outcomes.

Claim 5.10. *Let $p, q \in \mathbb{R}^n$ be two probability vectors such that $\mathrm{argmax}_{i\in[n]}\, p_i \neq \mathrm{argmax}_{i\in[n]}\, q_i$ (if it is not unique, then we pick the lexicographically first one). Then $\sum_{i\in[n]} p_i q_i \leq 1/2$.*

Proof. Without loss of generality, we assume that the coordinates of p are sorted in non-increasing order, i.e., $1 \geq p_1 \geq p_2 \geq \cdots \geq p_n \geq 0$. Since $\text{argmax}_{i\in[n]} p_i \neq \text{argmax}_{i\in[n]} q_i$, we have q_1 to not be the maximum coordinate.

We claim that there exists $(q'_1, \ldots, q'_n)$ such that the following holds:

- $\forall i \geq 3$, $q'_i = 0$,
- $\sum_{i\in[n]} p_i q_i \leq \sum_{i\in[n]} p'_i q'_i$.
- $q'_1 \leq q'_2$.

Suppose we instantiate $q'_1 = q_1$ and $q'_2 = \sum_{i\geq 2} q_i$ then the above three bullet points hold.

Now, we have the following:

$$\sum_{i\in[n]} p_i q'_i \quad = \quad p_1 q'_1 + p_2 q'_2.$$

Since $q'_1 \leq q'_2$, $q'_1 + q'_2 = 1$ and $p_1 \geq p_2$, the above expression is maximized when $q'_1 = q'_2 = \frac{1}{2}$. Thus, $\sum_{i\in[n]} p_i q'_i \leq \frac{1}{2}(p_1 + p_2) \leq \frac{1}{2}$. This further implies that $\sum_{i\in[n]} p_i q_i \leq \frac{1}{2}$. □

Claim 5.11. *For every $r \notin \mathsf{Bad}$ and every $k, k' \in \{0,1\}^\lambda$, $\Pr[G(k) \oplus G(k') = r] \leq 1/2$, where the probability is over the randomness of G.*

Proof. Fix k, k' and $r \notin \mathsf{Bad}$. First, recall that $r \notin \mathsf{Bad}$ means $F(k) \oplus F(k') \neq r$. The probability can be written as

$$\Pr[G(k) \oplus G(k') = r] = \sum_{z\in\{0,1\}^{3\lambda}} \Pr[G(k) = z] \Pr[G(k') = z \oplus r].$$

Now, we define the probability vectors $u, u' \in \mathbb{R}^{2^{3\lambda}}$ for the random variables $G(k), G(k')$ respectively. More precisely, the coordinate of u is defined to be $u_y := \Pr[G(k) = y]$; u' is defined similarly. We use the above notation to rewrite the quantity as follows.

$$\sum_{y\in\{0,1\}^{3\lambda}} \Pr[G(k) = y] \Pr[G(k') = y \oplus r] = \sum_{y\in\{0,1\}^{3\lambda}} u_y \cdot u'_{y\oplus r}.$$

Then we set p and q in Claim 5.10 to be the vertors that satisfy $p_y = u_y$ and $q_y = u_{y\oplus r}$ for all $y \in \{0,1\}^{3\lambda}$. Let $y_{\max}$, $y'_{\max}$ be the most likely outcome of $G(k)$, $G(k')$ respectively. Given that $r \notin \mathsf{Bad}$, we have $y_{\max} \oplus y'_{\max} \neq r$. Hence, u and u' satisfy the condition in Claim 5.10. Finally, by Claim 5.10, we can conclude that $\Pr[G(k) \oplus G(k') = r] \leq 1/2$. □

To prove statistical binding, we have

$$\begin{aligned}
&\Pr\left[\mathsf{Reveal}\langle C^*, R_\lambda, \sigma_{C^*R}\rangle = 0 \wedge \mathsf{Reveal}\langle C^*, R_\lambda, \sigma_{C^*R}\rangle = 1 : {\scriptstyle (\tau,\sigma_{C^*R})\leftarrow\mathsf{Commit}\langle C^*,R_\lambda\rangle}\right] \\
= &\mathop{\mathbb{E}}_{r\leftarrow\{0,1\}^{3\lambda}}\left[\max_{k,k'\in\{0,1\}^{\lambda^2}} \Pr\left[G(k_1)||\ldots||G(k_\lambda) = G(k'_1)\oplus r||\ldots||G(k'_\lambda)\oplus r\right]\right] \\
\leq &\Pr_{r\leftarrow\{0,1\}^{3\lambda}}[r \in \mathsf{Bad}] +
\end{aligned}$$

$$\mathop{\mathbb{E}}_{r \leftarrow \{0,1\}^{3\lambda}} \left[\max_{k,k' \in \{0,1\}^{\lambda^2}} \Pr\left[G(k_1)||\ldots||G(k_\lambda) = G(k'_1) \oplus r||\ldots||G(k'_\lambda) \oplus r\right] \mid r \notin \mathsf{Bad} \right].$$

The first term $\Pr[r \in \mathsf{Bad}]$ is at most $2^{-\lambda}$ as we shown. Let $\xi(n,p)$ be the probability that there are at least $2n/3$ heads when independently tossing a coin n times, where the coin satisfies that $\Pr[\mathsf{Head}] = p$. Then, the second term can be written as

$$\begin{aligned}
&\mathop{\mathbb{E}}_{r \leftarrow \{0,1\}^{3\lambda}} \left[\max_{k,k' \in \{0,1\}^{\lambda^2}} \Pr\left[G(k_1)||\ldots||G(k_\lambda) = G(k'_1) \oplus r||\ldots||G(k'_\lambda) \oplus r\right] \mid r \notin \mathsf{Bad} \right] \\
&= \mathop{\mathbb{E}}_{r \leftarrow \{0,1\}^{3\lambda}} \left[\xi\left(\lambda, \max_{k,k' \in \{0,1\}^{\lambda}} \Pr[G(k) \oplus G(k') = r]\right) \mid r \notin \mathsf{Bad} \right] \\
&\leq \xi\left(\lambda, \frac{1}{2}\right) = 2^{-\Omega(\lambda)},
\end{aligned}$$

where the inequality follows from Claim 5.11; the last equality follows from Lemma 2.5. □

5.3 Non-Adaptive CPA-Secure Quantum Private-Key Encryption with Classical Ciphertexts

Finally, we construct a non-adaptive CPA-Secure private-key encryption with classical ciphertexts from selectively secure QPRFs.

Definition 5.12 (Non-Adaptive CPA-Secure Quantum Private-key Encryption)**.** *We say that a tuple of QPT algorithms* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is a* non-adaptive CPA-secure quantum private-key encryption scheme *if the following holds:*

- **Correctness:** *There exists a negligible function $\varepsilon(\cdot)$ such that for every $\lambda \in \mathbb{N}$ and every message m,*
$$\Pr_{k \leftarrow \{0,1\}^{\lambda}} \left[\mathsf{Dec}(1^\lambda, k, \mathsf{Enc}(1^\lambda, k, m)) = m\right] \geq 1 - \varepsilon(\lambda).$$
- **Non-adaptive CPA security:** *For every polynomial $q(\cdot)$, any (non-uniform) QPT adversary A, there exists a negligible function $\varepsilon(\cdot)$ such that for all $\lambda \in \mathbb{N}$, the adversary A has at most $\varepsilon(\lambda)$ advantage in the following experiment:*

  1. *The challenger generates a key k by running $\mathsf{Gen}(1^\lambda)$ and a uniform bit $b \in \{0,1\}$.*
  2. *The adversary A is given input 1^λ.*
  3. *The adversary A chooses messages $(m_1^0, m_1^1) \ldots, (m_q^0, m_q^1)$ and sends them to the challenger.*
  4. *The challenger sends $\mathsf{Enc}(k, m_1^b), \ldots, \mathsf{Enc}(k, m_q^b)$ to the adversary A.*
  5. *The adversary A outputs a bit $b' \in \{0,1\}$.*
  6. *The challenger output 1 if $b' = b$, and 0 otherwise.*

Suppose $F : \{0,1\}^\lambda \times \{0,1\}^{m(\lambda)} \to \{0,1\}^{\ell(\lambda)}$ is a selectively secure QPRF with $m(\lambda) = \omega(\log \lambda)$ and pseudodeterminism $1 - O(\lambda^{-c})$ for arbitrary $c > 0$. In the classical case, selectively secure pseudorandom functions imply the existence of non-adaptive CPA-secure private-key encryption schemes. With a slight modification, we have the following construction.

Construction 5.13 (Non-Adaptive CPA-Secure Quantum Private-Key Encryption Scheme)**.**

1. $\mathsf{Gen}(1^\lambda)$: *on input* 1^λ, *output* $k \leftarrow \{0,1\}^{\lambda^2}$.
2. $\mathsf{Enc}(1^\lambda, k, m)$: *on input a key* $k \in \{0,1\}^{\lambda^2}$ *and a message* $m \in \{0,1\}^{\ell(\lambda)}$,
   - *Parse* k *as* $k_1||\ldots||k_\lambda$ *such that* $k_i \in \{0,1\}^\lambda$ *for every* $i \in [\lambda]$.
   - *Choose a uniform string* $r \leftarrow \{0,1\}^{m(\lambda)}$.
   - *For* $i \in [\lambda]$, *compute* $F(k_i, r)$.
   - *Output* $c = (r, m \oplus F(k_1, r), \ldots, m \oplus F(k_\lambda, r))$.
3. $\mathsf{Dec}(1^\lambda, k, c)$: *on input a key* $k \in \{0,1\}^{\lambda^2}$ *and a ciphertext* $c = (r, c_1, \ldots, c_\lambda) \in \{0,1\}^{m(\lambda)+\lambda\ell(\lambda)}$,
   - *Parse* k *as* $k_1||\ldots||k_\lambda$ *such that* $k_i \in \{0,1\}^\lambda$ *for every* $i \in [\lambda]$.
   - *For* $i \in [\lambda]$, *compute* $m_i := c_i \oplus F(k_i, r) \in \{0,1\}^{\ell(\lambda)}$.
   - *Output* $m := \mathsf{MAJ}(m_1, \ldots, m_\lambda)$.

Lemma 5.14. *Construction 5.13 satisfies the correctness property.*

Proof. The proof is similar to that of Lemma 5.3. The correctness follows from the pseudodeterminism of F and Lemma 2.5. □

The following lemma follows the proof of Lemma 7.3 in [AQY22] closely.

Lemma 5.15. *Construction 5.13 satisfies non-adaptive CPA security.*

Proof. We finish the proof with a hybrid argument. Consider the following hybrids.

- H_1 : The adversary receives $\left(\mathsf{Enc}(k, m_1^b), \ldots, \mathsf{Enc}(k, m_q^b)\right)$, which by definition is

  $$\Big((r_1, m_1^b \oplus F(k_1, r_1), \ldots, m_1^b \oplus F(k_\lambda, r_1)), \ldots, (r_q, m_q^b \oplus F(k_1, r_q), \ldots, m_q^b \oplus F(k_\lambda, r_q))\Big)$$

  where $r_1, \ldots, r_q$ are independently and uniformly chosen.

- $\mathsf{H}_{1.i}$ for $i \in \{0, 1, \ldots, \lambda\}$: The adversary receives

  $$\Bigg(\Big(r_1, m_1^b \oplus R_1(r_1), \ldots, m_1^b \oplus R_i(r_1), m_1^b \oplus F(k_{i+1}, r_1)), \ldots, m_1^b \oplus F(k_\lambda, r_1)\Big),$$
  $$\ldots, \Big(r_q, m_q^b \oplus R_1(r_q), \ldots, m_q^b \oplus R_i(r_q), m_q^b \oplus F(k_{i+1}, r_q)), \ldots, m_q^b \oplus F(k_\lambda, r_q)\Big)\Bigg)$$

  where $r_1, \ldots, r_q$ are independently, uniformly chosen and $R_1(\cdot), \ldots, R_i(\cdot)$ are independent random functions.

- H_2 : The adversary receives

  $$\Bigg(\Big(r_1, m_1^b \oplus R_1(r_1), \ldots, m_1^b \oplus R_\lambda(r_1)\Big), \ldots, \Big(r_q, m_q^b \oplus R_1(r_q), \ldots, m_q^b \oplus R_\lambda(r_q)\Big)\Bigg)$$

  where $r_1, \ldots, r_q$ are independently, uniformly chosen and $R_1(\cdot), \ldots, R_\lambda(\cdot)$ are independent random functions.

- H_3 : Instead of sampling $r_1, \ldots, r_q$ independently, they are sampled uniformly at random conditioned on them all being distinct. The adversary receives

$$\left(\left(r_1, m_1^b \oplus R_1(r_1), \ldots, m_1^b \oplus R_\lambda(r_1)\right), \ldots, \left(r_q, m_q^b \oplus R_1(r_q), \ldots, m_q^b \oplus R_\lambda(r_q)\right) \right)$$

where $R_1(\cdot), \ldots, R_\lambda(\cdot)$ are independent random functions.

Hybrids H_1 and $\mathsf{H}_{1.0}$ are identically distributed. For $i \in \{0, 1, \ldots, \lambda - 1\}$, hybrids $\mathsf{H}_{1.i}$ and $\mathsf{H}_{1.i+1}$ are computational indistinguishabile from the selective security of F. In particular, suppose there exist some $i \in [\lambda]$ and a QPT adversary A such that the difference between A's winning probabilities in $\mathsf{H}_{1.i}$ and $\mathsf{H}_{1.i+1}$ is non-negligible. Consider the following reduction R that breaks the selective security of the underlying QPRF F.

1. Receive $(m_1^0, m_1^1) \ldots, (m_q^0, m_q^1)$ from A.
2. Sample $x_1, x_2, \ldots, x_q \leftarrow \{0, 1\}^{m(\lambda)}$.
3. Query the oracle on $x_1, x_2, \ldots, x_q$ and obtain $y_1, y_2, \ldots, y_q$, where y_i's are either sampled from $F(k, x_i)$ with a uniform key k or i.i.d. uniform $\ell(\lambda)$-bit strings.
4. For $j \in \{1, 2, \ldots, q\}$, do the following
   (a) Sample independent random functions $R_1(\cdot), \ldots, R_i(\cdot)$ and compute $R_1(x_j)$, $R_2(x_j)$, $\ldots, R_i(x_j)$.[8]
   (b) Sample $k_{i+2}, \ldots, k_\lambda \leftarrow \{0, 1\}^\lambda$ and compute $F(k_{i+2}, x_j), \ldots, F(k_\lambda, x_j)$.
5. Sample a uniform bit $b \in \{0, 1\}$.
6. Send

$$\begin{gathered}(x_1, m_1^b \oplus R_1(x_1), \ldots, m_1^b \oplus R_i(x_1), m_1^b \oplus y_1, m_1^b \oplus F(k_{i+2}, x_1), \ldots, F(k_\lambda, x_1)), \\ (x_2, m_2^b \oplus R_1(x_2), \ldots, m_2^b \oplus R_i(x_2), m_2^b \oplus y_2, m_2^b \oplus F(k_{i+2}, x_2), \ldots, F(k_\lambda, x_2)), \\ \vdots \\ (x_q, m_q^b \oplus R_1(x_q), \ldots, m_q^b \oplus R_i(x_q), m_q^b \oplus y_q, m_q^b \oplus F(k_{i+2}, x_q), \ldots, F(k_\lambda, x_q))\end{gathered}$$

   to A and get the output b'.
7. If $b = b'$, then output 1. Otherwise, output 0.

If y_i's are the output of the QPRF F, then the reduction R perfectly simulates A's view in H_i. On the other hand, if y_i's are i.i.d. uniform bitstrings, then the reduction R perfectly simulates A's view in H_{i+1}. Hence, the distinguishing advantage of R is equivalent to the difference between A's winning probabilities in $\mathsf{H}_{1.i}$ and $\mathsf{H}_{1.i+1}$. However, this contradicts the selective security of F. Hybrids $\mathsf{H}_{1.\lambda}$ and H_2 are identically distributed. The statistical distance between hybrids H_2 and H_3 is $O(q^2/2^m) = \mathsf{negl}(\lambda)$ from a similar calculation of Lemma 7.3 in [AQY22]. Finally, the advantage of A in H_3 is 0 since all the messages are independently one-time padded.

□

[8] The reduction R uses lazy evaluation to simulate each random function instead of sampling the whole function table.

References

[AC02] Mark Adcock and Richard Cleve. "A quantum Goldreich-Levin theorem with cryptographic applications". In: *STACS 2002: 19th Annual Symposium on Theoretical Aspects of Computer Science Antibes-Juan les Pins, France, March 14–16, 2002 Proceedings*. Springer. 2002, pp. 323–334 (cit. on p. 44).

[AE07] Andris Ambainis and Joseph Emerson. "Quantum t-designs: t-wise Independence in the Quantum World". In: *Twenty-Second Annual IEEE Conference on Computational Complexity (CCC'07)*. 2007, pp. 129–140. DOI: 10.1109/CCC.2007.26 (cit. on p. 26).

[AGQY22] Prabhanjan Ananth, Aditya Gulati, Luowen Qian, and Henry Yuen. "Pseudorandom (Function-Like) Quantum State Generators: New Definitions and Applications". In: *Theory of Cryptography Conference*. Springer. 2022, pp. 237–265 (cit. on pp. 3, 5, 14, 31).

[AIK06] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. "Computationally private randomizing polynomials and their applications". In: *computational complexity* 15.2 (2006), pp. 115–162 (cit. on p. 6).

[AQY22] Prabhanjan Ananth, Luowen Qian, and Henry Yuen. "Cryptography from Pseudorandom Quantum States." In: *CRYPTO*. 2022 (cit. on pp. 3, 4, 6, 11, 31, 36, 37).

[BBSS23] Amit Behera, Zvika Brakerski, Or Sattath, and Omri Shmueli. *Pseudorandomness with Proof of Destruction and Applications*. Cryptology ePrint Archive, Paper 2023/543. https://eprint.iacr.org/2023/543. 2023. URL: https://eprint.iacr.org/2023/543 (cit. on p. 5).

[BCKM21] James Bartusek, Andrea Coladangelo, Dakshita Khurana, and Fermi Ma. "One-Way Functions Imply Secure Computation in a Quantum World". In: *Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16-20, 2021, Proceedings, Part I*. Vol. 12825. 2021, pp. 467–496. DOI: 10.1007/978-3-030-84242-0_17 (cit. on p. 5).

[BCQ22] Zvika Brakerski, Ran Canetti, and Luowen Qian. "On the computational hardness needed for quantum cryptography". In: *arXiv preprint arXiv:2209.04101* (2022) (cit. on p. 3).

[BY22] Zvika Brakerski and Henry Yuen. "Quantum garbled circuits". In: *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*. 2022, pp. 804–817 (cit. on p. 6).

[CHS05] Ran Canetti, Shai Halevi, and Michael Steiner. "Hardness amplification of weakly verifiable puzzles". In: *Theory of Cryptography: Second Theory of Cryptography Conference, TCC 2005, Cambridge, MA, USA, February 10-12, 2005. Proceedings 2*. 2005, pp. 17–33 (cit. on pp. 22, 41).

[DCEL09] Christoph Dankert, Richard Cleve, Joseph Emerson, and Etera Livine. "Exact and approximate unitary 2-designs and their application to fidelity estimation". In: *Phys. Rev. A* 80 (1 2009), p. 012304. DOI: 10.1103/PhysRevA.80.012304. URL: https://link.aps.org/doi/10.1103/PhysRevA.80.012304 (cit. on p. 26).

[DF87] Persi Diaconis and David Freedman. "A dozen de Finetti-style results in search of a theory". en. In: *Annales de l'I.H.P. Probabilités et statistiques* 23.S2 (1987), pp. 397–423. URL: http://www.numdam.org/item/AIHPB_1987__23_S2_397_0/ (cit. on p. 13).

[DIJK09] Yevgeniy Dodis, Russell Impagliazzo, Ragesh Jaiswal, and Valentine Kabanets. "Security amplification for interactive cryptographic primitives". In: *Theory of Cryptography: 6th Theory of Cryptography Conference, TCC 2009, San Francisco, CA, USA, March 15-17, 2009. Proceedings 6*. Springer. 2009, pp. 128–145 (cit. on pp. 9, 22, 40, 41, 43, 44).

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. "How to construct random functions". In: *Journal of the ACM (JACM)* 33.4 (1986), pp. 792–807 (cit. on pp. 3, 7).

[GLSV21] Alex B. Grilo, Huijia Lin, Fang Song, and Vinod Vaikuntanathan. "Oblivious Transfer Is in MiniQCrypt". In: *Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part II*. Vol. 12697. 2021, pp. 531–561. DOI: 10.1007/978-3-030-77886-6_18 (cit. on p. 5).

[Gol90] Oded Goldreich. "A note on computational indistinguishability". In: *Information Processing Letters* 34.6 (1990), pp. 277–281. DOI: 10.1016/0020-0190(90)90010-U (cit. on p. 3).

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A Levin, and Michael Luby. "A pseudorandom generator from any one-way function". In: *SIAM Journal on Computing* 28.4 (1999), pp. 1364–1396 (cit. on p. 3).

[IJK09] Russell Impagliazzo, Ragesh Jaiswal, and Valentine Kabanets. "Chernoff-type direct product theorems". In: *Journal of Cryptology* 22.1 (2009), pp. 75–92 (cit. on p. 40).

[Imp95] Russell Impagliazzo. "A Personal View of Average-Case Complexity". In: *Proceedings of the Tenth Annual Structure in Complexity Theory Conference, Minneapolis, Minnesota, USA, June 19-22, 1995*. 1995, pp. 134–147. DOI: 10.1109/SCT.1995.514853 (cit. on p. 3).

[IW97] Russell Impagliazzo and Avi Wigderson. "P= BPP if E requires exponential circuits: Derandomizing the XOR lemma". In: *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*. 1997, pp. 220–229 (cit. on p. 3).

[JLS18] Zhengfeng Ji, Yi-Kai Liu, and Fang Song. "Pseudorandom Quantum States". In: *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part III*. Vol. 10993. 2018, pp. 126–152. DOI: 10.1007/978-3-319-96878-0_5 (cit. on pp. 3, 4, 11).

[Kre21] William Kretschmer. "Quantum Pseudorandomness and Classical Complexity". In: *16th Conference on the Theory of Quantum Computation, Communication and Cryptography, TQC 2021, July 5-8, 2021, Virtual Conference*. Vol. 197. 2021, 2:1–2:20. DOI: 10.4230/LIPIcs.TQC.2021.2 (cit. on pp. 3, 5).

[LP20] Yanyi Liu and Rafael Pass. "On one-way functions and Kolmogorov complexity". In: *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2020, pp. 1243–1254 (cit. on p. 3).

[Mec19] Elizabeth S Meckes. *The random matrix theory of the classical compact groups*. Vol. 218. Cambridge University Press, 2019 (cit. on pp. 7, 13).

[MT09] Ueli Maurer and Stefano Tessaro. "Computational Indistinguishability Amplification: Tight Product Theorems for System Composition". In: *Advances in Cryptology — CRYPTO 2009*. Vol. 5677. 2009, pp. 350–368 (cit. on pp. 9, 22).

[MT10] Ueli Maurer and Stefano Tessaro. "A Hardcore Lemma for Computational Indistinguishability: Security Amplification for Arbitrarily Weak PRGs with Optimal Stretch". In: *Theory of Cryptography — TCC 2010*. Vol. 5978. 2010, pp. 237–254 (cit. on pp. 9, 22).

[Mul59] Mervin E. Muller. "A Note on a Method for Generating Points Uniformly on N-Dimensional Spheres". In: *Commun. ACM* 2.4 (1959), pp. 19–20. DOI: 10.1145/377939.377946. URL: https://doi.org/10.1145/377939.377946 (cit. on p. 13).

[MY21] Tomoyuki Morimae and Takashi Yamakawa. *Quantum commitments and signatures without one-way functions*. 2021. DOI: 10.48550/ARXIV.2112.06369 (cit. on p. 3).

[MY22] Tomoyuki Morimae and Takashi Yamakawa. *One-Wayness in Quantum Cryptography*. Cryptology ePrint Archive, Paper 2022/1336. https://eprint.iacr.org/2022/1336. 2022. URL: https://eprint.iacr.org/2022/1336 (cit. on pp. 3, 41).

[Nao89] Moni Naor. "Bit Commitment Using Pseudo-Randomness". In: *Advances in Cryptology - CRYPTO '89, 9th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 1989, Proceedings*. Vol. 435. 1989, pp. 128–136. DOI: 10.1007/0-387-34805-0_13 (cit. on p. 32).

[Nao91] Moni Naor. "Bit commitment using pseudorandomness". In: *Journal of Cryptology* 4.2 (1991), pp. 151–158. DOI: 10.1007/BF00196774 (cit. on p. 10).

[NC10] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010. DOI: 10.1017/CBO9780511976667 (cit. on p. 10).

[NW94] Noam Nisan and Avi Wigderson. "Hardness vs randomness". In: *Journal of computer and System Sciences* 49.2 (1994), pp. 149–167 (cit. on p. 3).

[RR94] Alexander A Razborov and Steven Rudich. "Natural proofs". In: *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*. 1994, pp. 204–213 (cit. on p. 3).

[RS19] Roy Radian and Or Sattath. "Semi-quantum money". In: *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*. 2019, pp. 132–146 (cit. on p. 41).

[SM62] S Kh Sirazhdinov and M Mamatov. "On convergence in the mean for densities". In: *Theory of Probability & Its Applications* 7.4 (1962), pp. 424–428 (cit. on pp. 8, 13).

[Yao82] Andrew C. Yao. "Theory and application of trapdoor functions". In: *23rd Annual Symposium on Foundations of Computer Science (sfcs 1982)*. 1982, pp. 80–91. DOI: 10.1109/SFCS.1982.45 (cit. on p. 41).

A Full Proof of Theorem 4.5

In this section, we aim to complete the proof of Theorem 4.5. The proof is essentially the same as that of [DIJK09]. In [DIJK09], Theorem 4.4 is proven by a direct product theorem in [IJK09]. This is partly because they consider more general settings, e.g., interactive primitives and direct products with thresholds. For our purpose, we can use the hardness amplification of weakly verifiable puzzles by Canetti, Halevi, and Steiner [CHS05]. Specifically, Lemma 14 in [DIJK09] can be proven by techniques in [CHS05]. We note that Radian and Sattath [RS19] observed the result in [CHS05] can be extended to the post-quantum setting without modifying the proof; Morimae and Yamakawa [MY22] further generalized the result in [CHS05] to the setting where the puzzle and solution are quantum under certain conditions.

We first recall the definition of weakly verifiable puzzles in [CHS05] and extend it to our setting where the puzzle generator, verifier and solver are QPTs while the puzzle and solution remain classical.

Definition A.1 (Weakly verifiable puzzles). *A weakly verifiable puzzle is a pair of QPT algorithms* $\Pi = (\mathsf{Gen}, \mathsf{Ver})$ *that satisfies the following:*

1. $\mathsf{Gen}(1^\lambda) \to (P, C)$ *: on input* 1^λ*, outputs a classical puzzle* P *along with a classical (secret) bitstring* C *for verification.*

2. $\mathsf{Ver}(P, S, C) \to \top/\bot$ *: on input a classical puzzle* P*, a classical solution* S *and a classical bitstring* C*, outputs the symbol* $\top$ *if the verification is successful or* $\bot$ *if it fails.*

Theorem A.2 ([CHS05, Theorem 1],[RS19, Theorem 20]). *Let* $\varepsilon : \mathbb{N} \to [0, 1]$ *be an efficiently computable function, let* $s : \mathbb{N} \to \mathbb{N}$ *be efficiently computable and polynomially bounded, and let* $\Pi = (\mathsf{Gen}, \mathsf{Ver})$ *be a weakly verifiable puzzle system. If* Π *is* $(1 - \varepsilon)$*-hard against QPT adversaries, then* Π^s*, the* s*-fold repetition of* Π*, is* $(1 - \varepsilon^s)$*-hard against QPT adversaries.*

Next, we note that Yao's next-bit unpredictability lemma [Yao82] can be extended to QPRGs. The proof is almost the same as the classical case except that QPRGs are pseudodeterministic – which means that the first i bits of the QPRG's output and its next bit need to be well-defined. To address this issue, we sample $G(k)$ once and define the first i bits and the next bit accordingly. For completeness, we present the proof below.

Lemma A.3 (Next-bit unpredictability lemma for QPRGs). *If a QPT algorithm* $G : \{0,1\}^\lambda \to \{0,1\}^{\ell(\lambda)}$ *is* $(1 - \delta)$*-pseudorandom, then for any QPT algorithm* A *and any* $i \in \{0, 1, \ldots, \ell(\lambda) - 1\}$*,*

$$\Pr\left[b' = y_{i+1} : \begin{smallmatrix} k \leftarrow \{0,1\}^\lambda, \\ y \leftarrow G(k), \\ b' \leftarrow A(y_{[1:i]}) \end{smallmatrix}\right] \leq \frac{1}{2} + \delta,$$

where $y_{[1:i]}$ *denotes the first* i *bits of* y*, and* y_{i+1} *denotes the* $(i + 1)$*-th bit of* y*.*

Let $G : \{0,1\}^\lambda \to \{0,1\}^{\ell(\lambda)}$ *be a QPT algorithm such that for any QPT algorithm* A *and any* $i \in \{0, 1, \ldots, \ell(\lambda) - 1\}$*,*

$$\Pr\left[b' = y_{i+1} : \begin{smallmatrix} k \leftarrow \{0,1\}^\lambda, \\ y \leftarrow G(k), \\ b' \leftarrow A(y_{[1:i]}) \end{smallmatrix}\right] \leq \frac{1}{2} + \delta.$$

Then G *is* $(1 - \delta\ell)$*-pseudorandom.*

Proof. (*Pseudorandomness implies next-bit unpredictability.*) For the sake of contradiction, suppose there exists some $i \in \{0, 1, \dots, \ell(\lambda) - 1\}$ and a QPT algorithm A such that

$$\Pr\left[b' = y_{i+1} : \substack{k \leftarrow \{0,1\}^\lambda, \\ y \leftarrow G(k), \\ b' \leftarrow A\left(y_{[1:i]}\right)}\right] > \frac{1}{2} + \delta.$$

We construct the following reduction R the breaks the security of G: on input $y \in \{0, 1\}^{\ell(\lambda)}$, do the following,

1. Run $A(y_{[1:n]})$.
2. Receive a bit b' from A.
3. Output 1 if $b' = y_{i+1}$, and 0 otherwise.

When the input string y is sampled from $G(k)$ with a random seed k, then the reduction perfectly simulates the view of A. On the other hand, if y is a uniform string, then the probability of $b' = y_{i+1}$ is $1/2$ since the information of y_{i+1} is never revealed. Hence, we have

$$\Pr\left[R(y) = 1 : \substack{k \leftarrow \{0,1\}^\lambda, \\ y \leftarrow G(k)}\right] - \Pr[R(y) = 1 : y \leftarrow \{0, 1\}^{\ell(\lambda)}] \geq \left(\frac{1}{2} + \delta\right) - \frac{1}{2} = \delta.$$

However, this contradicts the assumption that G is $(1 - \delta)$-pseudorandom.

(*Next-bit unpredictability implies pseudorandomness.*) For the sake of contradiction, suppose there exists a QPT algorithm A that breaks the pseudorandomness of G, i.e.,

$$\left|\Pr[A(y) = 1 : k \leftarrow \{0, 1\}^\lambda, y \leftarrow G(k)] - \Pr[A(y) = 1 : y \leftarrow \{0, 1\}^{\ell(\lambda)}]\right| > \delta\ell.$$

Consider the following hybrids:

- $\mathsf{H}_0 : k \leftarrow \{0, 1\}^\lambda$; $y \leftarrow G(k)$; Output y.
- H_i for $i \in \{1, \dots, \ell(\lambda) - 1\} : k \leftarrow \{0, 1\}^\lambda$; $y \leftarrow G(k)$; $u \leftarrow \{0, 1\}^{\ell(\lambda)}$; Output $y_{[1:\ell-i+1]}||u_{[\ell-i:\ell]}$.
- $\mathsf{H}_\ell : u \leftarrow \{0, 1\}^{\ell(\lambda)}$; Output u.

Hence, by hybrid argument, there exists some $i \in [\ell]$ such that

$$\left|\Pr_{\mathsf{H}_{i-1}}[A(y) = 1] - \Pr_{\mathsf{H}_i}[A(y) = 1]\right| > \frac{\delta\ell}{\ell} = \delta.$$

Without loss of generality, we can assume that $\Pr_{\mathsf{H}_{i-1}}[A(y) = 1] - \Pr_{\mathsf{H}_i}[A(y) = 1] > \delta$. Now, we construct a reduction R that takes as input the first $i - 1$ bits of $G(k)$ and runs A to predict the i-th bit of $G(k)$. Consider the following reduction R: on input $y = y_1|| \dots ||y_{i-1}$, do the following,

1. Sample uniform bits $u_i, \dots, u_\ell \in \{0, 1\}$.
2. Run $A(y_1|| \dots ||y_{i-1}, u_i, \dots, u_\ell)$.
3. Receive a bit b' from A.

4. Output u_i if $b' = 1$, and $u_i \oplus 1$ otherwise.

Note that the first two steps perfectly simulate H_i for A, thus $\Pr[b' = 1] = \Pr_{\mathsf{H}_i}[A(y) = 1]$. Moreover, conditioned on $u_i = y_i$, the reduction perfectly simulates H_{i-1} for A, thus $\Pr[b' = 1 \mid u_i = y_i] = \Pr_{\mathsf{H}_{i-1}}[A(y) = 1]$. Now, since u_i is chosen uniformly at random, we have $\Pr[b = 1] = \frac{1}{2}\Pr[b = 1 \mid u_i = y_i] + \frac{1}{2}\Pr[b = 1 \mid u_i \neq y_i]$, or equivalently $\Pr[b = 1 \mid u_i \neq y_i] = 2\Pr[b' = 1] - \Pr[b = 1 \mid u_i = y_i] = 2\Pr_{\mathsf{H}_i}[A(y) = 1] - \Pr_{\mathsf{H}_{i-1}}[A(y) = 1]$.
Then the success probability of R is given by

$$
\begin{aligned}
&\Pr[u_i = y_i \wedge b' = 1] + \Pr[u_i \neq y_i \wedge b' = 0] \\
&= \Pr[u_i = y_i]\Pr[b' = 1 \mid u_i = y_i] + \Pr[u_i \neq y_i]\Pr[b' = 0 \mid u_i \neq y_i] \\
&= \Pr[u_i = y_i]\Pr[b' = 1 \mid u_i = y_i] + \Pr[u_i \neq y_i](1 - \Pr[b' = 1 \mid u_i \neq y_i]) \\
&= \frac{1}{2}\Pr_{\mathsf{H}_{i-1}}[A(y) = 1] + \frac{1}{2}\left(1 - 2\Pr_{\mathsf{H}_i}[A(y) = 1] + \Pr_{\mathsf{H}_{i-1}}[A(y) = 1]\right) \\
&= \frac{1}{2} + \left(\Pr_{\mathsf{H}_{i-1}}[A(y) = 1] - \Pr_{\mathsf{H}_i}[A(y) = 1]\right) \\
&\geq \frac{1}{2} + \delta.
\end{aligned}
$$

However, this contradicts the next-bit unpredictability of G. $\square$

Building upon the argument in [DIJK09], we can interpret $y_{[1:i]}$ as the *puzzle* and the next bit y_{i+1} as the *solution*. Then we define the $(2s)$-fold puzzle as $y^1_{[1:i]}||\ldots||y^{2s}_{[1:i]}$, where $k_1, \ldots, k_{2s}$ are independently, uniformly chosen and $y^1 \leftarrow G(k_1), \ldots, y^{2s} \leftarrow G(k_{2s})$. The solution will be $y^1_{i+1}||\ldots||y^{2s}_{i+1}$. Here, we extend Lemma 14 in [DIJK09] that states the hardness amplification for the above puzzle to QPRGs.

Lemma A.4 (Bit-wise direct product lemma for QPRGs)**.** *Let $G : \{0,1\}^\lambda \to \{0,1\}^{\ell(\lambda)}$ be a QPRG such that for any QPT algorithm A, we have for all $i \in \{0, 1, \ldots, \ell(\lambda) - 1\}$,*

$$
\Pr\left[b' = y_{i+1} : \begin{array}{c} k \leftarrow \{0,1\}^\lambda, \\ y \leftarrow G(k), \\ b' \leftarrow A\left(y_{[1:i]}\right) \end{array}\right] \leq \frac{1}{2} + \delta,
$$

where $\delta(\lambda) \leq 0.49 + o(1)$. Then for any QPT algorithm A' we have for all $i \in \{0, 1, \ldots, \ell(\lambda) - 1\}$,

$$
\Pr\left[\bigwedge_{j=1}^{2s}\left(b'_j = y^j_{i+1}\right) : \begin{array}{c} k_1,\ldots,k_{2s} \leftarrow \{0,1\}^\lambda, \\ y^1 \leftarrow G(k_1),\ldots,y^{2s} \leftarrow G(k_{2s}), \\ (b'_1,\ldots,b'_{2s}) \leftarrow A'\left(y^1_{[1:i]},\ldots,y^{2s}_{[1:i]}\right) \end{array}\right] \leq \varepsilon,
$$

where $\varepsilon = e^{-\Omega(s)}$.

Proof. Following the arguments in [DIJK09], we model the above problem as a weakly verifiable puzzle. Specifically, the puzzle generator Gen is defined as follows: sample $k_1, \ldots, k_{2s} \leftarrow \{0,1\}^\lambda$ and then run $y^1 \leftarrow G(k_1), \ldots, y^{2s} \leftarrow G(k_{2s})$. The puzzle P is $\left\{y^1_{[1:i]}, \ldots, y^{2s}_{[1:i]}\right\}$. The classical (secret) bitstring C is $\{y^1_{i+1}, \ldots, y^{2s}_{i+1}\}$. The solution S is of the form $\{b'_1, \ldots, b'_{2s}\}$. The puzzle verifier Ver takes as input (C, P, S) and outputs $\top$ if and only if it satisfies $\bigwedge_{j=1}^{2s}\left(b'_j = y^j_{i+1}\right)$. Finally, we apply Theorem A.2 on the above puzzle and obtain $\varepsilon = O\left(\left(\frac{1}{2} + \delta\right)^{2s}\right) = e^{-\Omega(s)}$. $\square$

Moreover, Lemma 15 in [DIJK09] also can be generalized for QPRGs.

Lemma A.5 (Direct product theorem implies xor lemma for QPRGs)**.** *Let $G : \{0,1\}^\lambda \to \{0,1\}^{\ell(\lambda)}$ be a QPRG such that for any QPT A, we have for all $i \in \{0,1,\dots,\ell(\lambda)-1\}$,*

$$\Pr\left[b' = y_{i+1} : \substack{k \leftarrow \{0,1\}^\lambda, \\ y \leftarrow G(k), \\ b' \leftarrow A(y_{[1:i]})}\right] \le \frac{1}{2} + \delta,$$

where $\delta(\lambda) \le 0.49 + o(1)$. Then for any QPT algorithm A' we have for all $i \in \{0,1,\dots,\ell(\lambda)-1\}$,

$$\Pr\left[b' = \bigoplus_{j=1}^{s} y_{i+1}^j : \substack{k_1,\dots,k_s \leftarrow \{0,1\}^\lambda, \\ y^1 \leftarrow G(k_1),\dots,y^s \leftarrow G(k_s), \\ b' \leftarrow A'\left(y_{[1:i]}^1 \oplus \dots \oplus y_{[1:i]}^s\right)}\right] \le \frac{1}{2} + \varepsilon,$$

where $\varepsilon = e^{-\Omega(s)}$.

Proof. The proof is essentially the same as that of [DIJK09]. For completeness, we sketch the proof below. Suppose there exists a QPT adversary A' such that

$$\Pr\left[b' = \bigoplus_{j=1}^{s} y_{i+1}^j : \substack{k_1,\dots,k_s \leftarrow \{0,1\}^\lambda, \\ y^1 \leftarrow G(k_1),\dots,y^s \leftarrow G(k_s), \\ b' \leftarrow A'\left(y_{[1:i]}^1 \oplus \dots \oplus y_{[1:i]}^s\right)}\right] > \frac{1}{2} + \varepsilon.$$

holds for some $i \in \{0,1,\dots,\ell(\lambda)-1\}$ and $\varepsilon \notin e^{-\Omega(s)}$. Then we will construct a reduction A'' that on input a random string $r \in \{0,1\}^{2s}$ and $y_{[1:i]}^1,\dots,y_{[1:i]}^{2s} \in \{0,1\}^i$, output the inner product of r and $y_{i+1}^1||\dots||y_{i+1}^{2s}$, where $y^1 \leftarrow G(k_1),\dots,y^{2s} \leftarrow G(k_{2s})$ and $k_1,\dots,k_{2s}$ are independent, uniform seeds. The description of A'' is the following: on input $r \in \{0,1\}^{2s}$ and $y_{[1:i]}^1,\dots,y_{[1:i]}^{2s} \in \{0,1\}^i$, do the following,

1. If the number of 1's in r is not s, then output a random bit b.
2. Otherwise, let $z_1,\dots,z_s \in [2s]$ be the indices such that $r_{z_j} = 1$ for all $j \in [s]$.
3. Run $A'(y_{[1:i]}^{z_1} \oplus \dots \oplus y_{[1:i]}^{z_s})$.
4. Output whatever A' outputs.

First, note that when r has exactly s 1's, the inner product satisfies $\langle r, y_{i+1}^1||\dots||y_{i+1}^{2s}\rangle = y_{i+1}^{z_1} \oplus \dots \oplus y_{i+1}^{z_s}$. Moreover, the probability that r has exactly s 1's with probability $\Theta(1/\sqrt{s})$. This implies that A'' computes the above inner product with probability at least $1/2+\varepsilon'$, where $\varepsilon' = \Theta(\varepsilon/\sqrt{s})$. By averaging, with probability at least $\varepsilon'/2$ the tuples $(k_1,y^1),\dots,(k_{2s},y^{2s})$ are "good" such that A'' computes the inner product with a randomly chosen r with probability at least $1/2+\varepsilon'/2$. Now, using the Goldreich-Levin Theorem,[9] we can construct A''' which for every good $(k_1,y^1),\dots,(k_{2s},y^{2s})$, computes $y_{i+1}^1||\dots||y_{i+1}^{2s}$ with probability at least $\Theta((\varepsilon')^2/s)$. This implies that A''' computes $y_{i+1}^1||\dots||y_{i+1}^{2s}$ with probability at least $\Omega(\varepsilon^3/s^{5/2})$ which contradicts Lemma A.4. □

[9]The adversary A''' receives, as non-uniform advice, multiple copies of the non-uniform advice of A'' and thus, A''' can execute A'' many times in the Goldreich-Levin reduction. Alternately, we can use the quantum Goldreich-Levin theorem [AC02].

Finally, we complete the proof of strong security in Theorem 4.5, we restate the theorem for convenience.

Theorem A.6 (Theorem 4.5)**.** *Let $G : \{0,1\}^\lambda \to \{0,1\}^{\ell(\lambda)}$ be a wQPRG that has pseudodeterminism $1 - O(\lambda^{-c})$ and pseudorandomness $1 - \delta$ such that $c > 1$, $\delta(\lambda) \leq 0.49 + o(1)$ and $\ell(\lambda) > s(\lambda) \cdot \lambda$, where $s(\lambda) = \Theta(\lambda)$. Define the QPT algorithm $G^{\oplus s} : \{0,1\}^{s(\lambda)\lambda} \to \{0,1\}^{\ell(\lambda)}$ as $G^{\oplus s}(k_1, \dots, k_s) := \bigoplus_{i=1}^{s} G(k_i)$. Then $G^{\oplus s}$ is a sQPRG with pseudodeterminism $1 - O(\lambda^{-(c-1)})$ and output length $\ell(\lambda)$.*

Proof of Strong Security.

1. By "pseudorandomness implies next-bit unpredictability" in Lemma A.3, for a random seed k and any $i \in \{0,1\}^{\ell(\lambda)}$, the probability of outputting y_i given $y_{[1:i-1]}$ is at most $1/2 + \delta$ (this is where we need that $\delta(\lambda) \leq 0.49 + o(1)$), where $y \leftarrow G(k)$.

2. By Lemma A.4, for any $i \in \{0,1\}^{\ell(\lambda)}$, the probability of computing $y^1_{i+1}|| \dots ||y^{2s}_{i+1}$ from $y^1_{[1:i]}|| \dots \ ||y^{2s}_{[1:i]}$ for independent, uniform seeds $k_1, \dots, k_{2s}$ is at most $\varepsilon = e^{-\Omega(s)} = \mathsf{negl}(\lambda)$.

3. By Lemma A.5, for any $i \in \{0,1\}^{\ell(\lambda)}$, the probability of computing the XOR of $y^1_{i+1}, \dots, y^s_{i+1}$ (the $(i+1)$-th bit of $G^{\oplus s}(k_1, \dots k_s)$) from $y^1_{[1:i]} \oplus \dots \oplus y^s_{[1:i]}$ (the first i bits of $G^{\oplus s}(k_1, \dots k_s)$) for independent, uniform seeds $k_1, \dots, k_{2s}$ is at most $1/2 + \mathsf{poly}(s\varepsilon) = 1/2 + \mathsf{negl}(\lambda)$.

4. By "next-bit unpredictability implies pseudorandomness" in Lemma A.3, we can conclude that $G^{\oplus s}$ is $(1 - \ell(\lambda) \cdot \mathsf{poly}(s\varepsilon))$-pseudorandom, where $\ell(\lambda) \cdot \mathsf{poly}(s\varepsilon) = \mathsf{negl}(\lambda)$.

□